



HAL
open science

Artificial Intelligence for Ecosystem Monitoring using Remote Sensing and Digital Agriculture Data

Valentine Bellet

► **To cite this version:**

Valentine Bellet. Artificial Intelligence for Ecosystem Monitoring using Remote Sensing and Digital Agriculture Data. Artificial Intelligence [cs.AI]. Université de Toulouse, 2024. English. NNT : 2024TLSES013 . tel-04674275

HAL Id: tel-04674275

<https://theses.hal.science/tel-04674275v1>

Submitted on 21 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Doctorat de l'Université de Toulouse

préparé à l'Université Toulouse III - Paul Sabatier

Intelligence artificielle appliquée aux séries temporelles
d'images satellites pour la surveillance des écosystèmes

Thèse présentée et soutenue, le 29 février 2024 par

Valentine BELLET

École doctorale

SDU2E - Sciences de l'Univers, de l'Environnement et de l'Espace

Spécialité

Surfaces et interfaces continentales, Hydrologie

Unité de recherche

CESBIO - Centre d'Etudes Spatiales de la BIOSphère

Thèse dirigée par

Jordi INGLADA et Mathieu FAUVEL

Composition du jury

Mme Marie CHABERT, Présidente, Toulouse INP

Mme Francesca BOVOLO, Rapporteur, Bruno Kessler Foundation

M. Dino IENCO, Rapporteur, INRAE Occitanie-Montpellier

Mme Anne PUISSANT, Examinatrice, Université de Strasbourg

Mme Charlotte PELLETIER, Examinatrice, Université Bretagne Sud

M. Jordi INGLADA, Directeur de thèse, CNES

M. Mathieu FAUVEL, Co-directeur de thèse, INRAE Occitanie-Toulouse

Membres invités

M. Mickaël SAVINAUD, Groupe CS

ABSTRACT

In the context of climate change, ecosystem monitoring is a crucial task. It allows to better understand the changes that affect them and also enables decision-making to preserve them for current and future generations. Land Use and Land Cover (LULC) maps are an essential tool in ecosystem monitoring providing information on different types of physical cover of the Earth's surface (e.g. forests, grasslands, croplands). Nowadays, an increasing number of satellite missions generate huge amounts of free and open data. In particular, Satellite Image Time Series (SITS), such as the ones produced by Sentinel-2, offer high temporal, spectral and spatial resolutions and provide relevant information about vegetation dynamics. Combined with machine learning algorithms, they allow the production of frequent and accurate LULC maps. This thesis is focused on the development of pixel-based supervised classification algorithms for the production of LULC maps at large scale. Four main challenges arise in an operational context. Firstly, unprecedented amounts of data are available and the algorithms need to be adapted accordingly. Secondly, with the improvement in spatial, spectral and temporal resolutions, the algorithms should be able to take into account correlations between the spectro-temporal features to extract meaningful representations for the purpose of classification. Thirdly, in wide geographical coverage, the problem of non-stationarity of the data arises, therefore the algorithms should be able to take into account this spatial variability. Fourthly, because of the different satellite orbits or meteorological conditions, the acquisition times are irregular and unaligned between pixels, thus, the algorithms should be able to work with irregular and unaligned SITS. This work has been divided into two main parts. The first PhD contribution is the development of Stochastic Variational Gaussian Processes (SVGP) on massive data sets. The proposed Gaussian Processes (GP) model can be trained with millions of samples, compared to few thousands for traditional GP methods. The spatial and spectro-temporal structure of the data is taken into account thanks to the inclusion of the spatial information in bespoke composite covariance functions. Besides, this development enables to take into account the spatial information and thus to be robust to the spatial variability of the data. However, the time series are linearly resampled independently from the classification. Therefore, the second PhD contribution is the development of an end-to-end learning by combining a time and space informed kernel interpolator with the previous SVGP classifier. The interpolator embeds irregular and unaligned SITS onto a fixed and reduced size latent representation. The obtained latent representation is given to the SVGP classifier and all the parameters are jointly optimized w.r.t. the classification task. Experiments were run with Sentinel-2 SITS of the full year 2018 over an area of 200 000 km² (about 2 billion pixels) in the south of France (27 MGRS tiles), which is representative of an operational setting. Results show that both methods (i.e. SVGP classifier with linearly interpolated time series and the spatially informed kernel interpolator combined with the SVGP classifier) outperform the method used for current operational systems (i.e. Random Forest with linearly interpolated time series using spatial stratification).

Index terms: Earth Observation, Artificial Intelligence, Ecosystems, Large Scale Classification, Land Use and Land Cover maps, Satellite Image Time Series, Stochastic Variational Gaussian Processes, Sentinel-2, Irregular and Unaligned time series, Representation Learning

Dans un contexte de changement climatique, la surveillance des écosystèmes est une mission essentielle. En effet, cela permet de mieux comprendre les changements qui peuvent affecter les écosystèmes et ainsi prendre des décisions en conséquence afin de préserver les générations actuelles et futures. Les cartes d'occupation du sol sont un outil indispensable fournissant des informations sur les différents types de couverture physique de la surface de la Terre (e.g. forêts, prairies, terres agricoles). Actuellement, un nombre accru de missions satellites fournissent un volume important de données gratuites et librement accessibles. Les séries temporelles d'images satellites (SITS), dont celles issues de Sentinel-2, grâce à leurs très hautes résolutions, informent sur la dynamique de la végétation. Des algorithmes d'apprentissage automatique permettent de produire de manière fréquente et régulière des cartes d'occupation du sol à partir de SITS. L'objectif de cette thèse est le développement d'algorithmes de classification supervisée pour la production de cartes d'occupations du sol à grande échelle. Dans un contexte opérationnel, quatre principaux défis se dégagent. Le premier concerne le volume important de données que les algorithmes doivent être capables de gérer. Le second est lié à la prise en compte des corrélations entre les variables spectro-temporelles et leur extraction pour la classification. Le troisième, quant à lui, correspond à la prise en compte de la variabilité spatiale : pour des zones géographiques étendues, la statistique de la donnée n'est pas stationnaire. Enfin, le quatrième défi concerne l'utilisation de SITS irrégulièrement échantillonnées et non alignées, principalement dû aux conditions météorologiques (e.g. nuages) ou à des dates d'acquisitions différentes entre deux orbites. Cette thèse est divisée en deux contributions principales. La première contribution concerne la mise en place de processus gaussiens variationnels stochastiques (SVGP) pour des SITS à grande échelle. Des millions d'échantillons peuvent être utilisés pour l'apprentissage, au lieu de quelques milliers pour les processus gaussiens (GP) traditionnels. Des combinaisons de fonctions de covariance ont été mises en place permettant notamment de prendre en compte l'information spatiale et d'être plus robuste vis à vis de la variabilité spatiale. Cependant, les SITS sont ré-échantillonnées linéairement indépendamment de la tâche de classification. La deuxième contribution concerne donc la mise en place d'un ré-échantillonnage optimisé pour la tâche de classification. Un interpolateur à noyau prenant en compte l'information spatiale permet de produire une représentation latente qui est donnée à notre SVGP. Les expérimentations ont été menées avec les SITS de Sentinel-2 pour l'ensemble de l'année 2018 sur une zone d'environ 200 000 km² (environ 2 milliards de pixels) dans le sud de la France (27 tuiles MGRS). Ce dispositif expérimental est représentatif d'un cadre opérationnel. Les résultats obtenus montrent que les modèles issus des deux contributions sont plus performants que la méthode utilisée pour les systèmes opérationnels actuels (i.e. forêts d'arbres aléatoires avec des SITS linéairement ré-échantillonnées utilisant la stratification spatiale).

Mots clés : Télédétection, Intelligence artificielle, Écosystèmes, Classification à grande échelle, Carte d'occupation du sol, Séries temporelles d'images satellites, Processus gaussiens variationnels stochastiques, Sentinel-2, Séries temporelles irrégulières et non alignées, Apprentissage de représentations

REMERCIEMENTS

Tout d'abord, un grand merci à mes deux directeurs de thèse Jordi Inglada et Mathieu Fauvel qui m'ont encadrée et accompagnée durant ces 3 ans (et 6 mois !) de thèse. Merci de m'avoir transmis votre expertise en télédétection et en intelligence artificielle, merci pour vos précieux conseils et votre suivi sans relâche. Vous étiez tous les deux très complémentaires et j'ai beaucoup apprécié nos échanges. Merci *Mathieu* de m'avoir partagé ton expertise en *math* et plus précisément sur les processus gaussiens. Merci d'avoir pris le temps d'écrire et de réécrire les mêmes équations encore et encore ! J'ai beaucoup apprécié ta rigueur mathématique et surtout merci d'avoir été présent dès que j'en avais besoin. Je te remercie aussi pour tes conseils et ton accompagnement pour mon futur professionnel. Merci *Jordi* de m'avoir partagé ton expertise en *ordi* (j'étais obligée...). Ton pragmatisme, ta casquette d'ingénieur et tes conseils toujours très pertinents ont été précieux durant ces trois années. Un grand merci pour ton humour et toutes tes blagues, notamment celles issues de la relecture du manuscrit. Elles ont permis de rendre ce moment difficile beaucoup plus sympathique !

Un grand merci à l'ensemble des membres du jury, Francesca Bovolo, Dino Ienco, Marie Chabert, Charlotte Pelletier, Anne Puissant et Mickaël Savinaud, d'avoir accepté de juger ma thèse. Merci pour votre intérêt dans mon travail, ainsi que pour vos commentaires et suggestions très constructifs.

Je remercie également les membres de mon comité de thèse, Silvia Valero, Stéphane Girard et Thomas Oberlin, qui ont suivi ma thèse et m'ont donné de riches conseils.

Merci à l'équipe *iota2* pour m'avoir formé sur la chaîne, de m'avoir aidé à debugger (encore et encore !) et notamment un grand merci à Benjamin Tardy pour ton investissement et ta précieuse aide sans relâche lors de ces années de thèse.

Un grand merci à Gwendoline Le Corre, Melisande Albert et Silvia Valero pour les enseignements que j'ai effectué avec vous. Merci pour votre confiance et pour les libertés que vous m'avez laissées. J'ai beaucoup appris dans vos équipes pédagogiques et j'ai pris beaucoup de plaisir à travailler avec vous. Votre dynamisme, votre envie de transmettre et votre pédagogie sont incroyables, les étudiants ont beaucoup de chance de vous avoir ! Merci à tous les étudiants que j'ai eus au cours de ces trois dernières années dont certains que je n'ai malheureusement jamais rencontrés en présentiel à cause du COVID.

Ces années de thèse m'ont permis de continuer la médiation scientifique, mais cette fois-ci en passant de l'autre côté ! Je remercie Instants Sciences, Rachel pour les rencontres Exploreurs, toute l'équipe de Sciences en Bulle et plus particulièrement, Nadia qui m'a accompagnée au cours de cette aventure, et tous les doctorants qui ont participé à l'édition 2022. Merci à toute

l'équipe de Mentor'IA et notamment les organisateur.rice.s, Corinne, Marjorie et Dennis, ainsi que mes mentorées, Léonie, Sara et Hajar, pour l'ensemble des moments partagés.

Merci à l'équipe administrative et gestionnaire du CESBIO (Émilie, Delphine, Dominique, Ibrahim, Laura, Laurence, etc.) pour votre efficacité et votre aide ! Merci à Mehrez qui a été directeur du CESBIO pendant la plus grande durée de ma thèse et qui m'a donné l'opportunité d'intervenir à la journée internationale des filles et des femmes en sciences à destination de collégiennes et lycéennes. A la suite de cette journée, Elisa a effectué son stage de 3ème avec moi ! Je remercie également l'équipe IA du CESBIO pour les différents échanges que nous avons eu lors de nos réunions hebdomadaires, sans oublier les gâteaux... Merci Jordi, Mathieu, Julien, Silvia, Yoël, Iris, Katia, Kevin (et tous les anciens CDDs et stagiaires). Merci à Victor Cathala, stagiaire de M2 au CESBIO, qui a contribué à différentes études sur mon sujet de thèse. Merci à mes co-bureaux dans l'ordre d'apparition : Erwan, Esteban, Hué, Diane et Blanca, merci pour tous les moments qu'on a partagé ensemble ! Je remercie l'ensemble du CESBIO avec qui j'ai échangé lors des pauses-déjeuner et des pauses-café, des BBQ, des CESBIO plages, du Noël du CESBIO, du séminaire CESBIO, des apéros SMOS ou lors de conférences (LPS Bonn, IGARSS Pasadena). Merci aux adultes (dixit Jordi) et je pense plus particulièrement à Arnaud, Claire, Olivier, Simon, Stéphane, Sylvain, Tiphaine, Vincent, Youen pour les différents moments passés ensemble. Mais surtout merci à la team des non-permanents, actuels et anciens, du CESBIO : Ainhoa, Andrea, Blanca, Clémence, Diane, Edna, Flo, Gaëtan, Gaith, Henry, Hugo, Hélène, Iris, Johan, Juan, Juliette, Jérémy, Katia, Kevin, Laura, Martin, Mathilda, Micael, Nitu, Paul, Pierre, Richard, Rémi, Simon, Thibault, Vincent, Yann, Yoël, Zacharie, Zied (et désolée pour ceux que j'oublie !). Entre non-permanents, on a réussi à bien se serrer les coudes et à profiter tous ensemble, j'ai passé de très bons moments avec vous ! Je remercie plus particulièrement Juliette que j'ai rencontré lors de mon deuxième jour de thèse, qui a été une colocataire incroyable et une collègue extraordinaire, et qui restera une amie fabuleuse, merci pour tout. Si j'ai tenu ces 3 années de thèse, c'est surtout grâce à toi !

Merci à tous mes amis (pas besoin de vous citer vous vous reconnaitrez !), vous m'avez aidé à me vider l'esprit et à passer de très bons moments à vos côtés que ce soit lors de blablarun, de sorties vélo, de randos, de sorties ski/raquettes, de concerts, de brunch, d'aprem thé, de repas, de soirées déjantées (notamment à la coloco), de podcasts, de week-end, de vacances, de mariages, etc. Vous êtes extraordinaires.

Un grand merci à ma famille, mes parents, mes frères, ma sœur, mes belles-sœurs et mes nièces. Merci pour votre soutien, vos encouragements, votre amour, je vous aime de tout mon cœur. I per fi, gràcies a tu, per tot, t'estimo molt.

Abstract	3
Résumé	5
Remerciements	7
I. Introduction	21
General introduction	23
Introduction en français	29
1. Ecosystem monitoring using remote sensing data	37
1.1. Ecosystem importance	38
1.1.1. What is an ecosystem?	38
1.1.2. Monitoring ecosystem functions	39
1.1.3. Land Use and Land Cover (LULC) maps	41
1.2. Earth Observation	43
1.2.1. Elements of remote sensing	43
1.2.2. Short history of remote sensing	48
1.2.3. Satellite Image Time Series (SITS)	51
1.3. Production methods for LULC Maps	57
1.3.1. Manual methods	57
1.3.2. Automatic methods	58
1.3.3. Main operational LULC maps	62
2. Pixel-based supervised land cover classification using SITS at large scale	65
2.1. Challenges	66
2.1.1. Large amounts of data	66
2.1.2. Spatio-spectro-temporal structure	67
2.1.3. Spatial variability	68
2.1.4. Irregular and unaligned SITS	68
2.2. State of the art	70
2.2.1. Machine learning methods	70
2.2.2. Deep learning methods	74
2.2.3. Preprocessing techniques	78
2.3. Remaining challenges and contributions of this thesis	80

3. Description of the data used	83
3.1. Sentinel-2 image time series	84
3.1.1. Description	84
3.1.2. Products	84
3.1.3. Study area	87
3.2. Data preparation for Sentinel-2 SITS	88
3.2.1. Radiometric and geometric corrections	88
3.2.2. Spatial resampling	88
3.2.3. Feature extraction	88
3.2.4. Spatial information extraction	89
3.2.5. Temporal resampling	89
3.3. Reference data	96
3.3.1. Sources	96
3.3.2. Polygons	100
3.3.3. Eco-climatic regions	101
3.4. Data set selection	103
3.4.1. Polygon selection	103
3.4.2. Pixel selection	103
II. Gaussian Processes for land cover classification using SITS at large scale	105
4. Review on Gaussian Processes	107
4.1. Gaussian Distribution	108
4.1.1. Univariate Gaussian Distribution	108
4.1.2. Multivariate Gaussian Distribution	109
4.2. Univariate Gaussian Processes	113
4.2.1. Definition	113
4.2.2. Gaussian Process Regression	118
4.2.3. Binary Classification	124
4.3. Multivariate Gaussian Processes	125
4.3.1. Definition	125
4.3.2. Multi-output regression	128
4.3.3. Multi-class classification	132
4.4. Large scale Gaussian Processes	133
4.4.1. Model Approximation	134
4.4.2. Posterior Approximation by Variational Inference	137
5. SVGP classification: Method and experimental set-up	141
5.1. Large scale multi-class GP land cover classification	142
5.1.1. Training	142
5.1.2. Inference	144
5.1.3. Hyper-parameters	145
5.1.4. Model complexity	147

5.2. Experimental set-up	149
5.2.1. Configuration	149
5.2.2. Data set generation	149
5.2.3. Method set-up	153
5.2.4. Map production	156
5.2.5. Feature reduction	156
6. SVGP classification: Results	161
6.1. Comparison with competitive methods	162
6.1.1. Quantitative results	162
6.1.2. Qualitative results	170
6.2. Boundary study	174
6.2.1. Quantitative results	174
6.2.2. Qualitative results	175
6.3. Model evaluation	179
6.3.1. Hyper-parameters selection	179
6.3.2. Trainable parameters initialization	183
6.4. Analysis of the characteristics of the GP model	186
6.4.1. Posterior predictive distribution	186
6.4.2. Learned model parameters	189
6.5. Feature reduction	191
Perspectives	195
III. Attention-based interpolation with Gaussian Processes for land cover classification using irregular and unaligned SITS at large scale	197
7. Review on temporal resampling	199
7.1. Standard temporal resampling methods	201
7.1.1. Imputation methods	201
7.1.2. Filtering methods	203
7.1.3. Kernel-based methods	207
7.2. Transformer methods for temporal resampling	210
7.2.1. Main concepts of the attention mechanisms	210
7.2.2. Attention-based interpolation	211
7.3. Multi Time Attention Networks (mTAN)	212
8. EmTAN-SVGP classification: Method and experimental set-up	215
8.1. Spatially informed interpolator for GP classification	216
8.1.1. Spectro-temporal feature reduction	218
8.1.2. Spatial positional encoding	218
8.1.3. Trainable parameters	220
8.2. Experimental set-up	221
8.2.1. Data set generation	221

8.2.2.	Methods set-up	221
8.2.3.	Map production	224
9.	EmTAN-SVGP classification: Results	225
9.1.	Comparison with competitive methods	226
9.1.1.	Quantitative results	226
9.1.2.	Qualitative results	231
9.1.3.	Robustness to the temporal sampling	234
9.2.	Model evaluation	237
9.2.1.	Spectral and temporal feature reduction	237
9.2.2.	Spatial positional encoding	240
9.2.3.	Influence of the number of inducing points	242
9.3.	Analysis of the spatially informed interpolator	243
9.3.1.	Latent representation	243
9.3.2.	Versatility of the similarity kernel	244
IV.	General Conclusion	247
10.	Conclusion and perspectives	249
10.1.	Summary	249
10.2.	Perspectives	250
10.2.1.	Short-term	250
10.2.2.	Mid- and long-term	252
	Conclusion en français	255
V.	Appendices	257
Appendix A.	Supervised classification tools	259
A.1.	Data set selection	259
A.2.	Validation and accuracy assessment	260
A.2.1.	Training and validation losses	260
A.2.2.	Confusion matrices and metrics	261
A.2.3.	Statistical tests	265
Appendix B.	SVGP classification: Additional results	267
B.1.	Additional results: Comparison with competitive methods	267
B.1.1.	F-score	267
B.1.2.	F-score per class	269
B.1.3.	Precision and recall per class	272
B.1.4.	Confusion matrices	275
B.2.	Additional results: Feature extraction	283
B.2.1.	F-score	283

Appendix C. EmTAN-SVGP classification: Additional results	285
C.1. Additional results: Comparison with competitive methods	285
C.1.1. Confusion matrices	285
C.2. Additional results: Robustness to the temporal sampling	288
C.2.1. Precision and recall per class	288
Appendix D. Reproducible research	291
D.1. Part II	291
D.1.1. Data sets	291
D.1.2. Best trained models	291
D.1.3. Land cover maps	291
D.1.4. Code	291
D.2. Part III	291
D.2.1. Data sets	291
D.2.2. Best trained models	292
D.2.3. Land cover maps	292
D.2.4. Code	292
Acronyms	292
References	299

LIST OF FIGURES

1.	Comic strips "Sciences en Bulles".	32
2.	Photo taken at the launch of the comic book "Science en Bulles"	33
1.1.	Landscape structure components.	40
1.2.	Landscape structure examples.	40
1.3.	Active and passive sensors.	45
1.4.	Optical image.	45
1.5.	Comparison spectral signatures different land covers.	46
1.6.	Comparison spectral signatures at different times.	46
1.7.	NDVI vs time for different land covers.	47
1.8.	Number of satellites in orbit at the start of 2023.	49
1.9.	Aerial photographs of the CESBIO area.	50
1.10.	Comparison spatial resolution.	53
1.11.	Influence temporal resolution.	55
1.12.	Yearly volume of satellite data.	56
1.13.	Example of decision tree.	59
1.15.	Comparison between pixel-based and object-based approaches.	62
1.16.	Example of LUCAS survey.	63
2.1.	Representation of the spatial, spectral and temporal dimensions for one pixel.	67
2.2.	Mean spectral profiles of winter crops in three different locations.	68
2.3.	Illustration of two real irregular and unaligned pixel time series acquired by Sentinel-2.	69
2.4.	Pixel-based supervised classification.	70
2.5.	Linear SVM	72
2.6.	Kernel SVM	72
2.7.	Multilayer Perceptron (MLP) with one hidden layer.	75
2.8.	Comparison between satellite image from Beijing-2, ground truth and land cover maps obtained with a CNN-based model.	76
2.9.	Gaussian Processes are not fashionable.	81
2.10.	Spatial discontinuity in land cover classification computed with RF models between two eco-climatic regions.	82
3.1.	Number of articles mentioning "Sentinel-2" in several journals.	85
3.2.	Products of level 1C, 2A and 3A.	86
3.3.	Study area	87
3.4.	Spatial resampling	88
3.5.	Sentinel-2 features extraction	90
3.6.	Lambert 93 projection.	91

3.7.	Sentinel-2 orbits used for the study area.	91
3.8.	Temporal grids for three different tiles: $T30TXQ$, $T31TCH$, $T31TFL$	92
3.9.	NDVI time series for three pixels from different tiles: $T30TXQ$, $T31TCH$, $T31TFL$	93
3.10.	Valid dates in the study area	94
3.11.	Averaged number of valid dates per pixel for each month.	94
3.12.	Interpolated NDVI time series for three pixels from different tiles: $T30TXQ$, $T31TCH$, $T31TFL$	95
3.13.	Representation polygons.	100
3.14.	Repartition of the polygons in the study area.	100
3.15.	Eco-climatic regions in the study area.	101
3.16.	Surface (in km^2) of each eco-climatic region in the study area.	102
3.17.	Data set selection.	104
4.1.	Standard normal random variable's PDF.	108
4.2.	PDF bivariate Gaussian distribution.	110
4.3.	Two representations of 4 different realizations of a bivariate Gaussian distribution.	112
4.4.	Representation of 4 different realizations of a d dimensional Gaussian distribution.	112
4.5.	Representation of the realizations of two different Gaussian Processes.	113
4.6.	Illustration covariance functions.	117
4.7.	Representation of data used for the univariate regression example.	120
4.8.	Comparison of predictions with different regressors from the univariate regression.	120
4.9.	Comparison of predictions with different covariance functions from the univariate regression.	121
4.10.	Comparison of predictions with different length-scale values from the univariate regression.	122
4.11.	LMC configuration	127
4.12.	Representation of data used for the multi-output regression example.	130
4.13.	Comparison of predictions with two configurations of covariance functions from the multi-output regression.	131
4.14.	Comparison computation time with several number of training inputs	134
5.1.	Model for the prediction of one new input \mathbf{x}_*	145
5.2.	Number of trainable parameters for the λt -GP model as a function of the number of spectro-temporal features.	148
5.3.	Pixels used for the training in stratification and global configurations.	149
5.4.	Synthetic representation of a buffered zone.	152
5.5.	Real example of a buffered zone.	152
5.6.	Feature extraction as a pre-processing.	157
5.7.	End-to-end feature extraction.	158
6.1.	Boxplots of the OA for each competitive method in both configurations.	163
6.2.	Wilcoxon rank-sum tests results for each competitive method in both configurations.	164

6.3.	Barplots of the F-score per class for each competitive method in both configurations.	166
6.4.	Normalized confusion matrices for $\phi\lambda t$ -GPPC and $\phi\lambda t$ -LTAE in both configurations.	168
6.5.	Land cover map computed for each competitive method in both configurations.	173
6.6.	Land cover maps obtained on an boundary zone between two eco-climatic regions (tile $T31TCJ$)	178
6.7.	Comparison between OA and training times in seconds for different number of inducing points: $M \in \{30, 50, 100, 250\}$	180
6.8.	Comparison between OA and training times in seconds for different number of latent process: $L \in \{11, 23, 46\}$	180
6.9.	Comparison between OA and prediction times in seconds for different number of draws: $\{10, 50, 100\}$	182
6.10.	Land cover map with different number of draws ($\{10, 50, 100\}$) for three different runs.	182
6.11.	Boxplot of the OA for different methods of the selection of inducing points. . .	184
6.12.	Posterior predictive distributions for a correct and incorrect predict class membership.	187
6.13.	Joint density of the standard deviation and the mean of the posterior predictive distribution for the selected class membership and their respective marginal densities.	188
6.14.	Spatial location of inducing points (IP) for 2 different latent GP.	190
6.15.	Boxplot of the distribution of the length-scale values for each latent GP. . . .	191
6.16.	Comparison of the OA for the different feature extraction methods in both configurations.	193
7.1.	COR NDVI time series	201
7.2.	Imputation methods for the COR NDVI time series.	203
7.3.	Spline interpolation methods for the COR NDVI time series.	205
7.4.	Savitzky-Golay filter methods for the COR NDVI time series.	206
7.5.	Representation of the previous standard functions $D(t)$	208
7.6.	Kernel-based interpolation methods for the COR NDVI time series.	209
8.1.	End-to-end learning for the classification of one irregular and unaligned pixel time series \mathbf{X}^* and its associated representation \mathbf{Z}	216
8.2.	Number of trainable parameters θ_2 based on the number of inducing points M and the number of spectro-temporal features $R \times D'$	223
9.1.	Boxplots of the overall accuracy (OA) for the competitive methods.	227
9.2.	Barplots of the averaged metrics per class for each studied model computed over nine runs.	228
9.3.	Normalized confusion matrices for each studied method.	230
9.4.	Land cover map computed for each competitive method on an agricultural area around Toulouse.	232
9.5.	Land cover map computed for each competitive method on an other agricultural area around Toulouse.	233

9.6.	Boxplots of the OA for the EmTAN-SVGP and raw-LTAE models computed with artificially shifted acquisition dates from the test data	235
9.7.	Barplots of the F-score per class for the EmTAN-SVGP and raw-LTAE models computed with the test data set limited to the <i>T31TCJ</i> tile over nine runs. . .	236
9.8.	Averaged OA for several number of latent dates, number of latent spectral features and number of heads.	238
9.9.	Averaged training times for several number of latent dates, number of latent spectral features and number of heads.	239
9.10.	Spatial positional encoding computed over a regular grid of spatial coordinates.	241
9.11.	Comparison of three NDVI time series profiles for a pixel labeled COR (raw, gapfilled, representation).	243
9.12.	Normalized attention values computed on three different latent dates.	245
A.1.	Examples of sampling methods for two different data sets.	260
A.2.	Training and validation losses in over-fitting, under-fitting and good-fitting (optimum).	260
A.3.	Comparison between two selection methods for multiple data sets.	264
B.1.	Boxplots of the F-score for each competitive method in both configurations. .	268
B.2.	Barplots of the precision per class for each competitive method in both configurations.	273
B.3.	Barplots of the recall per class for each competitive method in both configurations.	274
B.4.	Normalized confusion matrices for each competitive method in both configurations.	282
B.5.	Comparison of the Fscore for the different feature extraction methods in both configurations.	284
C.1.	Normalized confusion matrices for each studied method.	287
C.2.	Barplots of the precision per class for the EmTAN-SVGP and raw-LTAE models computed with the test data set only on the <i>T31TCJ</i> tile over nine runs. .	288
C.3.	Barplots of the recall per class for the EmTAN-SVGP and raw-LTAE models computed with the test data set only on the <i>T31TCJ</i> tile over nine runs. . . .	289

LIST OF TABLES

1.1. ECV requirements for LULC maps.	41
1.2. LULC maps nomenclature.	42
1.3. Examples of active and passive sensors.	44
1.4. SITS characteristics for different satellites.	52
1.5. Characteristics of different LULC maps.	64
3.1. Description of the Sentinel-2 bands.	85
3.2. Description of the Sentinel-2 products.	86
3.3. OSO nomenclature.	97
3.4. CORINE (CLC 2012) nomenclature.	98
3.5. RPG nomenclature.	99
3.6. Eco-climatic regions nomenclature.	101
4.1. Description mean functions.	114
4.2. Description covariance functions.	115
4.3. Time and storage complexities for the main GP approximation methods.	140
5.1. Nomenclature used in Chapters 5 and 6	143
5.2. Number of trainable parameters for the several GP models.	148
5.3. Average number of pixels per class and regions for the <i>classification</i> data set.	151
5.4. Number of extracted pixels in the <i>boundary</i> data set for each buffer size.	153
5.5. Number of trainable parameters for each model in the <i>global</i> configuration classification.	155
5.6. Parameter values for the Adam optimizer for <i>GP</i> , <i>MLP</i> and <i>LTAE</i>	156
5.7. Different combinations for the temporal reduction.	157
5.8. Comparison of the different methods used for the feature extraction.	159
6.1. Averaged training and prediction times for each competitive method in both configurations.	169
6.2. Averaged percentage of agreement (between two adjacent models) for different sizes of boundary zones.	175
6.3. Averaged OA computed on labeled pixels for different sizes of boundary zones.	175
8.1. Nomenclature used in Chapters 8 and 9	217
8.2. Description of the trainable parameters θ_1 and their corresponding sizes.	220
8.3. Number of pixels for each data set.	221
8.4. Number of trainable parameters for each model.	224
8.5. Parameter values for the Adam optimizer for the models: Gapfilled-SVGP, EmTAN-SVGP, EmTAN-MLP, EmTAN-LTAE and raw-LTAE.	224

9.1. Averaged training and prediction times for each studied model.	231
9.2. Averaged OA and averaged training times with and without the spatial positional encoded matrix \mathbf{P}	240
9.3. Averaged OA and averaged training times for different number of inducing points M	242

Part I.
Introduction

Context

Over the last decade, the emergence of **Earth Observation (EO)** satellite missions with high revisit frequency and high spatial resolution has led to the availability of an unprecedented amount of data with heterogeneous modalities (e.g. optical and radar) at various resolutions (e.g. sub-metric and deca-metric). This massive amount of data provides relevant information about vegetation dynamics at large-scale. To fully benefit from this information, automatic methods are used to produce **Land Use and Land Cover (LULC)** maps. Conventionally, in machine learning, automatic methods require two steps: preprocessing and classification.

The preprocessing step is commonly used to improve the quality of the data for the classification process. Two main challenges occur in large scale: irregular and unaligned time series, and spatial variability. Regarding the first challenge, the **Satellite Image Time-Series (SITS)** contain clouds or cloud shadows which can interfere with the ground information. Therefore, preprocessing techniques are used to remove the unwanted elements and to realign the data. Concerning the second challenge, depending on climatic or topographic conditions, the same vegetation cover can have different spectro-temporal responses in different locations. Hence, preprocessing techniques, such as spatial stratification, are applied in order to be more robust to the spatial variability.

The classification step, independent from preprocessing, is used to assign a label to each pixel. One of the main current challenges of the classification algorithms is to extract relevant information from these massive amounts of data. Recently, with the emergence of deep learning, such frameworks with two steps that are optimized independently of each other, may be questionable. Indeed, major improvements have been observed using end-to-end learning, i.e., when the preprocessing step is learned jointly with the classifier. In this context of noisy data, Bayesian methods enable the combination of both steps without any preprocessing, while being robust and interpretable.

Contributions

In this PhD thesis work, I provide two main contributions

- The first contribution is the investigation of **Stochastic Variational Gaussian Processes (SVGP)** for large-scale **LULC** pixel-based classification with Sentinel-2 **SITS**. This development enables the training with millions of pixels, compared to few thousands for conventional **Gaussian Processes (GP)** methods. The spatial and spectro-temporal structure of the data is taken into account thanks to the inclusion of the spatial information in bespoke composite covariance functions provided by the **SVGP**. Besides, this development enables to reduce the spatial variability of the data. The **SITS** are temporally

re-sampled in a separate pre-processing step.

- The second contribution is the development of end-to-end learning by combining a time and space informed kernel interpolator with the previous **SVGP** classifier. The interpolator embeds irregular and unaligned **SITS** onto a fixed and reduced size latent representation. The obtained latent representation is given to the **SVGP** classifier and all the parameters are jointly optimized w.r.t. the classification task.

Outline of the thesis

The outline of this dissertation is organized as follow:

- **Part I:** This part introduces the different notions and challenges related to land cover classification at large scale. Chapter 1 describes how remote sensing data can be used for ecosystem monitoring. Chapter 2 proposes a review of the state-of-the-art classifiers used for pixel-based supervised land cover classification with **SITS** at large scale. Furthermore, the associated challenges are also introduced. Chapter 3 presents the data used (i.e. study area, Sentinel-2 **SITS**, reference data, etc.) in Parts II and III.
- **Part II:** This part corresponds to the first contribution. A review of **Gaussian Processes (GP)** is proposed in Chapter 4. Chapter 5 presents the method based on **Stochastic Variational Gaussian Processes (SVGP)** as well as the experimental set-up. The associated results are analyzed in Chapter 6.
- **Part III:** This part corresponds to the second contribution. A review of temporal re-sampling is proposed in Chapter 7. Chapter 8 presents the end-to-end learning method which combines a spatially informed kernel interpolator with the **SVGP** classifier introduced in Part II. This chapter also defines the experimental set-up. The associated results are presented in Chapter 9.
- **Part IV:** This part concludes the manuscript. A general conclusion and the perspectives are provided in Chapter 10.
- **Part V:** This part provides the appendices. The classification metrics used to assess the quality of the estimation are presented in Appendix A. Appendices B and C report additional results for Chapters 6 and 9, respectively. To ensure reproducibility, the data sets, the implementation of the models and also the produced land cover maps are provided in Appendix D.

Support

This PhD, supervised by Mathieu Fauvel and Jordi Inglada, has been done in the **CESBIO** laboratory in Toulouse. **CESBIO** is a joint unit with the **CNES**, the **CNRS**, the **IRD**, the **UT3** and the **INRAe**. This PhD was co-founded by CS-Group and by **CNES**. This PhD is supported by **ANITI** from Université Fédérale Toulouse Midi-Pyrénées under grant agreement (ANITI ANR-19-PI3A-0004). This PhD is part of the **ANITI** Chair "Fusion-based inference from heterogeneous data" held by Nicolas Dobigeon.

Benjamin Tardy, engineer at CS-Group provided support and help with the `iota2` software. Data and computational resources such as the **High-Performance Computing (HPC)** infrastructure were provided by **CNES**.

List of scientific productions

Peer-reviewed publications in international journals

Valentine Bellet, Mathieu Fauvel, Jordi Inglada, Julien Michel. End-to-end Learning for Land Cover Classification using Irregular and Unaligned SITS by Combining Attention-Based Interpolation with Sparse Variational Gaussian Processes, 2024. **(Accepted in JSTARS)**

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land Cover Classification With Gaussian Processes Using Spatio-Spectro-Temporal Features, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-21, 2023.

Oral communications

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land cover classification using Sparse Variational Gaussian Processes and spatio-spectro-temporal features. *Workshop "Earth Observation and Machine Learning for Agriculture"*, Feb 2023, Toulouse, France.

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Classification de séries temporelles massives d'images satellitaires par des processus gaussiens variationnels parcimonieux et des descripteurs spatio-spectro-temporels. *28° Colloque sur le traitement du signal et des images - GRETSI - Groupe de Recherche en Traitement du Signal et des Images*, Sep 2022, Nancy, France.

Posters

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Attention based interpolation coupled with feature extraction for land cover classification using sparse variational Gaussian Processes. *2023 IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, July 2023, Pasadena, USA.

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land cover mapping with Gaussian Processes at the country scale using sparse and variational approaches. *Living Planet Symposium 2022*, May 2022, Bonn, Germany.

Dissemination of my PhD work

Presentations of my PhD work:

- PhD student day of the **CESBIO** laboratory, December 2020, January 2022, January 2023 (abstract + oral presentation) + organization of the **CESBIO** PhD student day, December 2020.

- PhD student day for PhD students co-founded by the **CNES**, October 2022 (oral presentation + poster).
- PhD student day of the doctoral school (**SDU2E**), June 2022 (poster).
- PhD student day for PhD students in artificial intelligence from laboratories in the **OMP**, May 2022 (oral presentation + poster).
- PhD student day for PhD students in informatics and mathematics from **INRAe**, December 2020 (oral presentation).
- PhD student day for PhD and post-doc students from **ANITI**, November 2020 (abstract + oral presentation).

Technical presentation "Introduction to Gaussian Processes" as part of the seminars ds@cesbio, March 2021 (<https://src.koda.cnrs.fr/activites-ia-cesbio/ds-cb>).

Supervision

Supervision of an observation internship, Elisa a schoolgirl, (one week) November 2023.

Co-supervision of a Master's internship "Spectro-Temporal Feature Extraction for the classification on Satellite Image Time Series with Variational Gaussian Processes", Victor Cathala, currently an engineer at Magellium, from March to September 2022.

Science popularization

Civil society

- Participation in the 2022 edition of Sciences en Bulles: "climate wake-up call". Working in collaboration with scientific facilitators and an illustrator to produce comic strips on my thesis subject (between November 2021 and September 2022). Participation in the inauguration of the comic strip during the "Fête de la Science" at the Muséum national d'histoire naturelle in Paris (one day) and in the exhibition devoted to the comic strip in the hall of the central administrative building of **UT3** (two hours), October 2022.
- Meeting with a non-expert public as part of the "Rencontres Exploreurs" in the library in Gourdon (46), (one hour) February 2021.

Academic

- Mentoring of three women master's students for one school year as part of the "Mentor'IA" program organized by **ANITI**'s Gender Equality Commission, school years 2021-2022, 2022-2023 and 2023-2024.
- Intervention with schoolgirls during the Women in Science Day organized by **CESBIO**, (half a day) February 2023.
- Intervention with middle school students in Toulouse as part of a project entitled "the science of superheroes" run by Instant Science, (half a day) January 2023.

- Intervention with final year students in two high schools: Cahors (46), May 2021 and Blagnac (31), (one hour) June 2021.

Contexte

Au cours de la dernière décennie, l'émergence de missions satellitaires à haute fréquence de revisite et à haute résolution spatiale a conduit à la disponibilité d'une quantité sans précédent de données avec des modalités hétérogènes (e.g. optiques et radar) à diverses résolutions (e.g. sub-métrique et déca-métrique). Ces données massives fournissent des informations sur la dynamique de la végétation à grande échelle. Pour exploiter pleinement ces informations, des méthodes automatiques sont utilisées pour produire des cartes d'occupation du sol. Traditionnellement, dans l'apprentissage automatique, ces méthodes nécessitent deux étapes : le prétraitement et la classification.

L'étape de prétraitement est généralement utilisée pour améliorer la qualité des données en vue de la classification. Deux principaux défis se posent à grande échelle : les séries temporelles irrégulières et non alignées, et la variabilité spatiale. Concernant le premier problème, les séries temporelles d'images satellites contiennent des nuages ou des ombres de nuages qui peuvent interférer avec les informations au sol. Par conséquent, des techniques de prétraitement sont utilisées pour supprimer ces éléments indésirables et réaligner les données. A propos du deuxième défi, en fonction des conditions climatiques ou topographiques, la même couverture végétale peut avoir des réponses spectro-temporelles différentes dans des lieux distincts. Ainsi, des techniques de prétraitement, telles que la stratification spatiale, sont mises en place afin d'être plus robuste face la variabilité spatiale.

L'étape de classification, indépendante du prétraitement, est utilisée pour attribuer une catégorie à chaque pixel. L'un des principaux défis des algorithmes de classification est de traiter avec précision ces quantités massives de données. Récemment, avec l'émergence de l'apprentissage profond, un tel cadre avec deux étapes qui sont optimisées de manière indépendante, peut être remis en question. En effet, des améliorations majeures ont été observées en utilisant une méthode d'apprentissage de bout en bout, c'est-à-dire lorsque l'étape de prétraitement est apprise conjointement avec le classifieur. Dans ce contexte de données bruitées, les méthodes bayésiennes permettent de combiner les deux étapes sans aucun prétraitement, tout en étant robustes et interprétables.

Contributions

Dans cette thèse, mes deux principales contributions sont les suivantes :

- La première contribution concerne la mise en place de processus gaussiens variationnels stochastiques (SVGP) pour la classification supervisée de cartes d'occupation du sol à partir de séries temporelles d'images satellite (SITS) Sentinel-2 à grande échelle. Des millions d'échantillons peuvent être utilisés pour l'apprentissage, au lieu de quelques milliers pour les processus gaussiens (GP) traditionnels. Des combinaisons de fonctions de covariance ont été mises en place permettant notamment de prendre en compte

l'information spatiale et d'être plus robuste vis à vis de la variabilité spatiale. Les SITS utilisées sont ré-échantillonnées temporellement dans une étape de prétraitement distincte de la classification.

- La seconde contribution concerne le développement d'une méthode d'apprentissage de bout en bout : un interpolateur à noyau prenant en compte l'information spatiale est combiné avec le SVGP défini précédemment. L'interpolateur intègre les SITS irrégulières et non alignées dans une représentation latente fixe et de taille réduite. La représentation latente obtenue est donnée au SVGP et tous les paramètres sont optimisés conjointement par rapport à la tâche de classification.

Plan de la thèse

Cette thèse est structurée de la manière suivante :

- **Partie I** : Cette partie présente les différentes notions et les différents défis liés au développement de cartes d'occupation du sol à grande échelle. Plus précisément, le Chapitre 1 décrit comment les données de télédétection peuvent être utilisées pour la surveillance des écosystèmes. Le Chapitre 2 propose un panorama des méthodes de classification supervisée issues de la littérature utilisés pour la production de carte d'occupation du sol avec des séries temporelles d'images satellites (SITS) à grande échelle. Par ailleurs, les défis associés sont également présentés. Le Chapitre 3 présente les données utilisées (i.e. zone d'étude, données satellitaires Sentinel-2, données de référence, etc.) dans les Parties II et III.
- **Partie II** : Cette partie correspond à la première contribution. Un résumé sur les processus gaussiens (GP) est proposé dans le Chapitre 4. Le Chapitre 5 présente la méthode basée sur les processus gaussiens variationnels stochastiques (SVGP) ainsi que les expérimentations mises en place. Les résultats associés sont développés au Chapitre 6.
- **Partie III** : Cette partie correspond à la deuxième contribution. Une étude sur le ré-échantillonnage temporel est proposée dans le Chapitre 7. Le Chapitre 8 présente la méthode d'apprentissage de bout en bout qui combine un interpolateur à noyau prenant en compte l'information spatiale avec le SVGP présenté dans la partie II. Ce chapitre présente également les expérimentations. Les résultats associés sont présentés dans le Chapitre 9.
- **Partie IV** : Cette partie conclut le manuscrit. Une conclusion générale et des perspectives sont fournies dans le Chapitre 10.
- **Partie V** : Cette partie contient les différentes annexes. Les métriques de classification utilisées pour évaluer la qualité de l'estimation sont présentées dans l'Annexe A. Les Annexes B et C présentent les résultats supplémentaires pour les Chapitres 6 et 9, respectivement. Pour garantir la reproductibilité, les données, les modèles et les cartes d'occupation des sols produites sont fournis en Annexe D.

Et puis si vous n'avez pas le courage de lire cette thèse en entier, vous pouvez vous référer à la Figure 1 pour avoir un bref résumé.

3 SURVEILLER NOTRE PLANÈTE GRÂCE À L'INTELLIGENCE ARTIFICIELLE

Au-dessus de nos têtes, des satellites observent en permanence notre planète. Les données qu'ils récoltent sont de plus en plus nombreuses, et donc de plus en plus complexes à exploiter par des méthodes statistiques traditionnelles. C'est là qu'intervient désormais l'intelligence artificielle (IA) !

Dans mon laboratoire, des algorithmes d'IA ont déjà été mis en place pour cartographier l'occupation des sols à partir des images capturées chaque année par les satellites. Mais des améliorations sont toujours possibles... et c'est l'objectif de mon travail. À la clef : une précision accrue de la carte.

Avec le changement climatique, il est en effet fondamental d'observer nos écosystèmes avec attention pour mieux les comprendre et les préserver !



Chaque année, mon laboratoire produit une carte d'occupation du sol à l'échelle de la France métropolitaine sur laquelle on identifie différents milieux :

- Zones agricoles (blé, maïs, colza, etc.)
- Zones artificielisées (bâtiments et routes)
- Zones naturelles (forêts, landes, etc.)

Ces informations sont cruciales pour de nombreux travaux de recherche et applications. Elles servent par exemple à surveiller l'étalement urbain, à suivre l'évolution des terres agricoles... Ou encore à l'analyse des effets du changement climatique sur les écosystèmes.

Fait chaud, non ? Adieu.

Reviens vite.

Ces cartes sont issues des données fournies par les satellites Sentinel-2, lancés par l'ESA* en 2015 et 2017, qui se relaient pour balayer une même zone du territoire tous les cinq jours. Ils y recueillent des informations sur plusieurs bandes spectrales allant du visible à l'infrarouge.

À l'échelle de la France, les images produites sur une année entière constituent une base de données très volumineuse : environ 20 téraoctets de données, soit l'équivalent de 4 millions de photos de vacances!

* Agence spatiale européenne

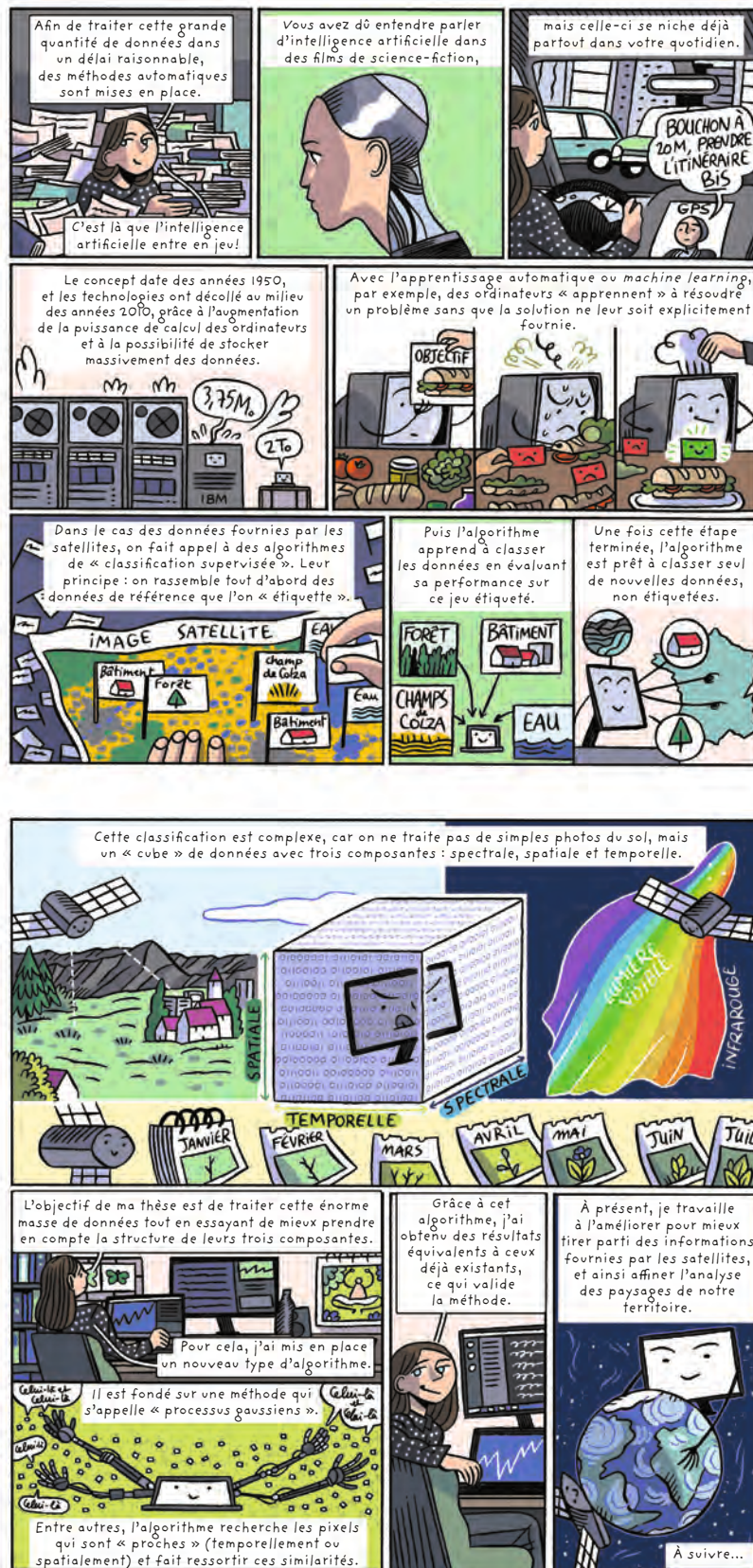


Figure 1: Planches issues de la bande dessinée Sciences en Bulles édition 2022 "réveil climatique". Elles proposent une vulgarisation scientifique de mon sujet de thèse. L'intégralité de la BD se trouve ici : <https://www.fetedelascience.fr/sciences-en-bulles-reveil-climatique>



Figure 2: Inauguration de la BD Sciences en Bulles en octobre 2022 lors de la fête de la Science au Muséum national d'histoire naturelle avec les autres doctorants de la BD, Fred et Jamy ainsi que Sylvie Retailleau, la ministre de l'enseignement supérieur de la recherche.

Financements

Cette thèse a été réalisée au sein du laboratoire **CESBIO** à Toulouse, sous la direction de Mathieu Fauvel et Jordi Inglada. Le **CESBIO** est une Unité Mixte de Recherche (UMR 5126) dont les tutelles sont : le **CNES**, le **CNRS**, l'**IRD**, l'**UT3** et l'**INRAe**. Ce doctorat a été co-financé par CS-Group et par le **CNES**. Cette thèse s'inscrit dans le cadre du projet **ANITI** de l'Université Fédérale Toulouse Midi-Pyrénées (ANITI ANR-19-PI3A-0004). Cette thèse est rattachée à la chaire **ANITI** "Inférence basée sur la fusion de données hétérogènes" de Nicolas Dobigeon.

Benjamin Tardy, ingénieur au CS-Group a apporté son soutien et son aide durant cette thèse notamment avec la chaîne de traitement iota^2 . Les données et les ressources informatiques telles que l'infrastructure Calcul Haute Performance (HPC) ont été fournies par le **CNES**.

Liste des productions scientifiques

Publications dans des revues internationales évaluées par des pairs

Valentine Bellet, Mathieu Fauvel, Jordi Inglada, Julien Michel. End-to-end Learning for Land Cover Classification using Irregular and Unaligned SITS by Combining Attention-Based Interpolation with Sparse Variational Gaussian Processes, 2024. **(Accepté à JSTARS)**

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land Cover Classification With Gaussian Processes Using Spatio-Spectro-Temporal Features, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-21, 2023.

Communications orales

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land cover classification using Sparse Variational Gaussian Processes and spatio-spectro-temporal features. *Workshop "Earth Observation and Machine Learning for Agriculture"*, Feb 2023, Toulouse, France.

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Classification de séries temporelles massives d'images satellitaires par des processus gaussiens variationnels parcimonieux et des descripteurs spatio-spectro-temporels. *28° Colloque sur le traitement du signal et des images - GRETSI - Groupe de Recherche en Traitement du Signal et des Images*, Sep 2022, Nancy, France.

Posters

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Attention based interpolation coupled with feature extraction for land cover classification using sparse variational Gaussian Processes. *2023 IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, July 2023, Pasadena, USA.

Valentine Bellet, Mathieu Fauvel, Jordi Inglada. Land cover mapping with Gaussian Processes at the country scale using sparse and variational approaches. *Living Planet Symposium 2022*, May 2022, Bonn, Germany.

Dissémination des travaux de thèse

Présentation de mes travaux de thèse :

- journée des doctorants du **CESBIO**, décembre 2020, janvier 2022, janvier 2023 (abstract + présentation orale) + organisation de la journée des doctorants du **CESBIO**, décembre 2020.
- journée des doctorants du **CNES**, octobre 2022 (présentation orale + poster).
- journée des doctorants de l'école doctorale **SDU2E**, juin 2022 (poster).
- journée des doctorants en intelligence artificielle des laboratoires de l'**OMP** (ENVIA), mai 2022 (présentation orale + poster).
- séminaire des doctorants du Pôle Informatique et Mathématiques pour les AgroBio-Sciences (IMABS) de l'**INRAe**, décembre 2020 (présentation orale).
- journée des doctorants et post-doctorants d'**ANITI**, novembre 2020 (abstract + présentation orale).

Présentation technique "Introduction aux Processus Gaussiens" dans le cadre des séminaires ds@cesbio, mars 2021 (<https://src.koda.cnrs.fr/activites-ia-cesbio/ds-cb>).

Encadrement

Encadrement d'un stage d'observation de 3ème, (une semaine) novembre 2023.

Co-encadrement d'un stage de Master 2 "Spectro-Temporal Feature Extraction for the classification on Satellite Image Time Series with Variational Gaussian Processes", Victor Cathala, actuellement ingénieur chez Magellium, de mars à septembre 2022.

Médiation scientifique

Grand public

- Participation à l'édition 2022 de Sciences en Bulles : "réveil climatique". Travail en collaboration avec des médiateurs scientifiques et une illustratrice pour produire des planches de BD sur mon sujet de thèse (entre novembre 2021 et septembre 2022). Participation à l'inauguration de la BD lors de la fête de la Science au Muséum national d'histoire naturelle à Paris (une journée) ainsi qu'à l'exposition consacrée à la BD dans le hall du bâtiment administratif central de l'UT3 (deux heures), octobre 2022.
- Rencontre avec un public non initié dans le cadre des "Rencontres Exploreurs" dans la bibliothèque de Gourdon (46), (une heure) février 2021.

Scolaire

- Accompagnement de trois étudiantes en master pendant une année scolaire dans le cadre du programme "Mentor'IA" organisé la Commission Mixité d'ANITI, années scolaires 2021-2022, 2022-2023 et 2023-2024.
- Intervention avec des collégiennes lors de la journée des femmes des sciences organisée par le CESBIO, (une demi journée) février 2023.
- Intervention avec une classe de 5ème à Toulouse dans le cadre d'un projet nommé "la science des super-héros" animé par Instant Science, (une demi journée) janvier 2023.
- Intervention avec des terminales dans deux lycées : Cahors (46), mai 2021 et Blagnac (31), (une heure) juin 2021.

CHAPTER 1

ECOSYSTEM MONITORING USING REMOTE SENSING DATA

1.1. Ecosystem importance	38
1.1.1. What is an ecosystem?	38
1.1.2. Monitoring ecosystem functions	39
1.1.3. Land Use and Land Cover (LULC) maps	41
1.2. Earth Observation	43
1.2.1. Elements of remote sensing	43
1.2.2. Short history of remote sensing	48
1.2.3. Satellite Image Time Series (SITS)	51
1.3. Production methods for LULC Maps	57
1.3.1. Manual methods	57
1.3.2. Automatic methods	58
1.3.3. Main operational LULC maps	62

1.1. Ecosystem importance

1.1.1. What is an ecosystem?

An ecosystem can be defined as the interaction between biotic components (i.e. living organisms such as plants, animals, bacteria) and abiotic components (i.e. non-living environments such as water, soil, atmosphere) [Chapin et al., 2011]. Biotic and abiotic components are highly interconnected. For example, without the abiotic component of soil, plants would not be able to survive and grow. Any disruption to one component can have cascading effects on the entire ecosystem.

Earth is composed of aquatic and terrestrial ecosystems. The terrestrial ecosystem to which humanity belongs to, covers only 30% of the Earth's surface, whereas the aquatic ecosystem covers more than 70%. Grasslands, forests or deserts are examples of terrestrial ecosystems [Knapp, 2020]. The aquatic ecosystem is composed of marine ecosystem (e.g. oceans and seas) and freshwater ecosystem (e.g. ponds, lakes, rivers, and streams) [Alexander and Fairbridge, 1999].

The term *ecosystem services* was defined by the Millennium Ecosystem Assessment (MEA), a study carried out from 2001 to 2005 which helped to evaluate the impact of anthropogenic activity on ecosystems [Millennium ecosystem assessment, 2005]. Ecosystem services are the direct and indirect contributions that ecosystems provide for human wellbeing and quality of life [Kremen and Ostfeld, 2005]. Ecosystem services are divided in four different categories [Millennium ecosystem assessment, 2005]:

- provisioning (e.g. water, food, wood, fuel),
- regulating (e.g. climate regulation, flood management),
- supporting (e.g. nutrient cycle, soil formation), and
- cultural (e.g. recreation, aesthetic).

Taking the example of the Pyrenees, the nearest mountains from Toulouse, various ecosystem services are provided. First, the stored water supplies drinking water, generates electricity and supports agriculture. Forests play an important role in climate regulation with the absorption of the carbon dioxide from the atmosphere. Forests also prevent soil erosion by stabilizing the soil with their roots. Finally, the Pyrenees are a popular destination for tourists: they have high recreational (e.g. hiking, skiing, etc.) and cultural (transhumance, Bethmale cheese, etc.) services.

Ecosystems provide services that depend on their function. Indeed, ecosystem services are only a sub-set of *ecosystem functions* for human needs [de Groot et al., 2010]: they are the benefits that humans obtain from ecosystem functions. For example, pollination corresponds to an ecosystem function where pollinators, such as bees, butterflies, birds, and bats, facilitate the transfer of pollen between flowers. It enables plant reproduction and fruit production, which results into crop production [Klein et al., 2007], [Vanbergen, 2013]. In this specific case, pollination, an ecosystem function, can also be considered as a valuable ecosystem service which

contributes to domestic (meadows, entomophilous crops) and wild plant production [Eardley et al., 2016].

1.1.2. Monitoring ecosystem functions

Ecosystems are continuously evolving. A driver can, directly or indirectly, cause a change in an ecosystem [Nelson, 2005]. A *direct driver* influences directly the ecosystem (e.g. invasive species), whereas, an *indirect driver* behaves diffusely by altering direct drivers and as well as other indirect drivers (e.g. socio-economic and demographic trends). Currently, changes due to anthropic actions are becoming more frequent in terrestrial and aquatic ecosystems [Ratajczak et al., 2018].

In 2005, the MEA revealed that since 1950, more than 60% of ecosystems had been degraded. The changes have occurred more rapidly since the second half of the XXth century than in any time in recorded human history [Millennium ecosystem assessment, 2005]. Around one quarter (24%) of the terrestrial ecosystems have been converted in cultivated systems. Moreover, around 35% of mangroves were lost, as well as 20% of coral reefs. More recently, the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services (IPBES) reported that around 75% of the land use environment had been changed by human actions [Oggenoorh and Faith, 2013]. In March 2023, the final synthesis report of the Sixth Assessment Report (AR6) released by the Intergovernmental Panel on Climate Change (IPCC), shows that climate impacts on ecosystems are more intense and widespread than expected [IPCC, 2023].

It is not only abrupt changes, such as land use and land cover changes, that affect ecosystem functions. Some practices, such as intensive agriculture [Tsiafouli et al., 2015], can have long-term effects on the ecosystem functions. Moreover, the neighborhood of an ecosystem can also have an influence on it. For example, a crop field that cuts a forest in two, will drastically reduce mobility of animals in the forest and also the ecosystem functions in the forest. Thus, the landscape structure can be used to study the impacts on ecosystem functions. *Landscape structure* corresponds to the arrangement of different land covers and uses across a landscape [Fahrig et al., 2011]. It is divided in two components: composition and configuration. *Composition* corresponds to the number of different land covers or land uses. *Configuration* corresponds to the spatial pattern of these land cover or land use types. Figure 1.1 represents landscapes with different compositions and configurations. The more different land cover types there are, the more complex is the composition. The more different spatial patterns there are, the more complex is the configuration. In general, more biodiversity is found in agricultural landscapes with complex configuration and complex composition [Estrada-Carmona et al., 2022]. Figure 1.2 represents two different agricultural landscape structures: a complex one (Figure 1.2a) and a simple one (Figure 1.2b). It is essential to study the landscape structure in order to understand and predict the evolution of ecosystem functions. Spatial information is essential, as aggregated data does not provide all the information about the structure. Thus, one possible technique is to produce land use and land cover maps.



(a) Complex composition / Simple configuration



(b) Complex composition / Complex configuration

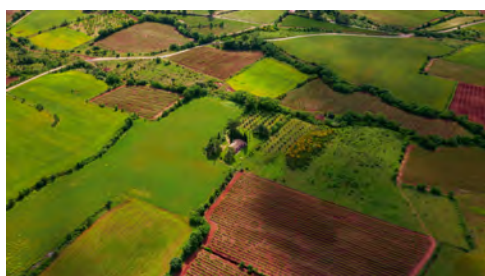


(c) Simple composition / Simple configuration



(d) Simple composition / Complex configuration

Figure 1.1: Representation of different landscape structures. Below left (c): structure with simple composition and simple configuration. Above right (b): structure with complex composition and complex configuration. Each color represents a different type of land cover.



(a) Complex configuration and complex composition (Salagou lake, France)



(b) Simple configuration and simple composition (Oregon, USA)

Figure 1.2: Representation of two different agricultural landscape structures. (Source for (a): <https://unsplash.com/photos/WPapb9IqRKw> and for (b): https://fr.wikipedia.org/wiki/Fichier:Crop_circles_north_of_Umatilla,_Oregon,_USA.jpg)

Table 1.1.: ECV product's requirements for LULC maps: goal and minimum requirements.

Level	Goal	Minimum requirement
Scale	Global	Regional
Spatial resolution (meter)	100-300	1000
Temporal resolution (month)	1	60
Production time (month)	3	60
Time span (year)	> 50	0
Measurement uncertainty ($2\sigma\%$ (including 95% confidence intervals))	5	35
Stability ($\%$ (including 95% confidence intervals))	5	25

1.1.3. Land Use and Land Cover (LULC) maps

A **Land Use and Land Cover (LULC)** map is a representation of the different types of land use and land cover in a given area over a given period of time. More precisely, land cover usually refers to the physical type (i.e. corn field or grassland) whereas land use indicates how people are using it (i.e. agriculture). Several components describe LULC maps properties: scale, spatial resolution, nomenclature, temporal coverage, temporal resolution, production time and time span. The *scale*¹ corresponds to the total area, it can be global, supranational, national, regional and local. The *spatial resolution*, for raster data, or **Minimum Mapping Unit (MMU)**, for vector image, is defined as "the smallest size areal entity to be mapped as a discrete entity" [Lillesand et al., 2015]. The *nomenclature* corresponds to the different land cover or land use types. The *temporal coverage* corresponds to the time period during which data were collected or observations were made. Usually the temporal coverage corresponds to one year and it is called "the reference year". The *temporal resolution* is defined as the time between each update. The *production time* or timeliness corresponds to the time between the last data and the map released. Finally, the *time span* is defined as the time period between the first and the last map.

The **Global Climate Observing System (GCOS)** identified land cover as one of the main important **Essential Climate Variables (ECV)** [Zemp et al., 2022]. They proposed different requirements for land cover maps [World Meteorological Organization (WMO); United Nations Educational and (ISC), 2022]. As stated in Table 1.1, the minimum suggested scale is regional and the minimum temporal resolution is five years. The goal is to produce LULC maps every month at global scale. They also suggested to use the **Land Cover Classification System (LCCS)** defined by the **Food and Agriculture Organization (FAO)** [Di Gregorio and Jansen, 1998]. The level 3 is composed of nine classes and its nomenclature is presented in Table 1.2. Other nomenclatures, proposed in the literature, are also presented in Table 1.2. A diversified nomenclature is needed to study complex landscape structures. The production of several LULC maps is described in Section 1.3. In order to meet these requirements, regular data on a global scale are needed. Thus, data collection and survey techniques, such as, **Earth Observation (EO)** are essential.

¹In this case, the scale does not correspond to the cartographic scale which denotes the size representation on the map compared to the object's size on the ground.

Table 1.2.: Examples of three different LULC maps nomenclatures.

Name	Classes
Anderson Level I [Anderson, 1976]	Urban or Built-up Land Agricultural Rangeland Forest Land Water Wetland Barren Land Tundra Perennial Snow or Ice
ICCS (level 3) [Di Gregorio and Jansen, 1998]	Cultivated and Managed Terrestrial Areas Natural and Semi-Natural Terrestrial Vegetation Cultivated Aquatic or Regularly Flooded Areas Natural and Semi-Natural Aquatic or Regularly Flooded Vegetation Artificial Surfaces and Associated Areas Bare Areas Artificial Waterbodies Snow and Ice Natural Waterbodies, Snow and Ice
International Geosphere Biosphere Programme (IGBP) [Belward, 1996]	Evergreen needleleaf forests Evergreen broadleaf forests Deciduous needleleaf forests Deciduous broadleaf forests Mixed forests Closed shrublands Open shrublands Woody savannas Savannas Grasslands Permanent wetlands Croplands Urban and built-up Cropland/natural Snow and ice Barren Water bodies

1.2. Earth Observation

EO corresponds to the acquisition of information about the Earth including remote sensing systems or in-situ data. Remote sensing systems, such as sensors on board of satellites or aircrafts, acquire information without direct contact with the object or the medium in contrast to in situ observations. Remote sensing systems allow to cover larger zones and give information more frequently than field surveys.

1.2.1. Elements of remote sensing

In order to better understand how remote sensing can be useful for monitoring ecosystems, few concepts are reviewed in the following.

Sensors that capture images in the remote sensing systems are mainly divided in two categories: active and passive, as shown in Figure 1.3. Active sensors provide their own illumination source and measure the energy that bounces back off objects on the Earth's surface. Passive sensors measure natural energy, usually sunlight, reflected off the Earth's surface or emitted by the Earth itself. Some examples of active and passive sensors are provided in Table 1.3. Optical images are produced by passive sensors, such as visible and near-infrared radiometers.

Optical images are a sampling of a spatial, spectral and temporal process, as shown in Figure 1.4. The spatial information is represented by the black squares, i.e. the pixels. The spectral information of one pixel is represented by a spectral profile. It is the reflectance across different wavelengths for a specific object. Reflectance corresponds to the ratio of the amount of light leaving a target with respect to the amount of light striking the target. The spectral information correspond to a specific date: the image was captured in August. In the following, spectral, spatial and temporal resolutions are defined more precisely.

Spatial resolution: Spatial resolution is defined as a measure of the smallest object that can be discriminated by the sensor. It is the ability to distinguish two closed objects.

Spectral resolution: Spectral resolution generally refers to the number of spectral bands and their width. Multi-spectral refers to the acquisition of around 5-10 bands whereas hyper-spectral refers to hundreds or thousands of narrower bands. Spatial and spectral resolutions are linked. It is difficult to have high spatial and spectral resolution at the same time. In addition to the number of bands, radiometric resolution is also important. Radiometric resolution defines the ability of a satellite to distinguish between different shades of color or gray in an image. With n bits, we have 2^n potential digital numbers between 0 to $2^n - 1$ to record the information. Current sensors have made it possible to increase the number of bands and the radiometric resolution.

To illustrate the importance of the spectral information, Figure 1.5 represents the spectral signature of different land covers. The spectral profiles were extracted from the ECOSTRESS spectral library [Baldridge et al., 2009], [Meerdink et al., 2019]. As shown in Figure 1.5, it is quite easy to differentiate the spectral signatures of the snow, the talc, the water or one of the vegetation samples in the visible spectrum. The spectral signature of grass and dry-grass are quite different. However, among vegetations, pine and oak have very similar spectral

Table 1.3.: Examples of active and passive sensors and their uses.

Sensor	Name	Use
Active	Synthetic Aperture Radar (SAR)	high-resolution imaging
Active	Laser Imaging, Detection, And Ranging (LIDAR)	3D imaging
Active	Radar Altimeter	measure the ocean topography
Active	Scatterometer	measure wind speed and direction
Passive	Visible and Near-Infrared Radiometer	optical imaging
Passive	Infrared Radiometer	optical imaging
Passive	Passive Microwave Radiometer	thermal imaging

signature in the visible bands. However, their spectral signatures are quite different in the infrared spectrum and it is easier to differentiate them.

In addition to the reflectance of the different bands, different vegetation indices can be defined [Xue et al., 2017]. One of the most used index is the **Normalized Difference Vegetation Index (NDVI)** which is used to distinguish the different vegetation covers. **NDVI** is calculated as a combination of the red (R) and near infrared (NIR) reflectance values [Rouse Jr et al., 1974]:

$$\frac{\text{NIR} - \text{R}}{\text{NIR} + \text{R}} \quad (1.1)$$

NDVI values range from -1 to 1 . Clouds and water are generally defined by negative values. Positive values near zero indicate absence of photosynthetic vegetation. Higher positive values of **NDVI** ranges from sparse ($0.1 - 0.5$) to dense (0.6 and above) photosynthetic vegetation.

Temporal resolution: Temporal resolution corresponds to the interval of time before a satellite revisits a particular point on the Earth's surface. Temporal resolution is also called time revisit.

The spectral signature is not enough to distinguish different land cover types and the temporal information is needed. Figure 1.6 represents three different vegetation land covers (corn, rapeseed and sunflower) at two different times of the year (in May and September). Depending on the time, different vegetation covers can have similar spectral signature. Indeed, rapeseed and sunflower have similar spectral signature in May, but different ones in September. Moreover, the same vegetation cover can have different spectral signature at different time of the year. Indeed, corn and rapeseed have completely different spectral signatures in May and in September. Depending on the period of the year, the plant has different stages of development. This cycle is called phenology. This can include flowering, leaf unfolding (or budburst), seed set and dispersal, and leaf fall in relation to climatic conditions [Davi et al., 2011]. In order to correctly identify a vegetation cover, it is important to take into account the temporal aspect. Generally, two different vegetation covers have different phenology cycles. In addition to the spectral signature, the representation of the **NDVI** as a function of the time is a major source of information. Figure 1.7 represents the **NDVI** for three different vegetation land covers (corn, rapeseed and sunflower) over the full year 2018. By plotting the **NDVI** versus the day of the year, corn, rapeseed and sunflower can be easily differentiated. Thus, the temporal information is very important and can help to differentiate two different land cover types. Having frequent dates makes it easier to monitor vegetation.

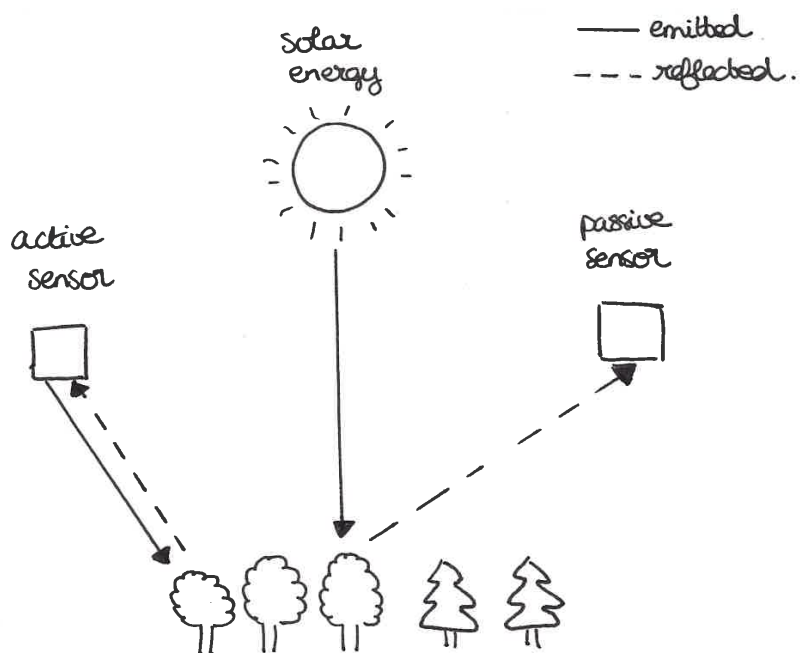


Figure 1.3: Active and passive sensors.

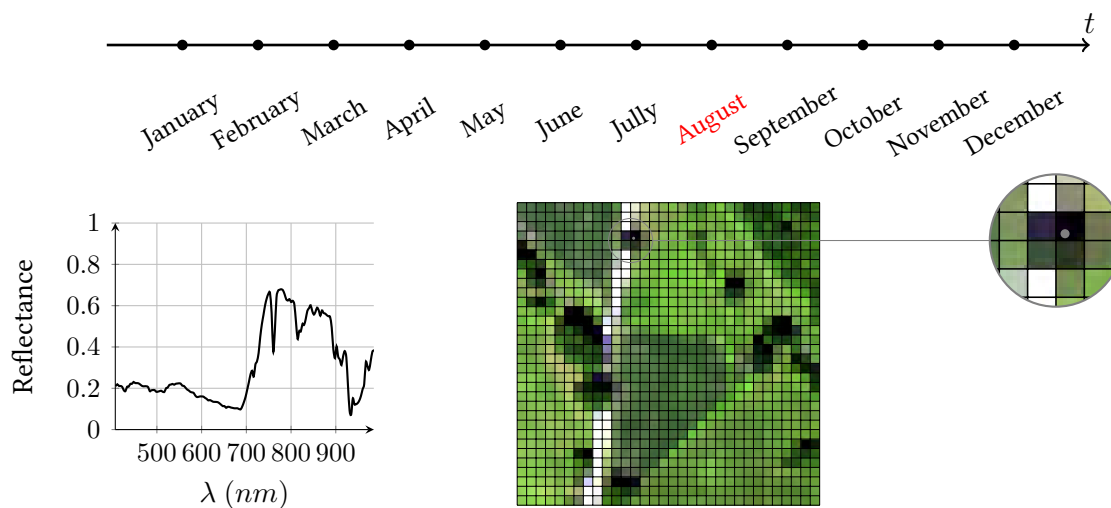


Figure 1.4: Optical image is a sampling of a spatial, spectral and temporal process. Black squares, in the optical image, correspond to pixels. For one pixel, the spectral signature is represented. This image was acquired in August.

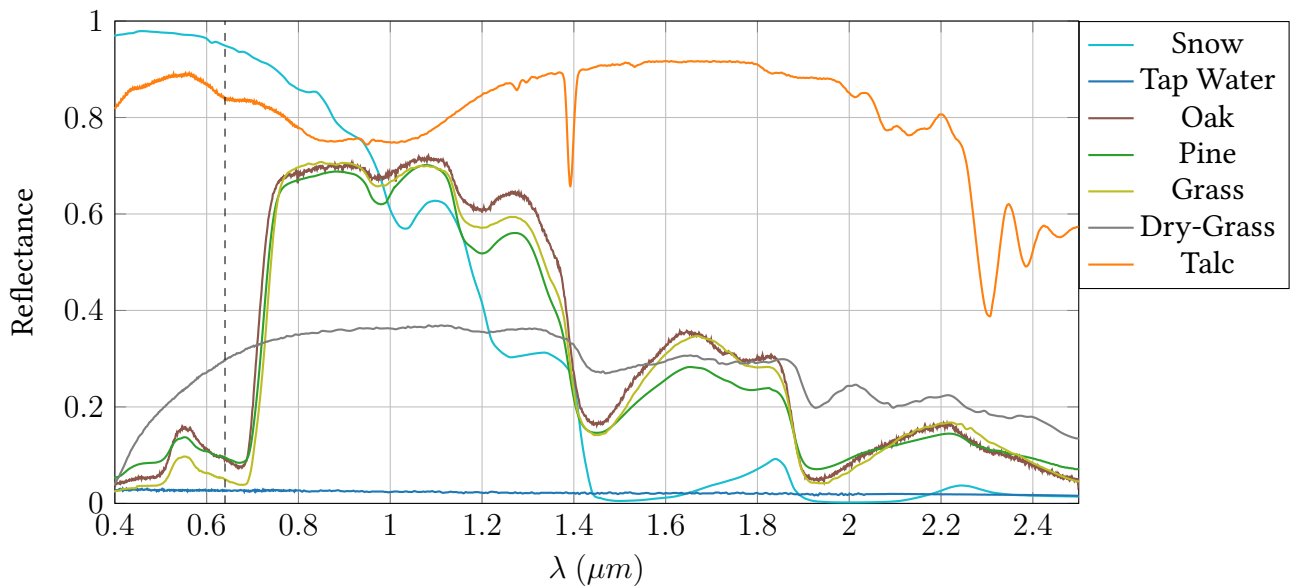


Figure 1.5: Spectral signatures of different land covers: snow, water, oak, pine, grass, dry-grass and talc. The vertical dashed line represents the limit between visible spectrum and infrared spectrum. (Source: ECOSTRESS spectral library version 1.0 <https://speclib.jpl.nasa.gov/>)

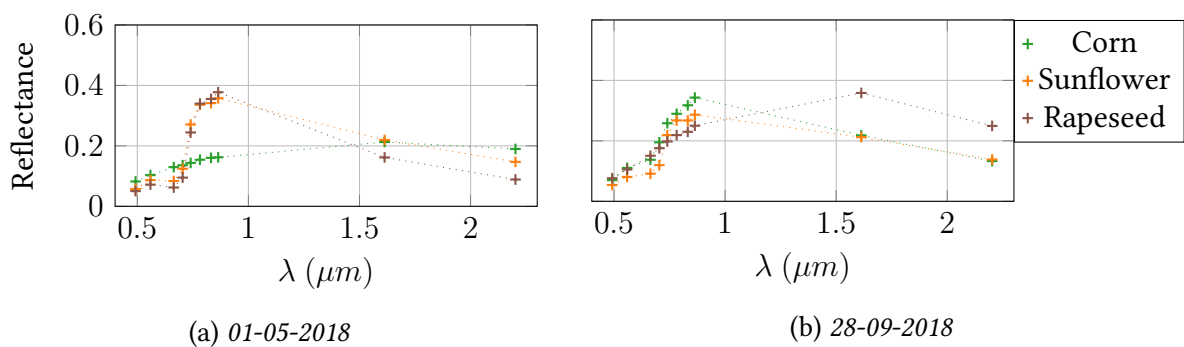


Figure 1.6: Spectral signatures of three different agricultural land covers: corn, sunflower and rapeseed at two different dates: 01/05/2018 and 28/09/2018. The spectral values correspond to the Sentinel-2's bands. (Source: <https://apps.sentinel-hub.com/sentinel-playground/>)

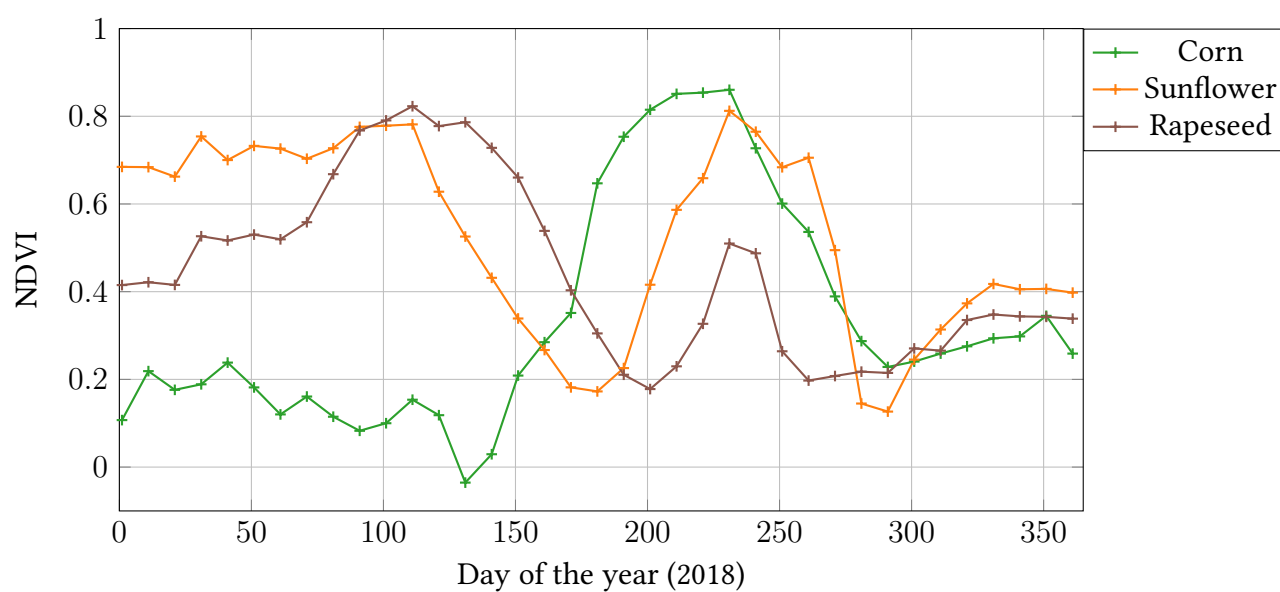


Figure 1.7: NDVI for three different agricultural land covers: corn, sunflower and rapeseed over the full year 2018. The NDVI was calculated from the Sentinel-2's bands with a linear interpolation of 10 days. (Source: <https://apps.sentinel-hub.com/sentinel-playground/>)

1.2.2. Short history of remote sensing

Remote sensing first appeared at the beginning of the 20th century with the development of aerial photography where cameras were mounted on aircrafts. After World Wars I and II, aerial photography began to be used for mapping and surveying purposes. Later, as sensor technologies improved, aircraft were equipped with other sensors, as for example multi-spectral sensors.

In the 1960s and 1970s, remote sensing technology took a big step forward with the launch of satellites that could take pictures of the Earth from space. The first satellite that was specifically designed for remote sensing was the Landsat-1 satellite², launched by **National Aeronautics and Space Administration (NASA)** in 1972. It carried a multi-spectral scanner and the study of the vegetation with **NDVI** was first introduced [Rouse et al., 1974].

The 1980s were characterized by the development of a large number of **EO** satellites by various countries: Bhaskara-I by India in 1979, SPOT-1 by France in 1986³, MOS-1 by Japan in 1987, etc. These satellites were equipped by several active and passive sensors, such as microwave radiometer, thermal radiometer, visible and infrared radiometer, synthetic aperture radar, etc. Most satellites were developed by space agencies (**NASA**, **CNES**, etc.) or by large aeronautic industry groups (Boeing, Airbus, etc.).

The 1990s and 2000s were marked by an explosion in the number of satellites, as technology improved. In 1999, Landsat-7 was launched and was able to provide images every 16 days with 30 m resolution on eight spectral bands. This satellite is part of the Landsat program managed by **NASA** and **United States Geological Survey (USGS)** which starts with the satellite Landsat-1. As a comparison, Landsat-1 provided images every 18 days with 80 m resolution on four spectral bands. **Moderate Resolution Imaging Spectroradiometer (MODIS)** is a passive sensor which has also made a significant contribution to remote sensing. It was launched aboard two **NASA** satellites, Terra, in 1999, and Aqua, in 2002. With these two satellites, images can be acquired every 1 or 2 days in 36 spectral bands at three spatial resolutions: 250 m, 500 m, and 1000 m. These decades have been also characterized by the emergence of private companies in the development and deployment of **EO** satellites. Indeed, in 1999, the first commercial remote sensing satellite Ikonos 1 was launched in the US. At the end of the 2000s, more than 150 **EO** satellites were in orbit [Tatem et al., 2008]. The 2000s were also marked by the development of **Unmanned Aerial Vehicles (UAV)**, also called drones [Everaerts et al., 2008]. They are small aircrafts that can fly without a pilot on board. These little platforms can be equipped by different sensors as **Laser Imaging, Detection, And Ranging (LIDAR)**, for example.

The 2010s were characterized by the launch of SPOT-6 and SPOT-7 satellites, respectively in 2012 and in 2014. These satellites are part of the SPOT program managed by the **CNES** and Airbus Defense and Space which started with SPOT-1. SPOT-6 and SPOT-7 provide satellite images on demand with high spatial resolution (i.e. four spectral bands at 6m and one panchromatic band at 1.5m). This decade was also characterized by the launch of the Sentinel-1 and Sentinel-2 satellites part of the Copernicus programme. The two satellites Sentinel-1, launched in 2014 and 2016, provide radar images whereas the two satellites Sentinel-2,

²Formerly named ERTS-A and then ERTS-1.

³Formerly named "Système Probatoire d'Observation de la Terre" and later "Satellite Pour l'Observation de la Terre". After a strike, this satellite was also called by **Centre National d'Études Spatiales (CNES)** agents: "Satellite Pour Occuper Toulouse" (<https://spacegate.cnes.fr/fr/spot-un-satellite-pour-occuper-toulouse>).

launched in 2015 and 2017, provide optical images. These satellites have revolutionized remote sensing by their resolution and also because the Copernicus Open Access Hub provides free and open access to Sentinel-1 and Sentinel-2 data.

More recently, due to the miniaturization of electronic components or the development of new materials, satellites could be much smaller and cheaper than their predecessors, which led to easier access to space. This period is called the "New Space" and different companies are involved, such as Planet Labs, Maxar Technologies, Capella Space, GIGSat, Satellogic, Umbra Lab, etc. For example, the company Planet Labs is characterized by the development of very small satellites called nanosatellites (about 10 kg) with different constellations. In January 2023, more than 1100 EO satellites were in orbit, with the largest number of satellites launched by the company Planet Labs, as illustrated in Figure 1.8. Aerial platforms are still used in remote sensing and huge archives of aerial photographs are available [Rapinel et al., 2018]. An example of aerial photographs of the *Centre d'Études Spatiales de la Biosphère (CESBIO)* laboratory from 1950 to today is represented in Figure 1.9.

Monitoring landscape structures involves long-term and large-scale projects. Satellites are the only tools that can systematically monitor large areas. Indeed, satellites follow predefined orbits, enabling them to capture data consistently over large areas, at regular intervals. Moreover, once launched, satellites can operate for several years, continuously collecting data at a fraction of the cost. In the following, we will focus on satellite images.

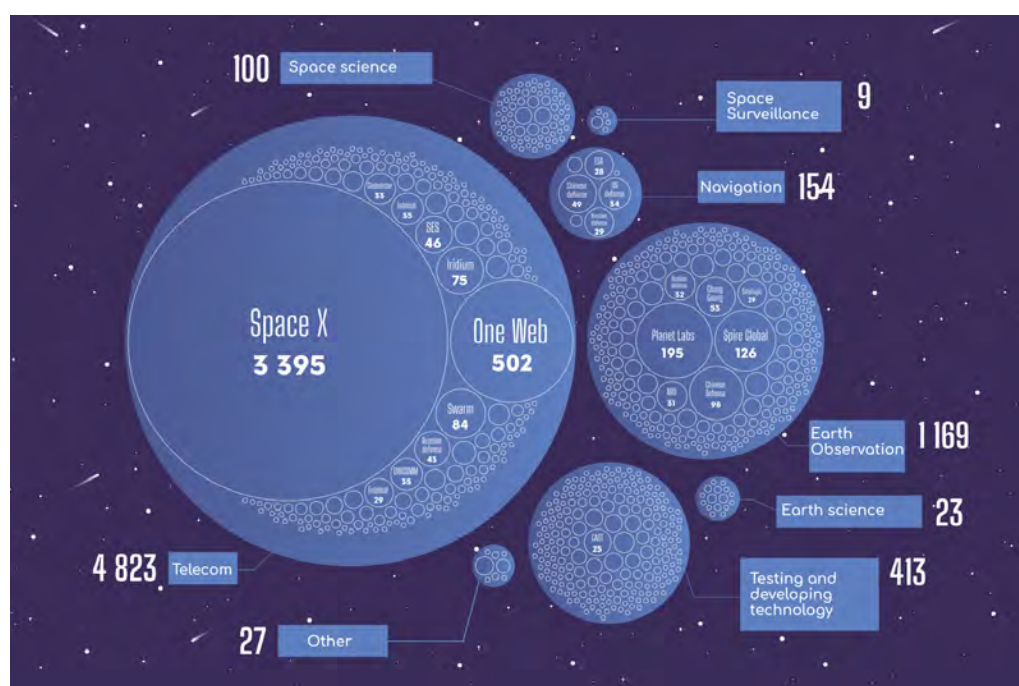


Figure 1.8: Number of satellites in orbit at the start of 2023. (Source: <https://www.arcep.fr/la-regulation/grands-dossiers-thematiques-transverses/lempreinte-environnementale-du-numerique/evenement-satellites-et-environnement.html>)

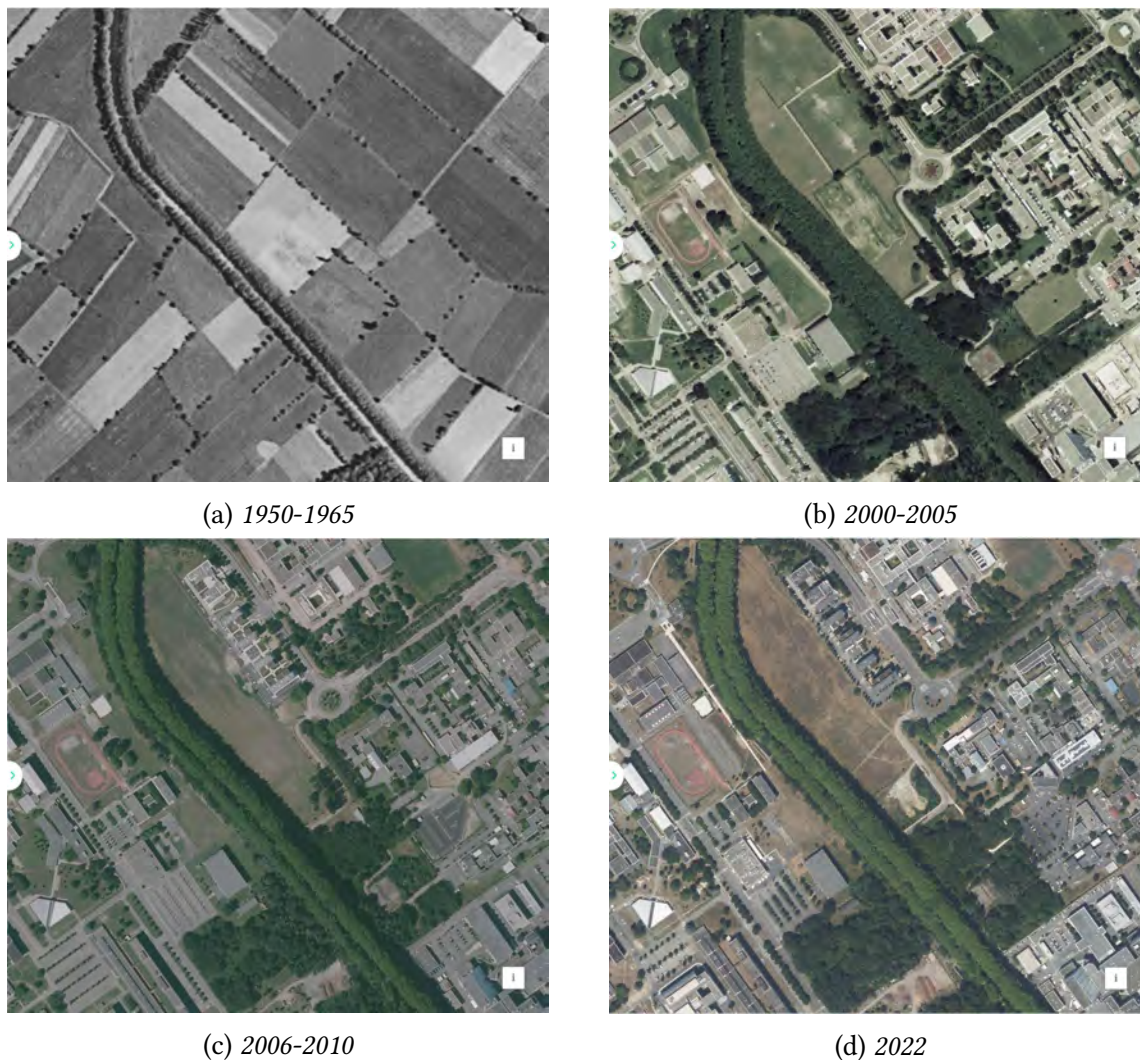


Figure 1.9: Aerial photographs of the CESBIO area from 1950 to today. Now, all the available aerial photography of France since 1919 from the Institut Géographique National (IGN) can be downloaded in the Geoportail website: <http://www.geoportail.gouv.fr/>. Between 1950 and 2000, there was a significant transformation of crops into buildings, which corresponds to the construction of the university and the laboratories. More recently, transformations have been more moderate, with some fields being converted into car parking lots.

1.2.3. Satellite Image Time Series (SITS)

Earth observation satellites are designed for different missions:

- tracking water resources, such as water management and irrigation [Deines et al., 2017], [Foster et al., 2020], estimation of evapotranspiration [Zhang et al., 2010], [Gallego-Elvira et al., 2013], water quality monitoring [Brando and Dekker, 2003], [Xu et al., 2019], [Lomelí-Huerta et al., 2021], [Niroumand-Jadidi and Bovolo, 2023] or snow cover monitoring [Romanov et al., 2000], [Gascoin et al., 2019];
- providing disaster response management, such as flood monitoring [Guo et al., 2021] or forest fire monitoring [Dell’Aglío et al., 2020];
- monitoring ecosystems, such as biodiversity monitoring [Tarantino et al., 2018], [Fauvel et al., 2020], agricultural monitoring [Feng et al., 2019], [Moeini Rad et al., 2019] or forest mapping [Karasiak et al., 2017], [Dalimier et al., 2021].

Depending on the application, different choices are made for the characteristics of the sensors. Considering ecosystem monitoring and more precisely **LULC** maps, the latter can be produced using either optical or radar satellite images. However, the majority of works are based on optical data [Congalton et al., 2014] and only very few studies use radar data [Longépé et al., 2011], [Abdikan et al., 2016]. The main reasons are that it is difficult to differentiate land covers with radar data. Spectral information in optical data allows for more precise discrimination between different land cover types especially for vegetation, whereas, some classes have similar radar backscatter responses. For these reasons, we will focus on optical satellite images.

Different compromises have to be made when designing Earth observation satellite missions. Technological constraints are limiting factors in achieving high spatial, spectral and temporal resolutions for optical images. These technological constraints include telescope size, signal-to-noise ratio, fields of view, orbits, revisits, on-board storage capacity, data transfer rate to the ground, etc. Thus, **EO** satellite missions are optimized for a specific type of application. Focusing on ecosystem monitoring and **LULC** maps, land covers such as crops are rapidly changing, in few days, and thus the temporal aspect is essential. Missions with a sufficient revisit time are required. In this work, we propose to work with **Satellite Image Time-Series (SITS)** which is defined as a sequence of images of the same location recorded at regular intervals throughout a given period of study. Table 1.4 represents **SITS** characteristics from three well-known **EO** missions used to produce **LULC** maps: Landsat 8-9, **MODIS** and Sentinel-2. By combining its two satellites, Landsat 8 and Landsat 9, images from Landsat are acquired every eight days. Sentinel-2, also a combination of two satellites (2A and 2B), provides images every five days. Finally, **MODIS** can provide images every one or two days.

In addition to the temporal aspect, the spatial aspect is also essential. Indeed, high spatial resolution is required to identify complex landscape structures. Figure 1.10 illustrates the capability of identifying structures in the image at different resolutions for **MODIS**, Landsat 8-9 and Sentinel-2 satellites. The area corresponds to the surroundings of the city Toulouse in May 2023 with true colors. With a spatial resolution of 500m, **MODIS** is well adapted for simple landscape structure such as the huge crop parcels in the USA (c.f. Figure 1.2b) and not to fragmented landscape structures such as in France (c.f. Figure 1.2a). With **MODIS** images, it is not possible to retrieve the landscape configuration for this area. However, with Landsat

Table 1.4.: *SITS characteristics for different satellites: Landsat 8-9, MODIS, Sentinel-2, SPOT 6-7.*

Satellite name	Spatial resolution (m)	Bands (#)	Temporal resolution (days)	Swath width (km)	Radiometric resolution (bits)
Landsat 8-9	30-15-100	9	8	185	12
MODIS	250-500-1000	36	1-2	2330	12
Sentinel-2	10-20-60	13	5	290	12

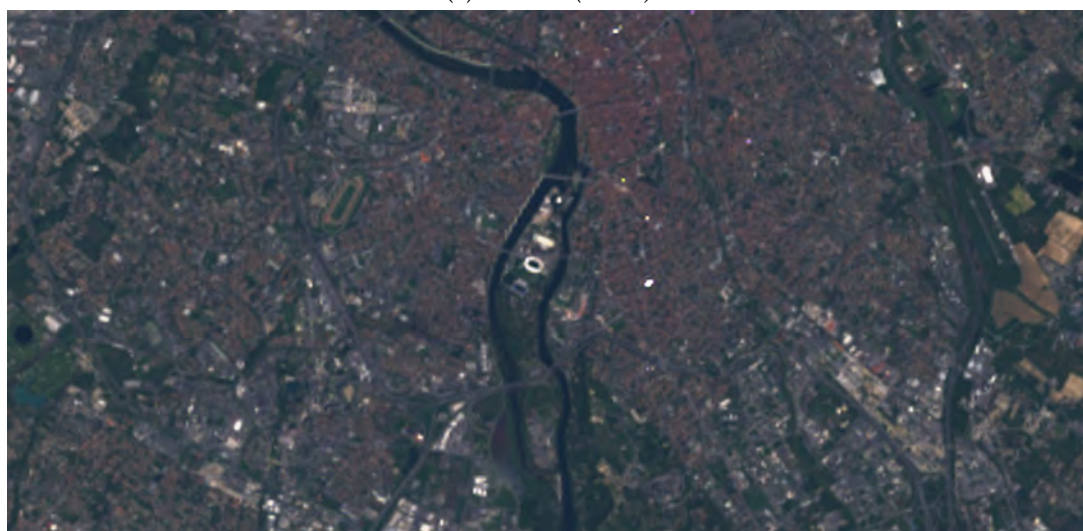
8-9 or Sentinel-2 images, the spatial patterns such as the river can be captured. Indeed, these two satellites have a finer spatial resolution, 30m and 10m, respectively. Moreover, Sentinel-2 image with its higher spatial resolution, provides more details about landscape objects than Landsat 8-9.

Figures 1.11a and 1.11b represent the different satellite images during one month for a field of rapeseed near to Toulouse for respectively, Sentinel-2 and Landsat 8-9. In this example, a rapid change in the land cover can be seen. Indeed, the flowers did not bloom at the beginning of the month and they stopped blooming at the end. As explained previously, the temporal information is really important to detect the modification on the phenology. However, even if Sentinel-2 and Landsat 8-9 provide data with a short revisit cycle, a large number of images can not be used due to the presence of clouds. In this example, for Sentinel-2, only 4 of the 7 images (05-04-2023, 10-04-2023, 20-04-2023, 05-05-2023) are usable. For Landsat 8-9, only 4 of the 10 images (10-04-2023, 18-04-2023, 19-04-2023, 04-05-2023) are valid. If clouds are present at a key moment in the phenological stage, this information is lost. Unlike optical data, radar sensors are not sensitive to clouds or to illumination conditions and can be jointly use with optical data. Recently, fusion between radar and optical data in areas with persistent cloud cover were proposed to improve the accuracy of land cover maps [Hill et al., 2020].

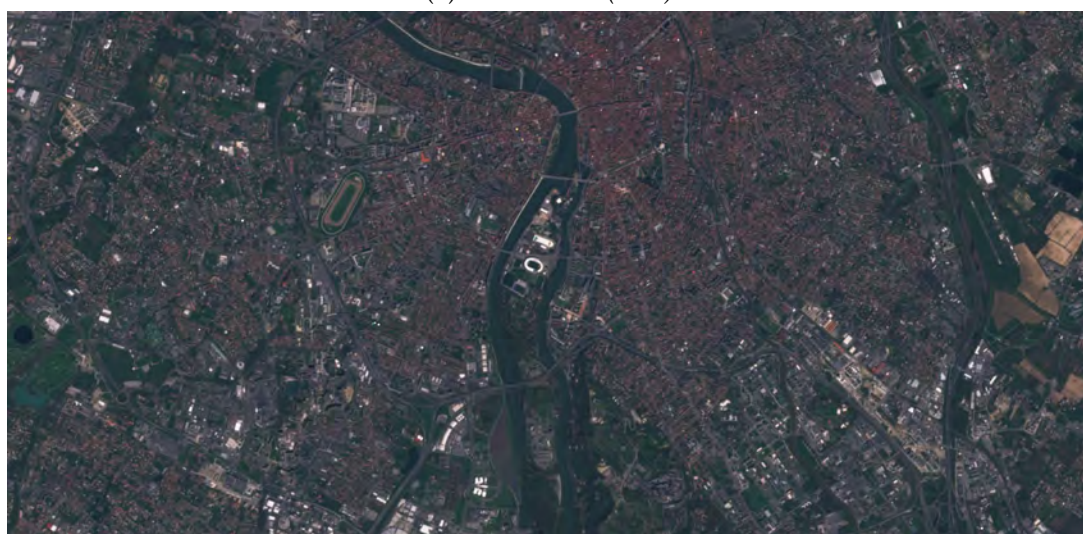
All these satellite data represent an unprecedented amount of data. Figure 1.12 shows that, in 2019, Sentinel-1 and Sentinel-2 represented around 3.5 petabytes of data, Landsat 7-8 around 0.25 petabytes and MODIS less than 0.25 petabytes. This difference in data volume is explained by the frequent temporal revisit and the high spatial resolution for Sentinel-2 compared to MODIS. In this work, we have decided to work with Sentinel-2 data in particular because of its frequent temporal revisit and its high spatial resolution. A complete description of the data used is provided in Chapter 3. In the following, several methods for the production of LULC maps are presented in order to deal with this huge amount of data.



(a) MODIS (500m)



(b) Landsat 8-9 (30m)



(c) Sentinel-2 (10m)

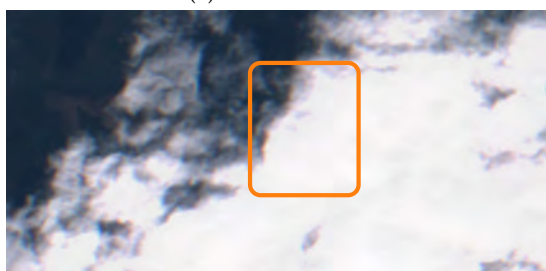
Figure 1.10: Image of Toulouse taken from different satellites with different spatial resolutions in May 2023 (in true colors).



(1) 05-04-2023



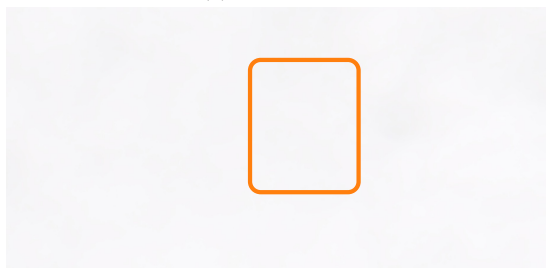
(2) 10-04-2023



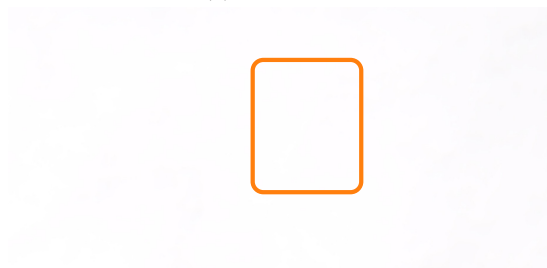
(3) 15-04-2023



(4) 20-04-2023



(5) 25-04-2023

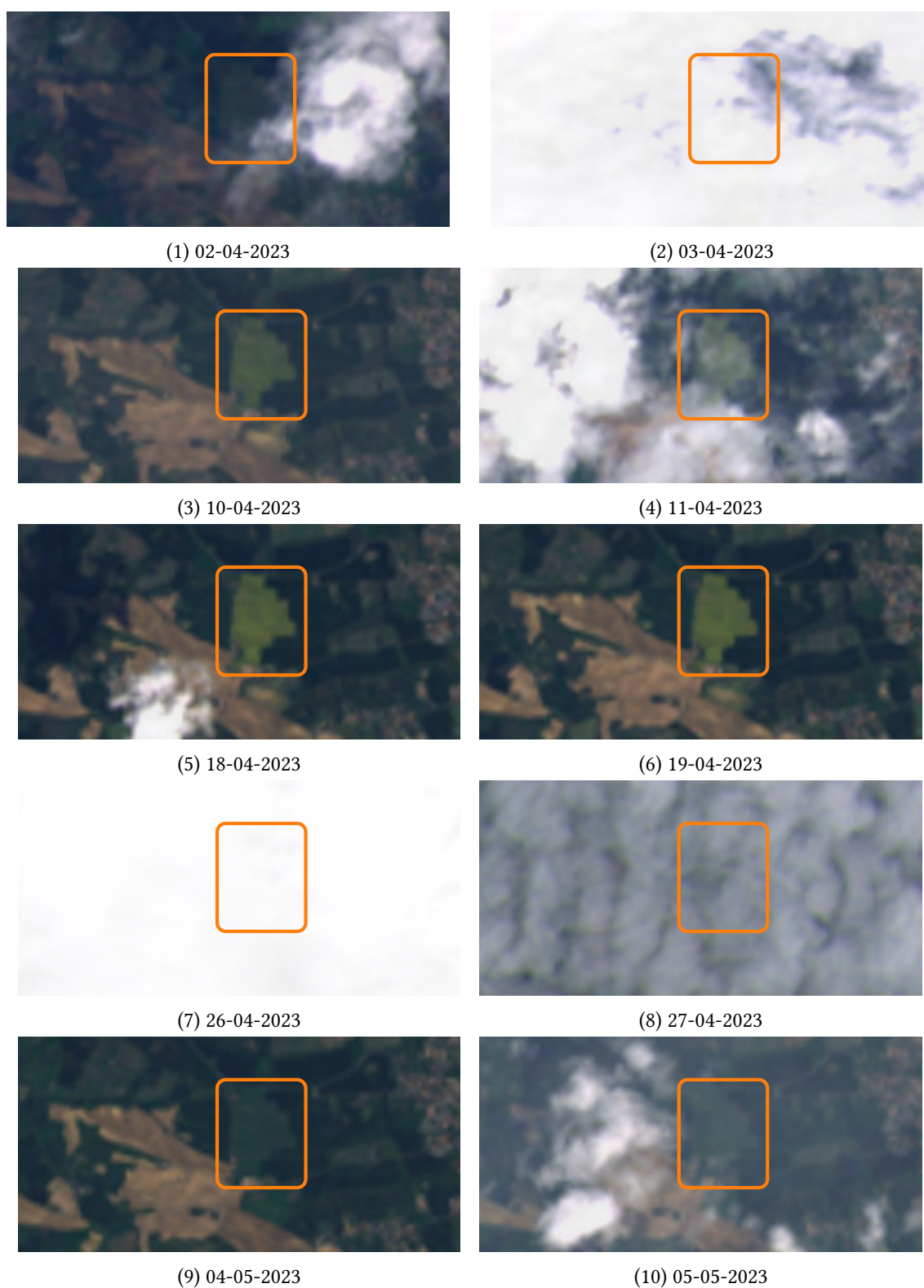


(6) 30-04-2023



(7) 05-05-2023

(a) *Sentinel-2*



(b) Landsat 8-9

Figure 1.11: Different satellite images in the agricultural area of Toulouse during around one month (from 02 April 2023 to 05 May 2023). The orange rectangle corresponds to a rapeseed field. White shapes correspond to clouds.

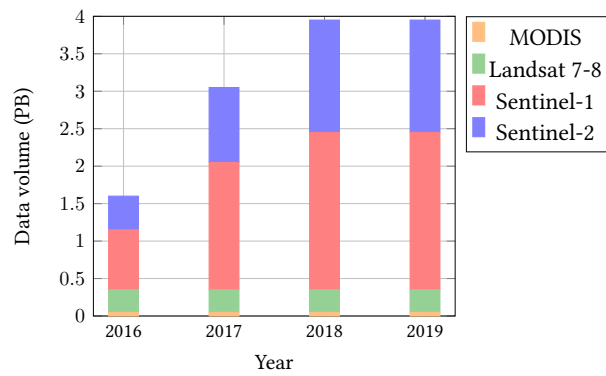


Figure 1.12: Data volume in petabytes (PB) from 2016 to 2019 for MODIS, Landsat 7-8, Sentinel-1 and Sentinel-2. (Source: [Soille et al., 2018]).

1.3. Production methods for LULC Maps

LULC maps are mainly produced from remote sensing images [Saah et al., 2020], and more precisely, from satellite image time series. In the beginning of remote sensing, images were analyzed by human operators. In recent years, mainly due to techniques able to easily process the huge amount of free and open access satellite data as well as better storage and computational power, an increasing number of LULC are produced with automatic methods [Hermosilla et al., 2022].

1.3.1. Manual methods

Different manual methods can be used to produce land cover and land use maps from satellite images. In general, these methods are combined with each other.

Photo interpretation

Photo interpretation or image interpretation corresponds to human experts that identify the different cover types with satellite images [Miller and Colwell, 1961]. Information is not only present in the pixels themselves, the geographical area or the climate needs to be taken into account. Several dates throughout the year are needed to correctly identify land covers and more precisely the different crops. Indeed, rapeseed is easy to identify in April on condition that there's at least one cloud-free image in this period, as shown in Figure 1.11. Each pixel needs to be assigned to a unique class label. Thus, photo-interpretation on a large area can be time consuming and costly. Another drawback of photo interpretation is the lack of consistency across areas and periods due to intra-operator variability (fatigue, inattention) as well as inter-operator variability (experience, knowledge of the area).

Field surveys

Photo-interpretation is usually combined with methods such as field surveys where human experts go on site and record the different cover types. To produce a land cover map at large scale, the results on the set of representative points are usually extrapolated. In order to distinguish between crops, field surveys often require several visits during the year. One of the major disadvantages of field surveys is that the schedule is very tight. Indeed, the surveyors are constrained by the weather, by the phenology or by the farmers. For example, a field survey needs to be done before the crop is harvested or before the flowers have finished blooming. Moreover, they require a huge number of experts and also plenty of time to cover large areas.

Crowd sourcing

Photo-interpretation and field surveys require a large number of experts. In crowd sourcing, non experts contribute to obtain information for experts with different collaborative tools such as web or mobile applications, social media, etc. This term can be applied to different projects, but when applied to geographic content it is called **Volunteered Geographic Information (VGI)**. Unlike field surveys, if a mistake is made, users can identify and correct it. The

first well-known map obtained with VGI is **OpenStreetMap (OSM)**⁴. Since 2004, it is updated and maintained by a large number of volunteers. Different data are used such as field surveys, aerial images or other freely licensed geodata sources. Another example of map produced by crowd sourcing is **WikiMapia**⁵. The web users can map the different land covers using Google Maps. However, even if the production time and cost are lower than with experts, this map is not accurate everywhere and systematically. Indeed, in some regions, where there are few data, the accuracy can be low: 65.8% in Luxembourg or 60.2% in Lebanon [Zhou et al., 2022].

All these manual methods require either huge number of experts, huge production time, or significant cost and sometimes their accuracies can be limited.

1.3.2. Automatic methods

Automatic methods were developed in order to process satellite image time series with minimal human intervention. Automatic methods are mainly divided into expert-based and data-driven methods. Expert-based methods rely on incorporating the knowledge and the expertise of human analysts into automatic methods. Data-driven methods, also known as machine learning methods, rely on algorithms and statistical techniques to classify land covers.

Expert-based methods

Rule-based classification systems incorporate expert knowledge in the form of if-then rules [Comber et al., 2005]. These rules are derived from the expertise of human analysts and are used to automatically classify land cover based on specific criteria. For example, a decision tree is a set of decision rules which convert the spectral reflectance from each pixel of satellite images into land cover classes [Friedl and Brodley, 1997], [Pal and Mather, 2003]. It is a very simple automatic method which does not make any statistical assumptions of the data distribution [Otukey and Blaschke, 2010]. Instead of rules only based on the spectral reflectance, vegetation indices can be also used, such as the **NDVI** [Lu et al., 2014], [Song, 2019], [Samrat et al., 2022]. An example of decision tree with 6 different land cover classes is represented in Figure 1.13. As shown in the figure, decision trees are interpretable. Indeed, it is easy to see why a pixel has been classified in this specific way. Results from decision trees can be produced quickly. However, it is not always easy for experts to find the decision rules as the number of variables increases. Data-driven methods enable to learn the rules from the data set itself by using for example some optimization algorithms [Hastie et al., 2001].

Data-driven methods

In data-driven methods, a model learns from the input data and then provides predictions on new observations. In classification, the model predicts discrete class labels whereas in regression, the model predicts a continuous quantity. Land cover maps are produced using classification methods. Classification methods are mainly divided in supervised and unsupervised techniques [Halder et al., 2011]. In supervised classification, a set of labeled pixels is used to train the model. Once the model is trained, it is able to classify new pixels not seen

⁴<https://www.openstreetmap.org>

⁵<https://www.wikimapia.org>

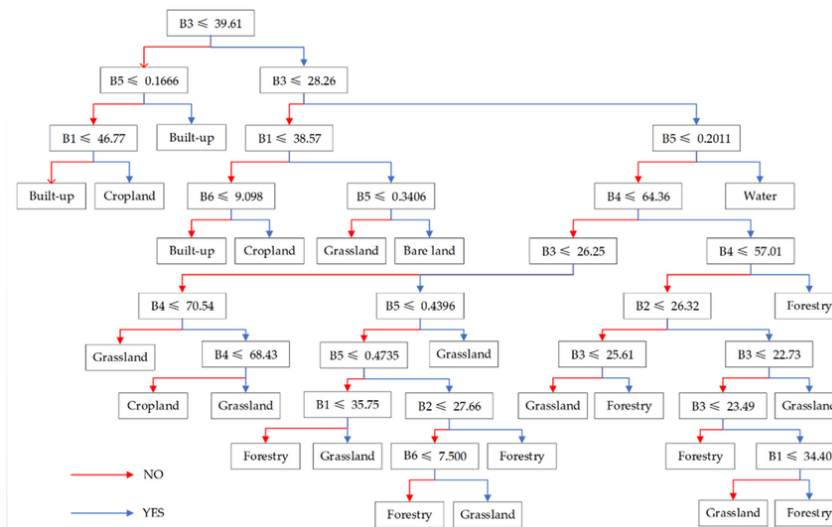


Figure 1.13: Example of decision tree for 6 different land cover classes (bare land, built-up, cropland, forestry, grassland, water). $B1$ to $B4$ correspond to spectral bands 1 to 4 of the Chinese EO satellite HJ-1B, $B5$ corresponds to the NDVI and $B6$ corresponds to the result of ISODATA classification. (Source: [Song, 2019])

during training, as shown in Figure 1.14a. This is currently the most widespread method for land cover classification [Ma et al., 2017]. It can be sometimes difficult to have correctly labeled pixels, and using incorrectly labeled pixels can be a source of error [Gupta and Gupta, 2019], [Pelletier et al., 2017]. Unsupervised classification does not require labeled pixels unlike supervised classification, as shown in Figure 1.14b. Some unsupervised methods were used for land cover classifications, such as clustering methods [Cihlar, 2000], [Franklin and Wulder, 2002], [Grekousis et al., 2015]. However, these methods require human intervention for the specification of the number of classes and also for the assignment of the class labels to the clusters found by the algorithm. Currently, few unsupervised methods have shown satisfactory results and supervised methods are preferred.

Parametric methods, such as **Maximum Likelihood Classification (MLC)** and **Gaussian Mixture Models (GMM)**, were the first supervised classification methods used showing satisfactory results. Indeed, they have shown great potential for various thematic applications [Landgrebe, 2005]. These methods are limited by the shape of the parametric distribution. At the same time, the first neural networks were applied to very small data sets [Benediktsson et al., 1990]. However, due to limited computational resources and the lack of large datasets, these methods could not be developed for several years. In the beginning of 2000, non parametric approaches based on kernel methods, such as **Support Vector Machine (SVM)** were used for land cover [Melgani and Bruzzone, 2004], [Pal, 2009]. Non-parametric methods are generally more robust to outliers in the data compared to parametric methods. Moreover, they do not require assumptions about the distribution of the class-conditional **Probability Density Function (PDF)**. However, **SVM** are computationally expensive and are not adapted to large scale. Simultaneously, **Random Forests (RF)**, another non parametric method was investigated showing very good results [Gislason et al., 2006], [Rodriguez-Galiano et al., 2012b]. Unlike **SVM**, **RF** can handle large amount of data. Moreover, **RF** are very easy to use because of the low sensibility to the parameters that need to be selected by the user. Therefore, **RF** were

largely developed for land cover classification [Shih et al., 2019].

Since 2015, with the increasing number of free and open access data and the increasing storage and computational power, deep neural networks have been largely developed [Ma et al., 2019]. Since then, the number of publications has almost doubled every year [Vali et al., 2020].

Independently of these methods, two main approaches are found: **Pixel-Based (PB)** and **Object-Based (OB)**. Both approaches are represented in Figure 1.15. In the **PB** approach, one land cover class is assigned per pixel. In the **OB** approach, the satellite image is first divided in objects and then these objects are classified [Whiteside et al., 2011]. In the first step of **OB** (i.e. the segmentation step), pixels are aggregated into objects based on homogeneity criteria, either spectral or spatial. In general, this step eliminates the salt and pepper effect associated to **PB** approaches [Blaschke et al., 2000]. Indeed, **OB** approaches are more adapted to higher resolution satellite images than **PB** approaches [Willhauck et al., 2000], [Mansor et al., 2002], [Oruc et al., 2004]. Besides, **OB** approaches have shown equivalent performance results to **PB** approaches for land cover classification with Sentinel-2 time series [Derksen et al., 2020]. However, **OB** approaches require massive computational and storage resources [Whiteside et al., 2011]. The advantages of **OB** approaches are not sufficient in comparison with the computational cost that is added. Thus, **PB** approaches are preferred for large-scale applications. A more complete description of pixel-based supervised methods is provided in the next chapter.

More recently, hybrid methods combining both supervised and unsupervised classification were introduced: semi-supervised and self-supervised classification. In semi-supervised classification, a large part of the training data is not labeled whereas a small part of the training data is labeled. Both data sets are used to train the model, as shown in Figure 1.14c. The unlabeled data is used to improve the performance of the model [Van Engelen and Hoos, 2020]. Very recently, some works were produced in land cover classification for SITS [Zhang and Yang, 2020], [Jing and Chao, 2020], [Lucas et al., 2021]. In self-supervised classification, the most common approach is to pre-train a model with the unlabeled data using a pretext task, as shown in Figure 1.14d. Then, this model is fine-tuned with the labeled data for the target task [Jing and Tian, 2020]. This method was introduced very recently and is currently enjoying great popularity in the remote sensing community [Wang et al., 2022]. Some works using self-supervised methods were produced for land cover classification, very recently [Ren et al., 2021], [Montanaro et al., 2022], [Scheibenreif et al., 2022], [Yuan et al., 2022], [Dumeur et al., 2024]. Yet, neither of these two methods is used for operational production of land cover maps in large scale, as shown in the following section.

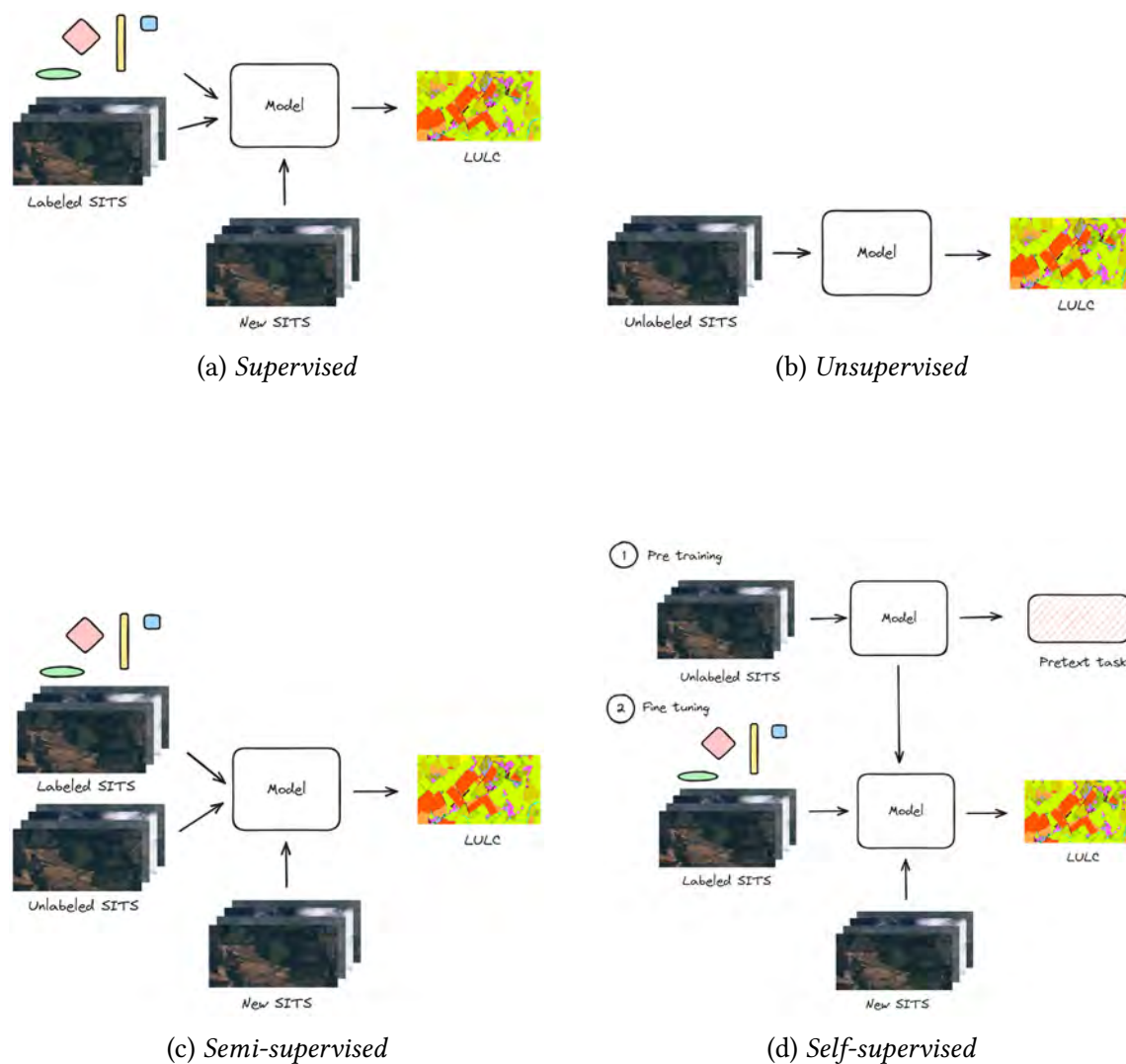


Figure 1.14: Comparison classification methods for land cover classification: supervised, unsupervised, semi-supervised and self-supervised.

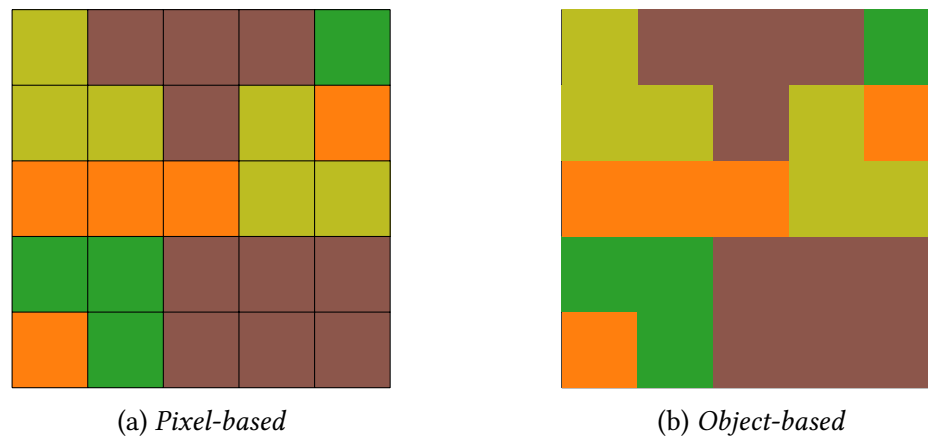


Figure 1.15: Comparison between pixel-based and object-based approaches. Each color represents a different type of land cover. In pixel-based approach, each black square corresponding to a pixel is assigned to a specific class. In object-based approach, the image is first split into several patterns and then each pattern is assigned to a specific class.

1.3.3. Main operational LULC maps

The main maps currently produced operationally are summarized in Table 1.5. They differ by their production method, their scale, the number of classes, the spatial resolution or even by the satellite data used. This table is not exhaustive and we mainly focus on Europe and France.

A large number of LULC maps are still produced with manual methods. At a global scale, there is no LULC map produced with manual methods. At European level, the CORINE Land Cover (CLC) is a land cover map coordinated by the European Environment Agency (EEA). It covers 39 countries over Europe, and for the majority of countries it is produced by photo interpretation [Büttner, 2014]. In some countries, semi-automatic solutions are applied. As shown in Table 1.5, the map is composed of 44 classes which is quite important for a land cover map compared to the others. There are five different versions of the CLC respectively in 1990, 2000, 2006, 2012 and 2018. The different versions do not use the same satellite data, i.e. the first version used Landsat-5 mono date, while the latest version (in 2018) used Sentinel-2 and Landsat-8 time series. This map requires huge production time and is only produced every six years. Also produced with photo interpretation and coordinated by the EEA, the Urban Atlas (UA) corresponds to LULC maps in 800 cities with more than 50 000 inhabitants across Europe. As shown in Table 1.5, it mainly uses high resolution satellite images such as SPOT, Pléiades, SuperView, etc. Its spatial resolution is better than the CLC providing additional information on cities. This map is only produced every six years, at the same time than the CLC. The Land Use/Cover Area frame statistical Survey (LUCAS) is a point field survey at European scale funded by Eurostat⁶, produced every three years since 2006 [d'Andrimont et al., 2020]. As shown in Table 1.5, 8 classes are used for the land covers and 14 classes described the land uses. As shown in Figure 1.16, this digital map is not spatially complete: it is made with reference points [Büttner and Maucha, 2006]. "Crowdsourcing LUCAS" is a survey produced by crowd sourcing as an extension of LUCAS each year when LUCAS is not planned [Laso Bayas et al., 2020]. At national scale, in France, the Occupation des Sols Grande Échelle (OCS GE)

⁶Statistical office of the European Union.



Figure 1.16: Example of one labeled sample (wheat) from the LUCAS survey. (Source: <https://land.copernicus.eu/imagery-in-situ/lucas/lucas-2018>)

is produced with data from different field surveys. The missing informations and also the validation of the field surveys is based on photo-interpretation on orthophoto [IGN, 2022]. As shown in Table 1.5, 14 classes are used for the land covers and 17 classes described the land uses. This map is updated frequently. Also in France, the **TERUTI** study, created in 1982, is based on field surveys but also on photo-interpretation. In 2006, at the creation of the **LUCAS** survey, the **TERUTI** study was fused with the European one given the **TERUTI-LUCAS** study. Recently, all the data collected since 2006, from five **LUCAS** surveys, were harmonized into one database for a total of more than one million of observations [d'Andrimont et al., 2020].

In recent years, there has been an increase in the number of **LULC** maps produced with automatic methods, as shown in Table 1.5. At global scale, there are four main **LULC** maps: **CGLS-LC100** [Buchhorn et al., 2020], **Esri** [Karra et al., 2021], **Worldcover** [Zanaga et al., 2022] and **Dynamic world** [Brown et al., 2022] with around a dozen land cover classes. There is currently only one production available for these products. **CGLS-LC100** map is produced using **RF**. More recently, the **WorldCover** map was released, produced using **Gradient Boosted Decision Tree (GBDT)** [Dorogush et al., 2017]. In 2021 and 2022, respectively, **Esri** and **Dynamic world** were produced using neural networks [ESRI, 2021]. At European scale, in 2021, **CLC+ Backbone** product was released. It provides a detailed European land cover map and is produced using a combination of image segmentation and neural networks [Probeck et al., 2021]. At national scale, in France, the **Occupation des SOIs (OSO)** land cover map is produced every year since 2016 [Inglada et al., 2017] and provides 23 land cover classes at 10m. The **OSO** map is the oldest map produced operationally with automatic methods and more precisely with **RF**. It is also the only map produced every year.

Several methods for producing **LULC** maps with remote sensing were presented. As shown previously, at the beginning of my thesis, pixel-based supervised methods was the state-of-the-art for land cover classification with **SITS** in large scale. Indeed, the **OSO** land cover map was the only operational map produced every year with pixel-based supervised methods. The next chapter will present the different challenges associated to these methods and will detail more precisely the different pixel-based supervised methods used in the literature.

Table 1.5.: Characteristics of different LULC maps.

Name	Scale	Update (year)	Time span	Classes (#)	Spatial resolution (m)	Data	Method
CGIAR-GLC100	World	1	2015-2019	12	100	PROBA-V	Automatic
GLC	Europe	6	1990-2018	44	500	Sentinel-2, Landsat-8	Manual
CLC+ Backbone	Europe	None	None	18	0.5 ha	Sentinel-2	Automatic
Dynamic World	World	2-5 days	2015-pres.	9	10	Sentinel-2	Automatic
Esri	World	1	2017-2022	9	10	Sentinel-2	Automatic
LUCAS (cover)	Europe	3	2006-2018	8		Field survey	Manual
LUCAS (use)	Europe	3	2006-2018	14		Field survey	Manual
OSO	France	1	2009-pres.	18	10	Sentinel-2	Automatic
OCS GE (cover)	France (W+S)	None	None	14	10	Orthophoto	Manual
OCS GE (use)	France (W+S)	None	None	17	10	Orthophoto	Manual
Urban Atlas	Europe	6	2006-2018	27	50 / 100	SPOT, Pleiades, etc.	Manual
WorldCover	World	1	2020-2021	11	10	Sentinel 1-2	Automatic

CHAPTER 2

PIXEL-BASED SUPERVISED LAND COVER CLASSIFICATION USING SITS AT LARGE SCALE

2.1. Challenges	66
2.1.1. Large amounts of data	66
2.1.2. Spatio-spectro-temporal structure	67
2.1.3. Spatial variability	68
2.1.4. Irregular and unaligned SITS	68
2.2. State of the art	70
2.2.1. Machine learning methods	70
2.2.2. Deep learning methods	74
2.2.3. Preprocessing techniques	78
2.3. Remaining challenges and contributions of this thesis	80

In this chapter, the challenges associated to large scale pixel based supervised algorithms are firstly presented. Rather than focusing on improving the quality of the data by using pre-processing techniques, we will focus on developing supervised classification algorithms able to work with raw data. Indeed, noise is present in the SITS (i.e. feature noise) and in the reference data (i.e. label noise) [Pelletier et al., 2017]. The aim is to develop algorithms that are robust to noise present in the features and in particular to missing data due to acquisition conditions. A review of the current algorithms proposed in the literature is presented. Remaining challenges and contributions of this work conclude this section.

2.1. Challenges

2.1.1. Large amounts of data

As mentioned in the previous chapter, there is an ever-increasing amount of available data at global scale. Indeed, for a given year, the volume of data for one satellite can reach several petabytes, as illustrated in Figure 1.12. The volume is defined as the number of pixels times the number of dates times the number of spectral features. By using data from several sensors and satellites (i.e. multi-modality), this volume grows even further and algorithms need to be adapted accordingly.

Wang et al. [Wang et al., 2020] identified three different computational issues for large scale Machine Learning (ML) algorithms:

1. *computational complexity*, which is related to the number of operations (time) and the memory footprint (space). In their paper, the authors refer to the model complexity. For example, some models, such as SVM, defined later, in Section 2.2.1, scale cubically w.r.t. the number of training pixels, making impossible to train with a data set larger than 10^5 pixels.
2. *computational efficiency*, which can be seen as a balance between what you achieve in terms of the objective of the calculation (accuracy) and the cost of the calculation (complexity). For this computational issue, the authors focus on optimization algorithms. Indeed, some recent algorithms allow the use of stochastic methods (e.g. batch gradient descent), which are more suitable than full-search methods.
3. *computational capabilities*, which can refer to the algorithm characteristics that make more efficient use of computing resources. In this paper, the authors take computational parallelism as an example. Indeed, some models are able to perform a large number of computationally intensive operations in parallel.

Even if the algorithm is able to deal with these computational issues, the training and the inference times need to be reasonably fast. For example, the OSO land cover map, described in Section 1.3.3, is produced every year. Taking a large amount of time, e.g. six months, to produce it is not acceptable. Nowadays, the production process requires just a few days. However, this is not the case for most algorithms. Indeed, the majority of algorithms require large learning times for large data sets [Parker, 2012], [L'Heureux et al., 2017]. Besides, even if the algorithm can be parallelized and therefore, can be accelerated, this does not reduce its energy consumption. Rather than being fast, algorithms should be as frugal as possible.

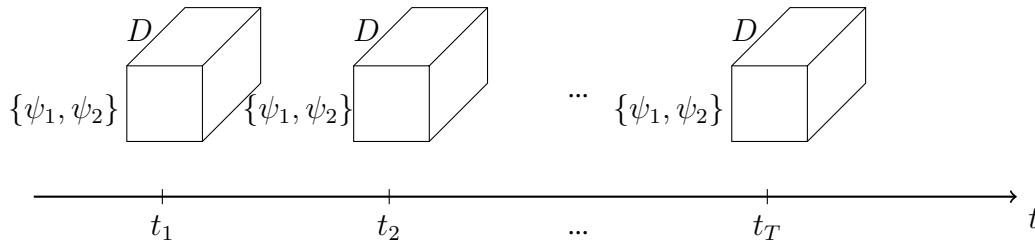


Figure 2.1: Representation of the spatial, spectral and temporal dimensions for one pixel. Each cube represents one pixel at different acquisition times ($\{t_1, \dots, t_T\}$). D represents the number of spectral bands and $\{\psi_1, \psi_2\}$ corresponds to the geographic coordinates of this pixel.

2.1.2. Spatio-spectro-temporal structure

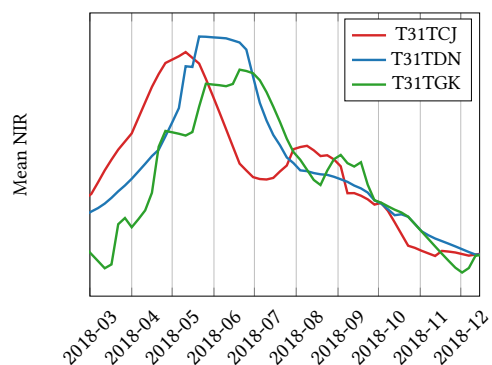
Improved spatial resolution can allow to detect more objects, improved temporal resolution can enable to observe more landscape evolution and finally improved spectral resolution can help to discriminate more materials in the same image. Thus, improved resolutions allow to defined LULC maps with more classes [Mallet and Le Bris, 2020]. One example is the OSO land cover map, which has gone from 17 classes to 23¹: the "winter crop" class was split in three more detailed classes (straw cereals, rapeseed and protein crops). This improvement was made possible because the Landsat-8 data with a spatial resolution of 30m every 16 days was replaced by Sentinel-2 data with a spatial resolution of 10m every 5 days.

However, this increased number of features can also lead to the curse of dimensionality [Russell, 2010]. With an increased number of features, the complexity of our data is also increased. The main problem is that the amount of data required to correctly fit models increases exponentially with the number of features. For the same amount of training data, Hughes showed that above a certain number of features, the performances of the classifier stop growing and start to decline [Hughes, 1968]. It is quite paradoxical, as a higher resolution can help to discriminate more classes, but the complexity of the data can also lead to decrease the accuracy performances.

By using only the spectro-temporal information, it can sometimes be hard to identify some classes which differ only in the spatial pattern. For example, the neighborhood pixels can be used to help identifying a class. Indeed, adjacent pixels are more likely to have similar values than distant pixels [Woodcock et al., 1988].

Besides, the spatial information can be added as an additional dimension. For example, the geographic coordinates can be used as new features [Rußwurm et al., 2023b]. The three dimensions (spatial, spectral and temporal) can be represented separately, as illustrated in Figure 2.1. By giving to the data a structure, we can introduce some prior knowledge. The correlations which depend on different dimensions can be taken into account and can help to reduce the complexity. Indeed, each pixel has a local spatial correlation, as well as a class-dependent spectral and temporal correlation structure that needs to be considered for an accurate classification [Curran and Atkinson, 1998]. The classification algorithm should be able to take into account these correlations and to extract meaningful features that are useful for the classification.

¹<https://www.theia-land.fr/en/product/land-cover-map/>



(a) Mean spectral profiles of winter crops.



(b) Location of the three sites: T31TCJ, T31TDN and T31TGK.

Figure 2.2: Mean spectral profiles of winter crops in three different locations.

2.1.3. Spatial variability

The spectro-temporal signature can be variable over the spatial domain. For example, Figure 2.2 represents the mean spectral profiles of winter crops in three different locations in France. The sites studied are not adjacent and have different meteorological and topographical conditions (plain versus mountain, different longitude and latitude, etc.). Spectral profiles have similar shape but they are not aligned with each other. Therefore, the class conditional probability distribution function is not stationary w.r.t. the spatial covariate.

This non-stationarity problem is not linked to the volume of data but to the wide geographical coverage. On small geographical areas with a large amount of data, this problem does not arise whereas on large area, such as at national scale, non-stationarity is emphasized. The classification algorithm has to be able to model spatially varying class-conditional probability distributions [Higdon et al., 1998], [Paciorek and Schervish, 2006].

2.1.4. Irregular and unaligned SITS

Currently, most conventional classifiers work with a representation of the data as a vector of fixed length. Each pixel is described by a vector where components represent the same type of information, i.e. the value of a given band on a given date, in the same order and in equal quantity. However, in general, SITS are not defined as regular time series of fixed size:

- Firstly, not all pixels are acquired on the same dates. Indeed, working with different sensors, induces different temporal acquisitions. Besides, even if there is only one sensor with a regular revisit cycle, acquisitions dates could be different. Indeed, in large areas, there are different revisit cycles because of satellite orbits (2 or 3 days difference between 2 adjacent swaths).
- Secondly, not all the swaths in an orbit have all the dates. Indeed, if clouds are present in an image, the reflectance corresponds to the one of the clouds and not to the one of the land covers. Therefore, dates for which the image consists essentially of clouds (i.e. above a certain threshold) are removed.
- Finally, locally, different meteorological conditions, such as haze, mist or cloud shadow can cause technical artifacts for one pixel at a given date. This information is considered

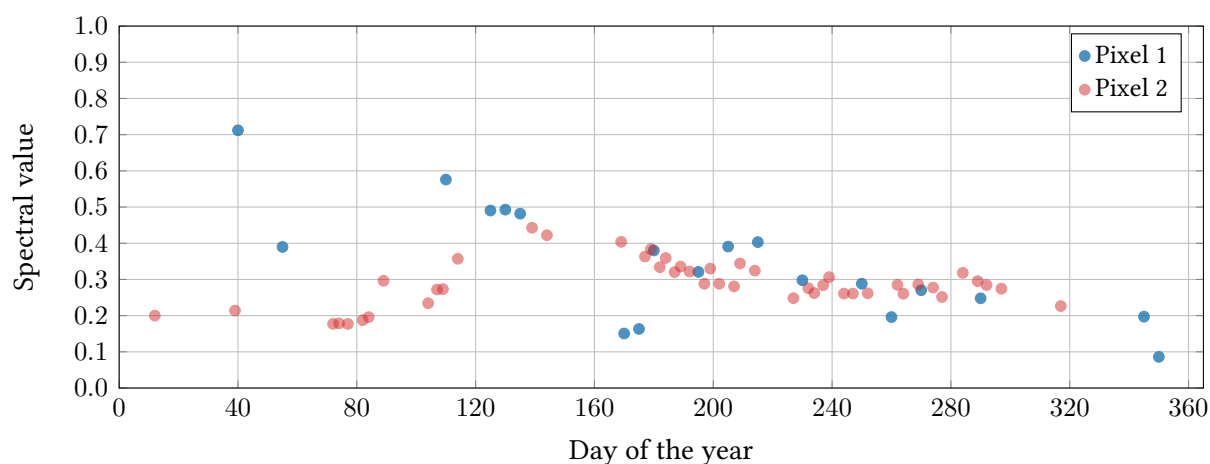


Figure 2.3: Two irregular and unaligned pixel time series acquired by Sentinel-2. Pixels 1 and 2 are respectively from the site T31TDN and T31TGK described in Figure 2.2b. They have different orbits and thus different temporal sampling. Only valid dates have been represented, cloudy dates have been removed. Pixel 1 time series is sized 19 and Pixel 2 time series is sized 46.

as corrupted and the pixel is declared as invalid. Therefore, the information at this date can be removed for this pixel leading to irregular temporal sampling. Therefore, the sampling of very close pixels may also be different.

To sum up, **SITS** can be irregularly sampled in the temporal domain: observations are not equally spaced in time. Moreover, sequences obtained can have different lengths. In addition, **SITS** can be unaligned in the temporal domain: observations are acquired on different dates. As an illustration, Figure 2.3 represents two real irregular and unaligned pixel time series acquired by Sentinel-2 from two different orbits. The classification algorithm should be able to work with unaligned and irregular **SITS**.

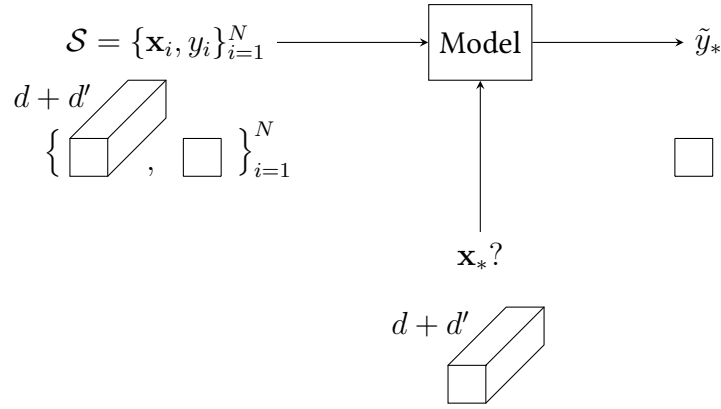


Figure 2.4: Prediction of one pixel \mathbf{x}_* for a model trained with the set of labeled pixels \mathcal{S} . \tilde{y}_* corresponds to the predicted label class.

2.2. State of the art

In the following, we define the i th pixel time series $\mathbf{x}_i(t_k)$ at time t_k by its spectral measurements $[x_i^1(t_k), \dots, x_i^j(t_k), \dots, x_i^D(t_k)]$ with $i \in \{1, \dots, N\}$, N the number of pixels and D the number of spectral features. We suppose a set of T temporal observations such as $t_k \in \{t_1, \dots, t_T\}$. Moreover, $y_i \in \{1, \dots, C\}$ is the target value (i.e. the class label) associated to the pixel \mathbf{x}_i , with C the number of classes. $\mathbf{x}_i = [\mathbf{x}_i(t_1), \dots, \mathbf{x}_i(t_T)] \in \mathbb{R}^d$ corresponds to the raw feature vector which is the concatenation of the spectral measurements for all observations with $d = D \times T$. Finally, the set of labeled pixels is denoted $\mathcal{S} = \{\mathbf{x}_i, y_i\}_{i=1}^N$. In Chapter 5, the spatial information, represented as two spatial coordinates, $\{\psi_{1i}, \psi_{2i}\}$, is associated to the pixel \mathbf{x}_i . Thus, we have: $\mathbf{x}_i = [\mathbf{x}_i(t_1), \dots, \mathbf{x}_i(t_T), \psi_{1i}, \psi_{2i}] \in \mathbb{R}^{d+d'}$ with $d' = 2$.

For each new input \mathbf{x}_* , the model predicts a label class \tilde{y}_* . y_* corresponds to the true label if available. Figure 2.4 represents the prediction of one pixel \mathbf{x}_* for a model trained with the set of labeled pixels \mathcal{S} .

In the following sections, different approaches proposed in the literature for pixel-based supervised methods with satellite image time series in large scale are described. The most widely used classification methods over the last ten years are described.

2.2.1. Machine learning methods

Over the last two decades, ML methods have shown successful results in land cover classification. The literature focuses on non-parametric methods, since they are generally more robust to outliers in the data compared to parametric methods. Moreover, they do not require assumptions about the distribution of data within each class [Mountrakis et al., 2011]. ML methods have very good performance results and are very versatile [Chang and Bai, 2018, Borra et al., 2019].

Support Vector Machines (SVM)

Support Vector Machine (SVM) [Cristianini and Shawe-Taylor, 2000] is a binary classifier that finds an optimal separating hyperplane between two classes with the maximum possible margin, as represented in Figure 2.5. The margin corresponds to the smallest distance between this hyperplane and the data points (support vectors) from both classes. To improve learning capability, the data points can be mapped in a higher-dimensional space where they become more easily linearly separable, as illustrated in Figure 2.6. This is done implicitly with the kernel function. In remote sensing, the most commonly used kernels are the polynomial kernel and the **Radial Basis Function (RBF)** kernel [Maxwell et al., 2018]. The **RBF** kernel can handle nonlinear relationships between the d features and the C classes. More details about kernel functions are given in Chapter 4. This binary classifier is extended to the multi-class case with techniques such as "one against all" (one **SVM** per class) or "one against one" (one **SVM** for each pair of classes) [Hsu and Lin, 2002].

SVM were widely applied in land cover classification with multi-spectral and hyper-spectral images [Camps-Valls et al., 2004], [Melgani and Bruzzone, 2004], [Bazi and Melgani, 2006], [Camps-Valls et al., 2014]. The joint use of spatial and spectral information can improve classification results. A typical example is the use of composite kernels made of disjoint spatial and spectral features for **SVM** hyper-spectral image classification [Fauvel et al., 2012]. Composite spatial and spectral kernels can help to take into account the spatio-spectral structure of the data and, therefore, reduce the spatial variability [Camps-Valls et al., 2006].

In land cover classification, very few works deal with temporal data. Muñoz-Mari *et al.* [Munoz-Mari et al., 2009] proposed a **SVM** with composite kernels showing good performances because the spectro-temporal structure was taken into account. Indeed, the use of composite kernels has increased the performance compared to classical kernels. Other works with also limited number of dates in the year were proposed and with vegetation indices [Devadas et al., 2012], [Kumar et al., 2015], [Zheng et al., 2015]. All these works were implemented with small data sets (i.e. around 500 pixels) as the computational complexity of training process for non linear **SVM** is between $\mathcal{O}(N^2)$ and $\mathcal{O}(N^3)$ [Bottou et al., 2007]. Thus, it becomes quickly intractable as the number of samples N increases. Therefore, **SVM** have rarely been used for large-scale mapping despite their learning capacity. Moreover, **SVM** are not able to deal with sequences with different lengths. Preprocessing techniques are needed to transform these irregular and unaligned time series into regular time series. These techniques are described in Section 2.2.3.

Gaussian Processes (GP)

Gaussian Processes (GP) combine both Bayesian and kernel methods [Rasmussen and Williams, 2005]. Indeed, as a Bayesian method, the prediction is probabilistic, it is thus possible to assess prediction uncertainties. Moreover, kernel functions can be used, such as in **SVM**. Thus, the structure of the data can be taken into account thanks to composite kernels. Their main advantages compared to **SVM** is that their parameters can be learned through gradient descent [Rasmussen and Williams, 2005, Chapter 5]. Moreover, such as **SVM**, **Gaussian Processes (GP)** can be interpretable through their parameters (e.g. temporal correlation for the length-scale parameter in a **RBF** covariance function [Rasmussen and Williams, 2005], [Constantin et al., 2021]). A detailed description of **GP** is provided in Chapter 4, as it is the basis of this PhD work.

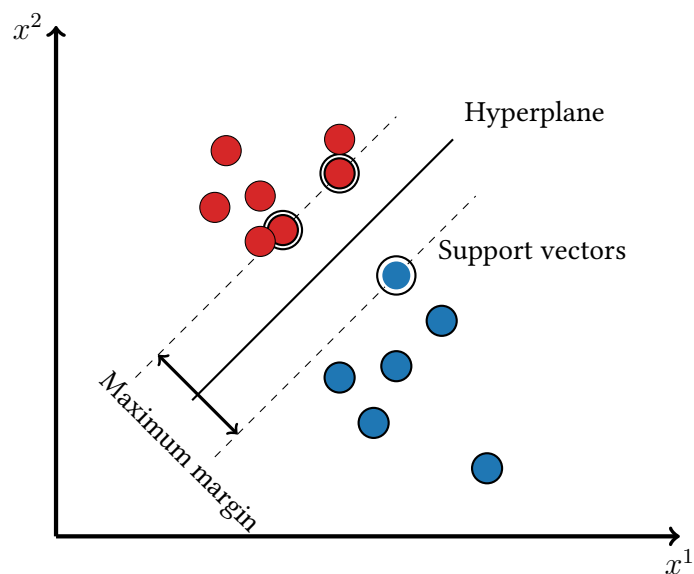


Figure 2.5: Support Vector Machines for two different classes: red and blue points. The classes are linearly separable in 2D with two spectral features x^1 and x^2 . The distance between the two dotted lines corresponds to the maximum margin. The solid line corresponds to the hyperplane. The circled points correspond to the support vectors.

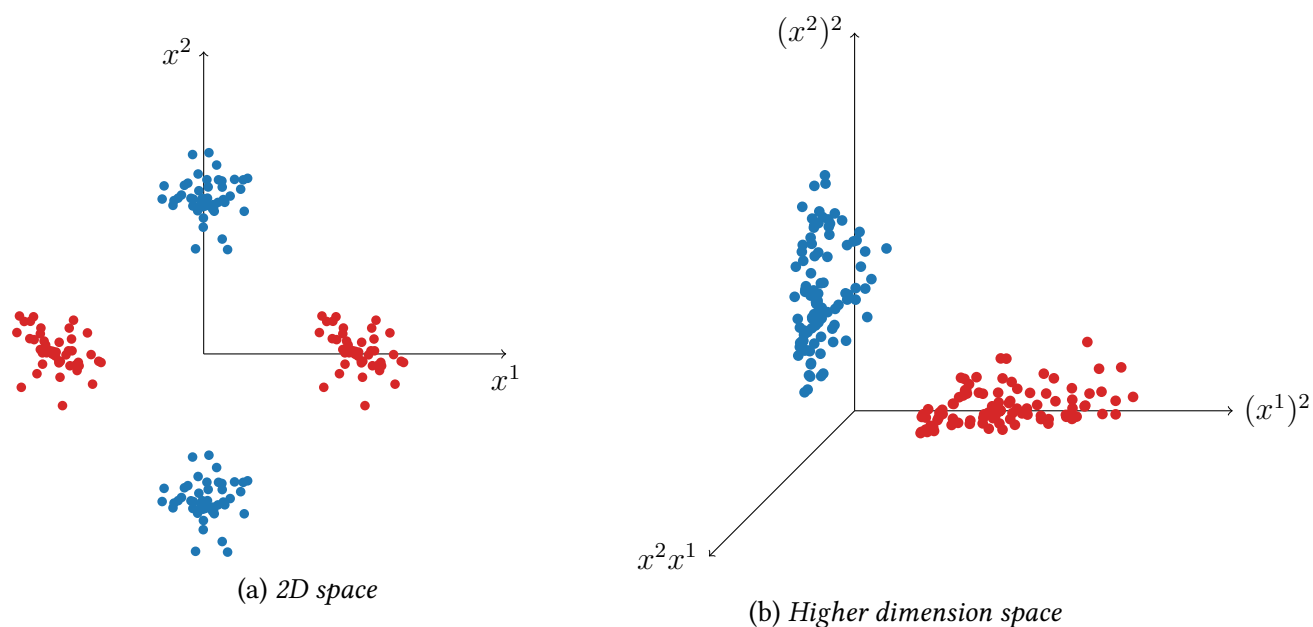


Figure 2.6: In 2D space with two spectral features x^1 and x^2 , the two classes, red and blue points, are not linearly separable. These data points are mapped in a higher-dimensional space by using a polynomial kernel. In higher dimension space, the classes are now linearly separable.

In remote sensing, they have been successfully applied for biophysical parameter estimation (e.g. chlorophyll, **Leaf area index (LAI)**, etc.) [Camps-Valls et al., 2016]. They have also been flourishingly applied for atmospheric parameter retrieval [Camps-Valls et al., 2012], for the surface temperature and moisture or ocean color parameters retrieval [Svendsen et al., 2020] or for the reconstruction of cloud-free time series [Caballero et al., 2023].

Fewer works are found in classification. Indeed, Gaussian processes are more difficult to use in classification than in regression. Classification involves discrete class labels, making the probabilistic nature of Gaussian processes less straightforward to apply. Some approximation approaches can be used to overcome this problem and are detailed in Chapter 4. Moreover, conventional **GP** are limited to few thousands of training inputs since the complexity of their training process is $\mathcal{O}(N^3)$ for regression and $\mathcal{O}(CN^3)$ for classification. For these reasons, classification was mainly applied on small data sets, such as hyper-spectral data sets [Fauvel et al., 2015], [Yang et al., 2015b], [Sun et al., 2015] or multi-spectral data sets [Bazi and Melgani, 2010] with one date.

In recent years, several solutions have been proposed to deal with large amounts of data [Liu et al., 2020]. These methods allow to drastically reduce the computing complexity. Some works dealing with **SITS**, in large scale, for cloud detection [Morales-Alvarez et al., 2018] or land cover classification [Constantin et al., 2021], [Constantin et al., 2022] were proposed. However, these methods are limited either in terms of scale or learning capacity. Very few methods that perform well in computer vision have been developed for remote sensing.

Random Forest (RF)

Another **ML** algorithm widely investigated in the remote sensing literature is **Random Forests (RF)** [Breiman, 2001]. **RF** are composed of multiple decision trees which are learned independently on a bootstrap sample of the training data. Given a new input, the prediction corresponds to the majority vote (classification) or the average of predictions (regression) from these multiple decision trees. **RF** have only few parameters [Biau and Scornet, 2016]: the number of trees, the number of features in each split and the splitting rule [Probst et al., 2019]. Furthermore, unlike **SVM**, they are little sensitive to the parameter values and, therefore, there is no need for cross-validation. Unlike **SVM** and **GP**, **RF** have expanded rapidly because they can handle large amounts of data.

A lot of studies have shown good performances by using **RF** for land cover classification with **SITS** [Adam et al., 2014], [Belgiu and Drăguț, 2016], [Ma et al., 2017], [Camargo et al., 2019]. Moreover, **RF** have been favorably applied to large scale problems [Pelletier et al., 2016], [Inglada et al., 2017], [Leinenkugel et al., 2019]. As shown in the previous chapter, it is the most widely used method for producing operational **LULC** maps.

However, even if **RF** have proved their effectiveness in large-scale land cover classification with **SITS**, they are not capable of identifying new features. For complex problems, users usually need to add handcrafted features in addition to the existing ones in order to emphasize relevant spectral or temporal information. Section 2.2.3 presents the different spectral and temporal handcrafted features used in the literature. **RF** are also not able to take into account the spatial variability of the **SITS** unless specific features or other strategies are used. Pre-processing techniques are used to reduce this variability, they are described in Section 2.2.3. Such as **SVM**, **RF** are not able to deal with sequences with different lengths. They require preprocessing techniques in order to deal with irregular and unaligned **SITS**. Conventional

preprocessing techniques are described in Section 2.2.3.

2.2.2. Deep learning methods

The first LULC map produced with neural networks appeared in the 90s on a very small data set [Benediktsson et al., 1990]. The main advantage of Deep Learning (DL) methods is their ability to extract features (i.e. spatial, spectral and temporal patterns) instead of hand-crafting with preprocessing techniques. Recently, they have experienced a renaissance, due to the increased free distribution of Big Earth Observation Data, the development of computing resources (e.g. GPU, HPC, etc.) and the availability of open source deep learning frameworks (e.g. *Tensorflow* [Abadi et al., 2015] or *Pytorch* [Paszke et al., 2019]). Since then, there has been a resurgence of existing or newly developed DL methods: Multi-layer Perceptron (MLP) [Rumelhart et al., 1986], Convolutional Neural Networks (CNN) [LeCun et al., 1989], Recurrent Neural Networks (RNN) [Hochreiter and Schmidhuber, 1997] and transformer [Vaswani et al., 2017]. Their use for LULC with SITS is reviewed in the following.

Multilayer Perceptron (MLP)

In the early 2000s, Artificial Neural Networks (ANN) showed satisfactory results in the remote sensing community [Hilbert and Ostendorf, 2001], [Kavzoglu, 2009]. The Multi-layer Perceptron (MLP) is a specific ANN composed of one input layer, at least one hidden layer and one output layer with different number of neurons in each layer. Each feature $x^j(t_k)$ corresponds to one neuron in the input layer. The input features are combined with adjustable weights in the hidden layers, producing a feature space. This feature space can capture the spectral and temporal relationships between pixels. The final output layer produces class probabilities in order to assign each pixel to a land cover class. Thus, the number of neurons in the output layer corresponds to the number of classes (C). An example of a MLP with one hidden layer is represented in Figure 2.7.

A loss function is used to measure the error between the predicted and the actual values. The loss minimization is performed using the backpropagation algorithm: first the gradients of the loss function are calculated, then the parameters are updated. This process is repeated until the minimum of the loss function is reached. The model parameters which are initialized randomly are updated during the training to minimize the loss. Backpropagation corresponds to the implementation of gradient descent in MLP. Three different types of gradient descent can be found. Gradient descent (GD) was the earliest method. It is also called batch gradient descent as for each iteration, all the training data are used. Thus, for large-scale data it can be computationally expensive and require large memory. The Stochastic Gradient Descent (SGD) method was developed in order to deal with larger data sets [Robbins and Monro, 1951]. For each iteration, one sample is selected to calculate the gradient which in general leads to noisy results. A compromise between the two methods was proposed called mini-batch stochastic gradient descent. For each iteration, the gradient is calculated with subsets of all observations, and the optimization of MLP can be performed efficiently for large data sets.

In land cover classification, MLP has produced some good results in terms of accuracy [Yuan et al., 2009], [Lavreniuk et al., 2015]. Indeed, MLP is able to extract features that are useful for the classification. In addition, the development of computational resources and deep learning frameworks has helped to cope with huge volumes of data. Like SVM and RF, MLP cannot

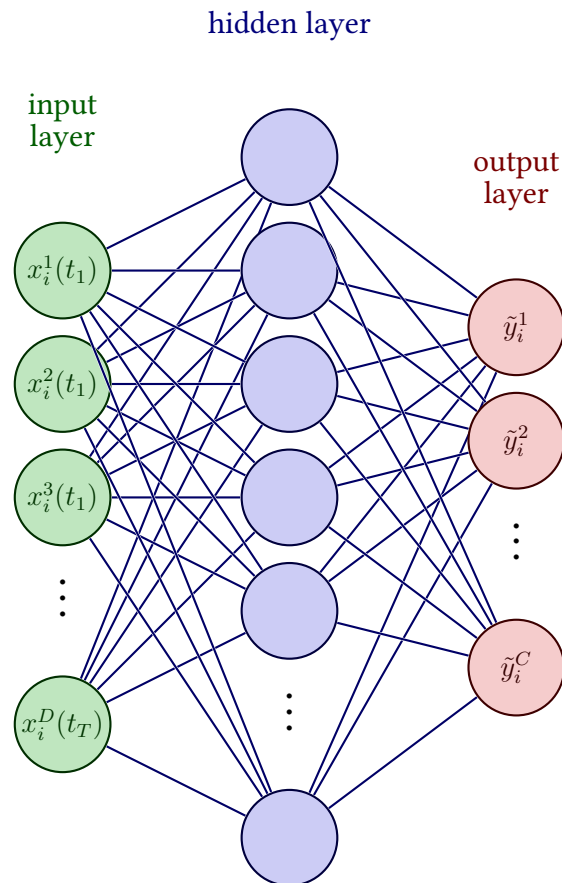


Figure 2.7: Multilayer Perceptron (MLP) with one hidden layer. The input layer is composed of d neurons for one pixel x_i . The output layer is represented by C neurons. Each value of the output layer is processed through a softmax function. The label class y_i for the pixel x_i correspond to $\text{argmax}(\text{softmax}(\tilde{y}))$ with $\tilde{y} = \{\tilde{y}^1, \dots, \tilde{y}^C\}$.

deal with sequence with different lengths as **MLP** concatenates pixels into a vector of fixed size. Irregular and unaligned time series cannot be processed by **MLP**. Moreover, the spatial arrangement of pixels as well as the spatial variability are not taken into account.

Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) were developed in order to take into account the spatial structure of the image. They were firstly developed for image classification. The earliest **CNN** architecture was LeNet [Lecun et al., 1998] developed in 1998, followed by AlexNet [Krizhevsky et al., 2012] in 2012. A pixel is processed by considering the pixels around it in a window. These pixels constitute a patch corresponding to the input. The hidden layers include one or more convolution and pooling layers. Convolution layers are used to learn local patterns and features from the image. Pooling layers reduce the size of the image while retaining the most important information. By combining a large number of layers, more abstract and higher-level features are extracted. After the hidden layers, fully connected layers are used to perform the classification [Li et al., 2022]. Later, **Fully Convolutional Network (FCN)** [Long et al., 2015] were developed for semantic segmentation. Instead of classifying

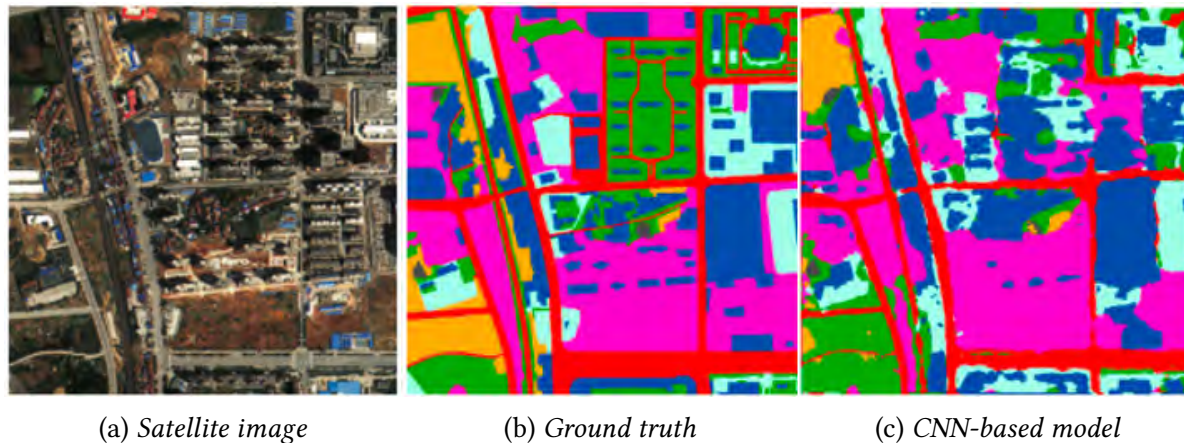


Figure 2.8: Comparison between satellite image from Beijing-2, ground truth and land cover maps obtained with a CNN-based model. There is a lack in the geometry, predictions are too smooth in the straight borders. (Source: [Zhang et al., 2019])

the whole image, each pixel in an image is assigned to a specific class i.e. pixel-based classification. Therefore, there is no fully connected layers. Since then, other CNN architectures were developed in computer vision such as: U-Net [Ronneberger et al., 2015], ResNet [He et al., 2016], SegNet [Badrinarayanan et al., 2017], etc.

In land cover classification, CNN models, with different architectures, have shown very good results in terms of accuracy [Kussul et al., 2017], [Stoian et al., 2019a]. They outperform widely used ML methods, such as RF and SVM [Carranza-García et al., 2019]. Thanks to their convolutional layers, the spatial structure of the SITS is taken into account. However, the main structures, such as crop field borders, roads or buildings, can be not clearly represented and be too smooth (i.e. rounded), as illustrated in Figure 2.8. Indeed, they rely heavily on texture to extract spatial feature [Geirhos et al., 2019]. An adaption of the U-Net was proposed to overcome this problem however it was computationally expensive [Stoian et al., 2019b]. The authors of [Yao et al., 2019] proposed to add a coordinate convolution module [Liu et al., 2018c] into a DenseNet in order to strengthen object boundaries. Spatial information was added by putting coordinate information into feature maps. This network outperformed CNN architectures such as U-Net. By adding coordinate information, performance results with CNN have been improved [Yao et al., 2019]. The computational cost was rather high.

However, a large proportion of CNN works are applied on single dates. Moreover, none of these techniques takes account of the temporal aspect of SITS. To deal with the temporal aspect, Temporal CNN or 1D-CNN was proposed [Pelletier et al., 2019]: convolutions are applied along the temporal dimension. This method shows good performances in terms of accuracy but they need regular aligned SITS.

Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNN) were defined for sequential data [Medsker and Jain, 2001]. RNN can capture the dynamics of sequential data thanks to recurrent connections. For each new input, RNN uses the previous inputs to predict the output. Unlike all the methods described above (SVM, RF, MLP, CNN), RNN are able to deal with inputs of variable length sequences. Thus, RNN can take into account irregular time series. However, RNN assume

the same acquisition times. Thus, they do not support unaligned time series. Moreover, as **RNN** process the data sequentially, they are slow to train because of the lack of parallelization abilities. Instead of using the backpropagation algorithm to optimize the parameters such as **MLP** or **CNN**, **RNN** use **Backpropagation through time (BPTT)** algorithm. Like other techniques based on backpropagation, it leads to gradient problems (i.e. vanishing and exploding), but they are exacerbated. For vanishing gradient, the gradient becomes too small, thus the parameters are not updated and the algorithm generally stops learning. For exploding gradient, the gradient becomes too large, thus the model has numerical issues and it is difficult to have good performances. By using different **RNN** architectures, such as **Gated Recurrent Unit (GRU)** [Chung et al., 2014] or **Long Short Term Memory (LSTM)** [Neil et al., 2016], the vanishing gradient problem has been limited.

These architectures were successfully applied in land cover classification [Rußwurm and Körner, 2017], [Ienco et al., 2017], [Sharma et al., 2018], [Rußwurm and Körner, 2018b]. However, **RNN** do not take into account the spatial information. To include spatial information, different methods were proposed for hyper-spectral images. In the spatial-sequential **RNN** [Zhang et al., 2018], handcrafted features based on texture and differential morphological profiles are used in addition to the features. In the spectral-spatial **RNN** [Liu et al., 2018a], neighborhood pixels are used. By combining **RNN** and **CNN**, spatial and temporal features can be taken into account together. Conv-LSTM [Rußwurm and Körner, 2018a] which combines convolutional operations with **LSTM** has shown good results for cloud segmentation. In land cover classification with **SITS**, DupLO [Interdonato et al., 2019] also combines convolutional and recurrent neural networks and has shown good accuracy results. Rustowicz et al. [M Rustowicz et al., 2019] proposed a model called U-Net + ConvLSTM: a U-Net [Ronneberger et al., 2015], used as a spatial encoder, is followed by a temporal encoder, ConvLSTM. In land cover classification with **SITS**, Chamorro Martinez et al. [Chamorro-Martinez et al., 2021] proposed to use a **FCN** as a spatial encoder, followed by a temporal encoder, ConvLSTM.

The transformer architecture

CNN require a large number of layers (i.e. very deep network) in order to capture dependencies globally. **RNN** mainly capture dependencies locally, between close elements of a sequence. The transformer architecture [Vaswani et al., 2017], initially used for sequential data in **Natural Language Processing (NLP)**, can catch long range dependencies, more globally, thanks to its attention mechanism [Wen et al., 2023]. Transformers are able to process sequences in parallel which was not the case for **RNN**. Attention mechanisms are permutation invariant, i.e. the model remains the same even if the order of its inputs is altered [Cai et al., 2023]. Therefore, positional encoding is used in transformers in order to take into account the order [Vaswani et al., 2017]. Besides, transformers are able to deal with irregular time series thanks to the temporal positional encoding and padding, as explained in Chapter 7.

In 2020, Rußwurm and Körner [Rußwurm and Körner, 2020] pioneered the use of self-attention for land cover mapping using irregular Sentinel-2 **SITS**. They obtained the best results compared with **RNN** and **CNN**. It was rapidly followed by a modified version of the transformer [Garnot et al., 2020], also outperforming **CNN** and **RNN** networks. A **Temporal Attention Encoder (TAE)** is used to encode each pixel time series into a single embedding, by using attention mechanism. In order to avoid unnecessary computations and parameters, a lighter version of this network, called **Lightweight Temporal Attention Encoder (LTAE)**, was

developed [Garnot and Landrieu, 2020]. The method outperforms most of state-of-the-art time series classification algorithms. However, the LTAE was only applied to irregular time series in large scale classification and not to unaligned time series (the data set used in [Garnot et al., 2020] was produced on only one tile). Recently, the authors proposed a version combining a U-Net [Ronneberger et al., 2015] with the LTAE called U-TAE [Garnot and Landrieu, 2021]. Other ways of using attention mechanisms were proposed, with for example an attention-based model using Thermal Positional Encoding (TPE) instead of classical positional encoding [Nyborg et al., 2022]. This TPE uses the accumulated degree days instead of the day of the year. More recently, a model called TSViT based on the vision transformer, ViT [Dosovitskiy et al., 2021], was proposed for SITS classification. ViT works with patches, such as CNN, whereas Transformer architecture works with sequences of pixels. Compare to CNN, ViT offer advantages in scenarios where there are global dependencies. However, it requires much more large data sets than CNN. TSViT outperforms the U-TAE on three different data sets [Tarasiou et al., 2023].

However, Kondmann *et al.* [Kondmann et al., 2021] points out that extracting representative features from SITS is not a trivial task. Indeed, despite the complexity of the LTAE, it only slightly exceeds the performance of a Random Forest with handcrafted spectral features [Kondmann et al., 2021]. Moreover, models based on the attention mechanism have a quadratic complexity with respect to the input size.

2.2.3. Preprocessing techniques

In the following, preprocessing techniques (temporal re-sampling, handcrafted features and spatial stratification) used by some classifiers in order to improve the classification performances are presented. Some classifiers require temporal resampling as they are not able to deal with irregular and unaligned time series. Moreover, some of them use handcrafted features in order to extract meaningful information. Finally, spatial stratification can be performed for some classifiers in order to reduce the spatial variability.

Temporal re-sampling

Some classifiers are not able to deal with input sequences with different lengths, such as SVM, RF, MLP or CNN. Preprocessing techniques are used to transform irregular and unaligned time series into regular time series.

Selecting multiple dates based on different criteria is one of the techniques. A variety of criteria can be defined: selecting dates on two different seasons [Rodriguez-Galiano et al., 2012b], selecting optimal dates based on feature importance [Nitze et al., 2015], selecting dates with the fewer clouds [Jin et al., 2018], [Solano-Correa et al., 2022], selecting dates corresponding to growth cycle [Nguyen et al., 2020], etc. However, the number of dates is often limited and small. Thus, some information may be lost. Moreover, it can not be generalized on a large scale. Indeed, it does not work with time series from two different orbits.

Temporal aggregation is another preprocessing technique used to deal with irregular and unaligned time series. Temporal aggregation transforms a time series with high frequency (every few days) to low frequency (every month, every quarter). Statistics from the aggregated data (i.e mean, median, min, max, etc.) are used as inputs for the classifier [Xie et al., 2019], [Phan et al., 2020]. However, Carrasco *et al.* [Carrasco et al., 2019] showed that temporal

aggregation does not lead to better classification accuracies than two dates well chosen. Besides, some methods produce composition of time series data from all the cloud free available images [Griffiths et al., 2013], [Zhu and Woodcock, 2014], [Hermosilla et al., 2018]. However, if some areas are very cloudy, they may have very little or even no date leading to poor classification.

Interpolation is another method used to transform irregular and unaligned time series into regular time series. It is widely used in many works [Inglada et al., 2017], [Kondmann et al., 2021] and has shown good performances in terms of accuracy.

Instead of using preprocessing techniques and conventional classifiers, some works proposed to use the **Dynamic Time Warping (DTW)**. It was first introduced in speech recognition in 1968 [Vintsyuk, 1968] to measure the similarity between two temporally shifted time series with different lengths. It identifies the best alignment between them by allowing a non-linear mapping of one time series to another, and minimizing the distance between these two time series. Petitjean *et al.* [Petitjean et al., 2012] proposed to use **DTW** for land cover classification with Formosat-2 image time series. Dusseux *et al.* [Dusseux et al., 2013] also used **DTW** for grassland classification using biophysical variable temporal profiles derived from Landsat and SPOT image time series. **DTW** allows to find the best alignment between two time series, however it does not include information on inter and intra-annual phenological cycles [Maus et al., 2019]. Thus, the **Time-Weighted Dynamic Time Warping (TWDTW)** was proposed by introducing time weight factor, as an extension of the **DTW** [Maus et al., 2016]. Later, the Spatial Parallel **TWDTW** allowed to parallelize the **TWDTW** algorithm and to take into account the spatial dimension [de Oliveira et al., 2019]. Even if it achieved almost linear speed up, it was not able to deal with very large data-sets. Thus, there is no work applied to large scale land cover classification using **TWDTW**.

Handcrafted features

Some **ML** classifiers, such as the **RF**, are not able extract features in contrast to deep learning methods.

In order to take into account the spectral structure of the data, it is common to add additional spectral features. Indeed, handcrafted features can help the classifier to learn the decision rule. Different spectral features can be defined, such as vegetation indices, i.e. **NDVI**, **Enhanced Vegetation Index (EVI)**, **Soil Adjusted Vegetation Index (SAVI)**, brightness, greenness, wetness, etc. [Nitze et al., 2015], [Valero et al., 2016], [Inglada et al., 2017], [Thonfeld et al., 2020]. Other spectral features can also be defined such as the color, water index (i.e. **Normalized Difference Water Index (NDWI)**), built-up index (i.e. **Normalized Difference Built-up Index (NDBI)**), etc.

Spatial handcrafted features can also be added in order to take into account the spatial structure of the data and thus reduce the spatial variability. Morphological features such as the texture [Haralick, 1979] provide some information about the visual characteristics of the image: roughness, smoothness, regularity, symmetry, etc. In land cover classification, some works were proposed [Ghimire et al., 2010], [Rodriguez-Galiano et al., 2012a], [Jin et al., 2018]. Topography informations such as the elevation, the slope or the aspect can also be added to the features [Le et al., 2022]. Finally, geographic coordinates, such as the longitude and latitude, can also be an additional source of information [Yang and Huang, 2021]. However, all these spatial features may become lost amongst the large amount of information available. Indeed, adding the spatial features among all the handcrafted and spectral features may have

a negligible impact.

Finally, temporal information can help to improve the performances of classifiers, especially in order to discriminate the different crops [Vuolo et al., 2018]. However, several classifiers, such as RF or MLP, are not able to take into account the temporal order of the SITS. For example, we train two RF models: one with a time series and another one with the same time series but with the features in a different order. It leads to two different models i.e. the weights are different because the features are not in the same place. However, their performances are very similar. Therefore, switching the temporal order of the time series leads to the same results. In order to take into account the temporal structure of the data, temporal features can be added. They can correspond to the statistical values (i.e. mean, median, min, max, etc.) extracted from the spectral values, the handcrafted spectral features or even the spatial handcrafted features. They can also be the key dates of the phenological cycle for some vegetation classes (i.e. sowing, threshing, cropping). However, Pelletier *et al.* showed that the addition of temporal features has little effect on classification performance with RF [Pelletier et al., 2016].

Spatial stratification

Spatial stratification corresponds to divisions in spatial domain. The area is divided into strata. In each stratum, a classification model is learned. Inglada *et al.* [Inglada et al., 2017] proposed to stratify France into eco-climatic regions, as defined in [Joly et al., 2010]. For each stratum, an independent RF model was trained. The non-stationarity of the data was handled by this spatial stratification. Indeed, each pixel inside the eco-climatic region has the same topographic and meteorological requirements. Several works using spatial stratification have also been shown to improve the classification accuracy [Cano et al., 2017], [Moraes et al., 2021], [Costa et al., 2022]. Costa *et al.* [Costa et al., 2022] produce a land cover map of Portugal with Sentinel-2 time series based on RF classifier and spatial stratification.

2.3. Remaining challenges and contributions of this thesis

Section 2.1 introduces the challenges associated with large-scale land cover classification and Section 2.2 presents the state-of-the-art. However, there are still challenges ahead, and in this thesis we propose to help address them.

Firstly, in large scale land cover classification, we need to use a model able to deal with large volumes of data. Methods are highly dependent on the number of training inputs N and the number of features d . At the beginning of the thesis, RF was the only method used operationally for land cover classification in large scale. Since then, neural networks have been the focus of much development, and GP have been on the sidelines, as illustrated in Figure 2.9. However, GP are a very promising tool thanks to the Bayesian framework. For example, they can provide the full posterior distribution. Moreover, uncertainty is not only incorporated in the inference stage but also in the training stage. Indeed, uncertainties are considered for the model parameters. This probabilistic aspect helps to reduce over-fitting. Recently, Bayesian neural networks (BNN) [Jospin et al., 2022] were developed in order to add this Bayesian framework to neural networks. However, like classical (non Bayesian) neural networks, they lack interpretability due to the large number of layers and parameters. The



Figure 2.9: *Even if Gaussian Processes are not fashionable, they have many interesting properties. In this thesis, we decided to explore this possibility for land cover classification in large scale. (Image credit: Kai Arulkumaran)*

main bottleneck of conventional **GP** is their complexity. However, recently, approximation methods were proposed in computer vision to deal with the large amounts of data. In this work, we propose to use approximate **GP** in order to perform land cover classification in large scale. This contribution is developed in Part II.

Secondly, the spatio-spectro-temporal structure of the data should be taken into account as well as the correlations between features. Some deep learning methods are specialized for modeling the temporal structure of the data, such as **RNN** or the spatial structure, such as **CNN**. Nowadays, hybrid methods are able to extract the spatio-temporal structure. However, as described previously, deep learning methods are hardly interpretable. On the other hand, **GP** are able to take into account the spatio-spectro-temporal structure of the data thanks to composite kernels. In this work, we propose to use composite kernels with the **GP** defined in large scale. This contribution is developed in Part II. Moreover, a structured spectro-temporal reduction is proposed in Part III in order to better take into account the structure of the **SITS** and to reduce the complexity of the **GP** model.

Thirdly, we need methods able to deal with the non-stationarity of the data. Spatial stratification is currently one of the methods most widely used operationally. However, no spatial constraints are imposed during the learning or the prediction steps and the models can behave differently at the boundaries between strata. Thus, the transition between two spatial strata can show artifacts due to the discontinuity in the predictions by models of adjacent strata. Discontinuities in prediction for **RF** models (**OSO** approach) between two eco-climatic regions are illustrated in Figure 2.10. We propose to study the impact of the spatial stratification and use composite kernels in order to reduce the spatial variability. This contribution is

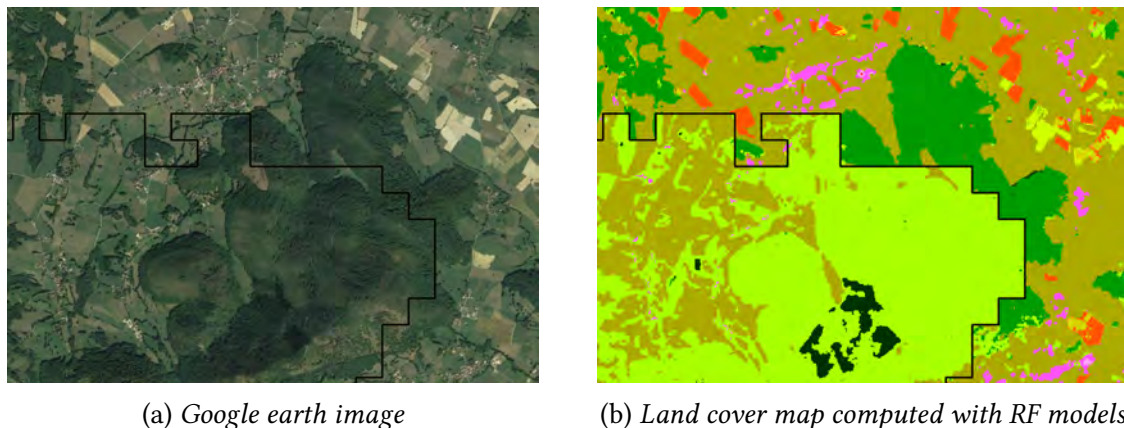


Figure 2.10: *Spatial discontinuity in land cover classification computed with RF models between two eco-climatic regions. The black line is the separation between regions. The color nomenclature is described in Table 3.3.*

developed in Part II. In Part III, another approach is proposed to add the spatial information based on a spatial informed kernel.

Finally, most of the algorithms used operationally work with regular time series of the same length. Preprocessing techniques were developed in order to transform irregular and unaligned time series into regular time series that can be used by the classifier. However, most of the techniques developed, such as linear interpolation, are performed independently w.r.t. the classification task. Therefore, relevant information for the classification task can be lost when producing these re-sampled observations. Indeed, Li *et al.* [Li and Marlin, 2016] showed that an independent interpolation method directly followed by a classification method performed worse than methods trained end-to-end. Finally, by performing end-to-end learning by combining a time and spatial informed kernel interpolator with the GP classifier defined in Part II, irregular and unaligned data can be processed. This contribution is proposed in Part III.

CHAPTER 3

DESCRIPTION OF THE DATA USED

3.1. Sentinel-2 image time series	84
3.1.1. Description	84
3.1.2. Products	84
3.1.3. Study area	87
3.2. Data preparation for Sentinel-2 SITS	88
3.2.1. Radiometric and geometric corrections	88
3.2.2. Spatial resampling	88
3.2.3. Feature extraction	88
3.2.4. Spatial information extraction	89
3.2.5. Temporal resampling	89
3.3. Reference data	96
3.3.1. Sources	96
3.3.2. Polygons	100
3.3.3. Eco-climatic regions	101
3.4. Data set selection	103
3.4.1. Polygon selection	103
3.4.2. Pixel selection	103

Before describing the different methods used and the associated results, this section presents the data used in Part II and Part III.

3.1. Sentinel-2 image time series

3.1.1. Description

Sentinel-2 is a program from the Copernicus Programme operated by [European Space Agency \(ESA\)](#) [[Bertini et al., 2012](#)]. The first satellite, Sentinel-2A, was launched in June 2015 followed by the second one, Sentinel-2B, in March 2017. Sentinel-2A was planned to run until 2022 but it is still working. A third satellite, Sentinel-2C, is planned for 2024, followed by a fourth one Sentinel-2D, in 2025.

With an orbit at around 785 km, the two satellites provide free and open data every 5 days at the equator and every 2-3 days at high latitudes. Each of them is composed of a [Multi-Spectral Instrument \(MSI\)](#) with a 290 km-wide coverage. [Table 3.1](#) represents the description of the 13 bands with their respective wavelengths and spatial resolutions. The data has high spectral and spatial resolutions (four spectral bands at 10 m, six spectral bands at 20 m, and three spectral bands at 60 m per pixel). B1 is used to measure the optical thickness of the atmosphere. B2, B3 and B4 are respectively, blue, green and red bands in the visible spectrum. They are useful for characterizing vegetation, urban areas and also water. B5, B6, B7, B8 and B8A are mainly used to identify vegetation. B9 allows the detection of the water vapour and B10, the cirrus clouds (wispy clouds). Finally, B11 and B12 enable the measure of soil moisture and vegetation characteristics and is also useful for the differentiation between snow and clouds.

Its low cost and its beneficial characteristics (frequent revisit, high spatial and spectral resolutions) make Sentinel-2 data widely used in remote sensing applications. [Figure 3.1](#) represents the number of articles mentioning "Sentinel-2" in the main remote sensing journals: [TGRS](#) (Transactions on Geoscience and Remote Sensing), [JSTARS](#) (Journal of Selected Topics in Applied Earth Observations), [RSE](#) (Remote Sensing of Environment) and [International Society for Photogrammetry and Remote Sensing \(ISPRS\)](#) Journal of Photogrammetry and Remote Sensing from 2013 to today. Since 2018, one year after the launch of the second satellite, Sentinel-2B, this number has risen sharply. Moreover, Sentinel-2 image time series are particularly well adapted for [LULC](#) classification. As shown in [Figure 3.1](#), the number of articles mentioning both "Sentinel-2" and "land cover" has also increased over the years.

3.1.2. Products

The mission generates products with different levels (0, 1A, 1B, 1C, 2A, 3A), as described in [Table 3.2](#). Some levels (0, 1A, 1B) are not released to users or only to expert users. They correspond to sub-images of a detector. From level 1C, images are released to users. They are reprojected in a cartographic reference frame. Indeed, these levels (1C, 2A, 3A), represented in [Figure 3.2](#), provide images in an area of size 110 km by 110 km, called a tile.

Table 3.1.: Description of the Sentinel-2 bands.

Band	Wavelengths (nm)	Spatial resolution (m)
B1 – Coastal aerosol	421-463	60
B2 – Blue	426-558	10
B3 – Green	523-595	10
B4 – Red	633-695	10
B5 – Vegetation red edge	689-719	20
B6 – Vegetation red edge	725-755	20
B7 – Vegetation red edge	762-802	20
B8 – NIR	726-938	10
B8A – Narrow NIR	843-885	20
B9 – Water vapour	925-965	60
B10 – SWIR – Cirrus	1342-1404	60
B11 – SWIR	1522-1704	20
B12 – SWIR	2027-2377	20

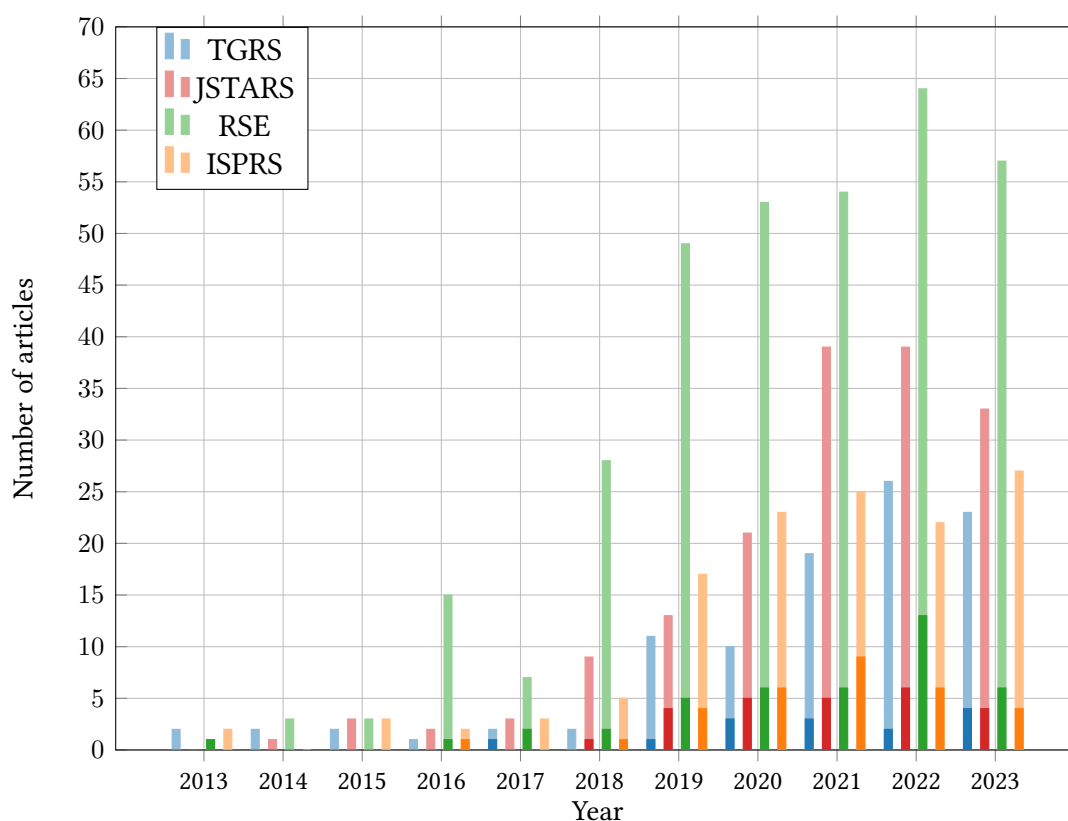


Figure 3.1: Transparent and filled bars correspond to the number of articles, respectively, mentioning "Sentinel-2", or "Sentinel-2" AND "land cover", in the abstract or title in several journals. TGRS (Transactions on Geoscience and Remote Sensing) and JSTARS (Journal of Selected Topics in Applied Earth Observations) are journals published by Institute of Electrical and Electronics Engineers (IEEE). RSE (Remote Sensing of Environment) and ISPRS Journal of Photogrammetry and Remote Sensing are published by Elsevier. The year 2023 is not complete and correspond to articles from January to the end of August 2023. (Source: <https://app.dimensions.ai/discover/publication>).

Table 3.2.: Description of the Sentinel-2 products.

Level	Description	User's type	Scale (km ²)
0	Compressed raw image data	Not release to users	23x25
1A	Decompressed raw image data of level 0	Not release to users	23x25
1B	Top-of-atmosphere radiances	Expert users	23x25
1C	Top-of-atmosphere reflectances	All users	110x110
2A	Atmospherically corrected surface reflectances with cloud/shadow mask	All users	110x110
3A	Monthly synthesis from images of level 2A	All users	110x110

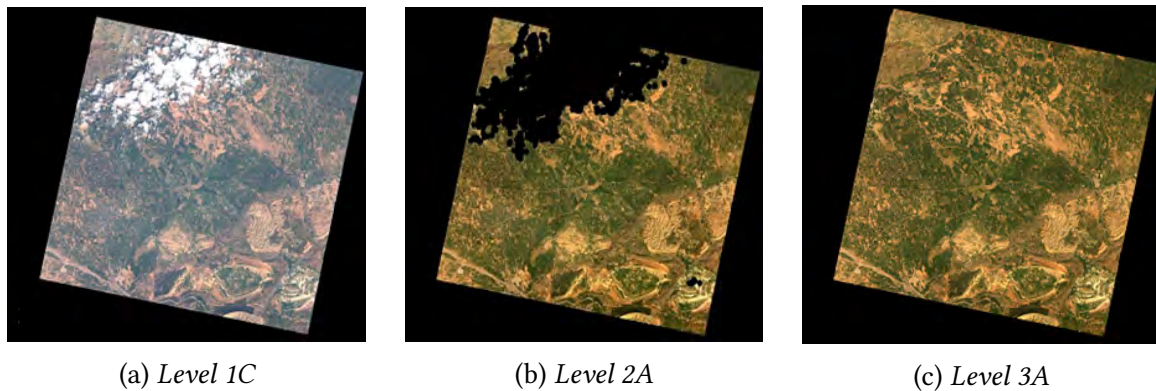


Figure 3.2: Products of level 1C, 2A and 3A. The image corresponds to a Formosat-2 image of about 24 km by 24 km (16 March 2006). (Source: <https://labo.obs-mip.fr/multitemp/theias-l3a-product-format/>)

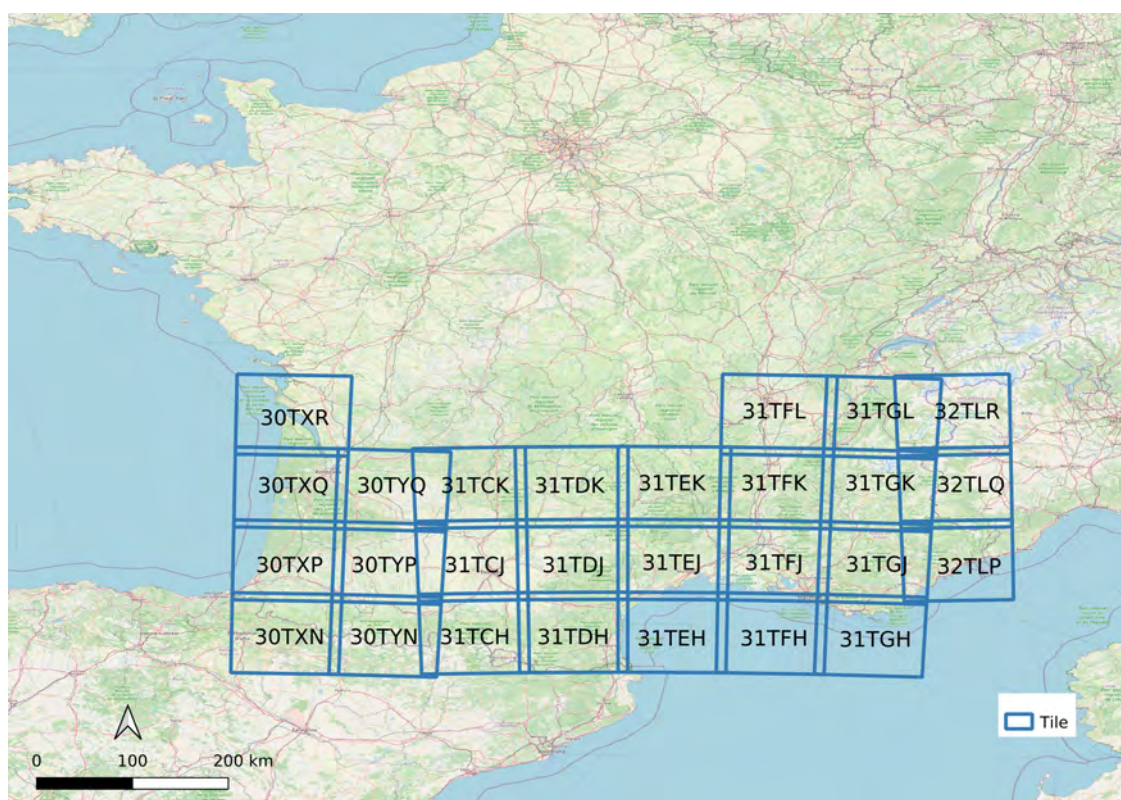


Figure 3.3: Location of the 27 studied tiles where a blue square corresponds to one tile as provided by the Theia Data Center. Each tile is displayed with its name in the Military Grid Reference System (MGRS) nomenclature used for Sentinel-2 products (background map © OpenStreetMap contributors).

3.1.3. Study area

The study area is located in the south of metropolitan France and it covers an area of approximately 200 000 km² corresponding to around two billion pixels. It is composed of 27 Sentinel-2 tiles as illustrated in Figure 3.3. This area provides a wide variety of landscapes. Coastal areas are found on the shores of the Atlantic Ocean and the Mediterranean Sea. The Pyrenees, the Massif Central and the Alps form a vast range of mountainous areas. Many cities, including some of the most densely populated in France, are part of this zone: Marseille, Lyon, Toulouse, Nice, Montpellier, Bordeaux, Toulon, Saint-Etienne, Grenoble, etc. (order according to number of inhabitants). Finally, a large diversity of rural areas ranging from intensive agricultural areas to forests can also be found. All Sentinel-2 tiles were downloaded from the Theia Data Center¹. Different corrections were applied, as described in the following sections.

¹<https://www.theia-land.fr/en/products/>

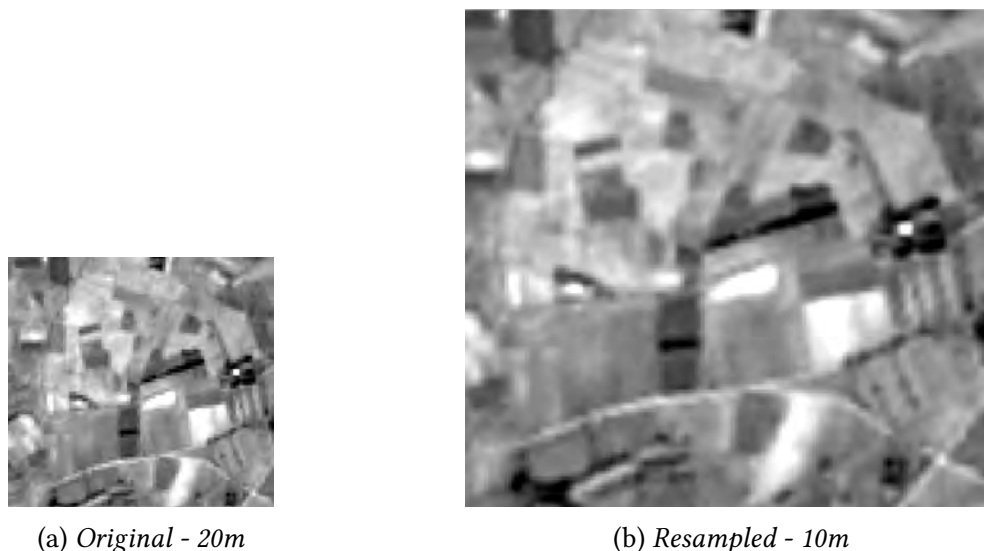


Figure 3.4: Example of an image from the band B5 spatially resampled from 20m to 10m with bicubic interpolation. (Source: https://ecampus.paris-saclay.fr/pluginfile.php/1011133/mod_resource/content/0/Copernicus_Agri_Ocsol_TP.pdf)

3.2. Data preparation for Sentinel-2 SITS

3.2.1. Radiometric and geometric corrections

In this work, the Sentinel-2 products of level 2A are provided by the **MACCS-ATCOR Joint Algorithm (MAJA)** processing chain [Baetens et al., 2019]. MAJA is able to detect clouds and their shadows, to estimate the atmospheric content in aerosols and water vapor and to correct the atmospheric, adjacency and slope effects. It provides the surface reflectance time-series as well as the cloud and shadow masks.

3.2.2. Spatial resampling

In this work, a total of 10 bands is used: B2, B3, B4, B5, B6, B7, B8, B8A, B11 and B12. The nomenclature of these bands and their characteristics are presented in Table 3.1. Among these bands, four of them (B2, B3, B4, B8) have a spatial resolution of 10m whereas six of them (B5, B6, B7, B8A, B11, B12) have a spatial resolution of 20m. Bands at 20 m/pixel are spatially upsampled to 10 m/pixel using bicubic interpolation, as implemented in the Orfeo ToolBox and its SuperImpose application [Grizonnet et al., 2017]. An example of an image from the band B5 spatially resampled from 20m to 10m is provided in Figure 3.4. Having data cubes of the same size for all bands facilitates processing.

3.2.3. Feature extraction

In addition to the spectral channels, three spectral indices are also calculated: **NDVI**, **NDWI**, and **Brightness**. These indices were selected by the **Centre d'Expertise Scientifique Occupation des SOIs (CES OSO)** to produce the **OSO land cover map**. CES OSO of Theia² is a group made

²<https://www.theia-land.fr/>

up of researchers and scientists from different national laboratory teams working on the **OSO** land cover map. Figure 3.5 represents a Sentinel-2 image as a true color **RGB** composition and this same image for these three spectral indices: **NDVI**, **NDWI**, and Brightness. The blue, green, orange and red polygons represent a water source, a forest, a cultivated field and a bare soil, respectively.

As described in Chapter I, the **NDVI** is used to measure photosynthetic vegetation activity. It corresponds to the combination of the Sentinel-2 spectral bands B4 and B8 and can be written such as:

$$\frac{B8 - B4}{B8 + B4} \quad (3.1)$$

In Figure 3.5, both forest and cultivated field polygons correspond to **NDVI** values close to 1 (white values). The bare soil polygon corresponds to **NDVI** close to 0 (gray values) and the water polygon corresponds to **NDVI** values close to -1 (black values).

The **NDWI** is used to monitor content changes in surface water [McFEETERS, 1996]. It is the combination of the Sentinel-2 spectral bands B3 and B8 and can be written as:

$$\frac{B3 - B8}{B3 + B8} \quad (3.2)$$

In general, such as for **NDVI**, **NDWI** values range from -1 to 1 . High **NDWI** values correspond to water bodies whereas low **NDWI** values correspond to drought, non-aqueous surfaces. As shown in Figure 3.5, the water polygon stands out to other polygons. Indeed, the forest, cultivated field or bare soil polygons are not identifiable and cannot be distinguished from one another.

The Brightness Index is used in order to measure the brightness of soil [Khan et al., 2005]. As shown in Figure 3.5, the Brightness, such as the **NDVI**, is useful to differentiate cultivated field and bare soil. In this work, it is defined as the Euclidean norm of all the bands [Inglada et al., 2017].

The 10 resampled bands and these three spectral indices are used as features to produce the land cover maps in Part II and Part III. Finally, a total of $D = 10 + 3 = 13$ spectral features are defined for each pixel \mathbf{x}_i at each time t_k .

3.2.4. Spatial information extraction

In addition to spectral information, two geographic coordinates, ψ_{1i} and ψ_{2i} , can also be extracted for each pixel \mathbf{x}_i . These spatial features are in meters in the Lambert 93 projection. The coordinates are centered on the point: $(X0, Y0) = (700000, 6600000)^3$ (in meters), as illustrated in Figure 3.6. The corresponding EPSG code is 2154. These geographic coordinates are used in two different ways in Part II and Part III.

3.2.5. Temporal resampling

All available acquisitions between January 2018 and December 2018 for the 27 Sentinel-2 tiles are used in this work. Five orbits are covering the study area, as shown in Figure 3.7. Combining the 5 orbits, $T = 303$ unique acquisition dates are available.

³This point corresponds roughly to the middle of France (including Corsica). Centering in longitude is not very important. However, centering in latitude is very important: it corresponds to the median latitude between the two latitudes where the projection cone intersects the Earth.

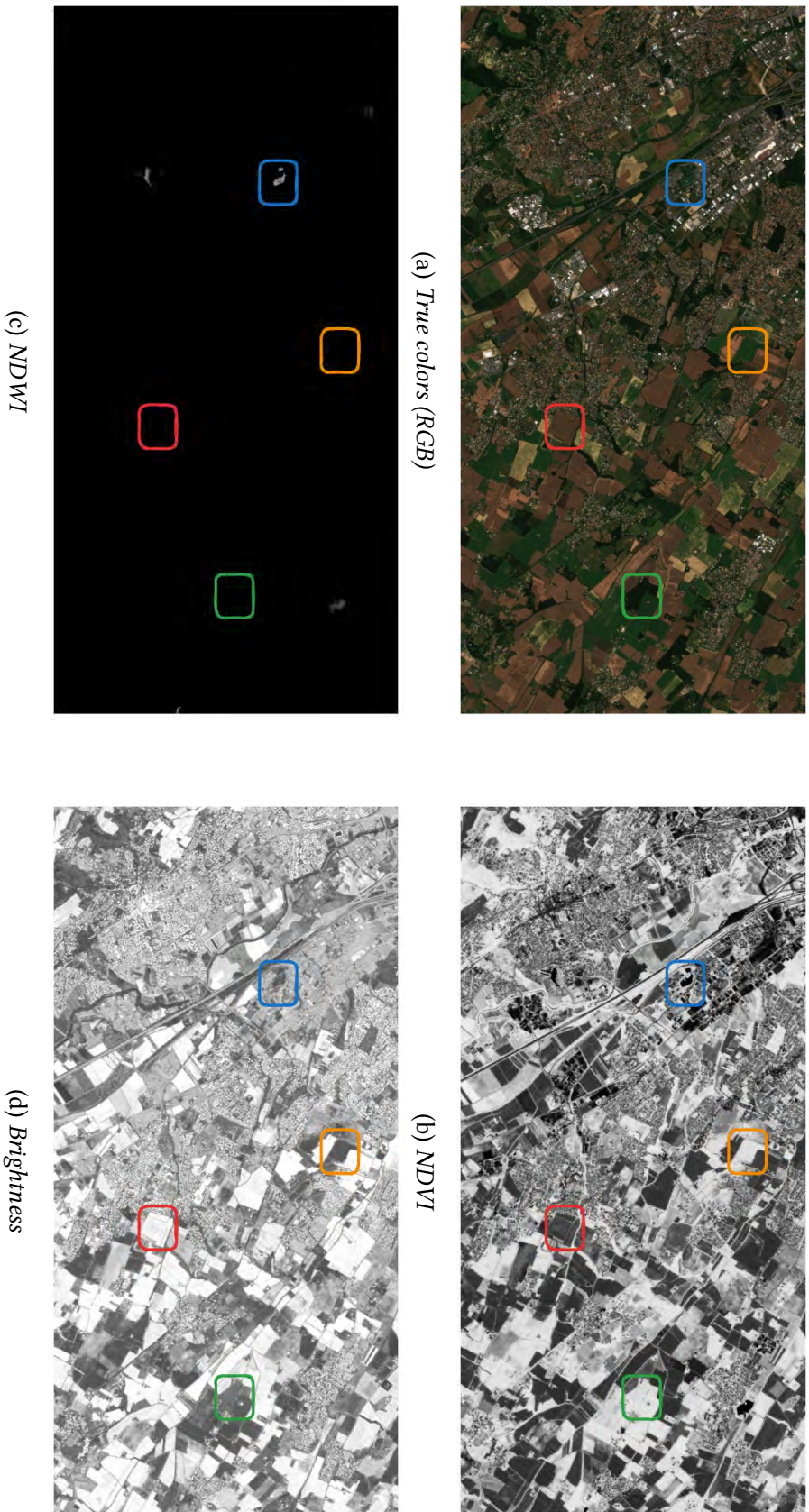


Figure 3.5: Sentinel-2 image with different band combinations: Red, Green, Blue (RGB), NDVI, NDWI and Brightness (05/07/2023). For the spectral indices (NDVI, NDWI and Brightness), high values are represented in white whereas low values are represented in black. (Source: <https://apps.sentinel-hub.com/sentinel-playground/>)



Figure 3.6: Representation of the point (X_0, Y_0) in Lambert 93 projection (background map © OpenStreetMap contributors).

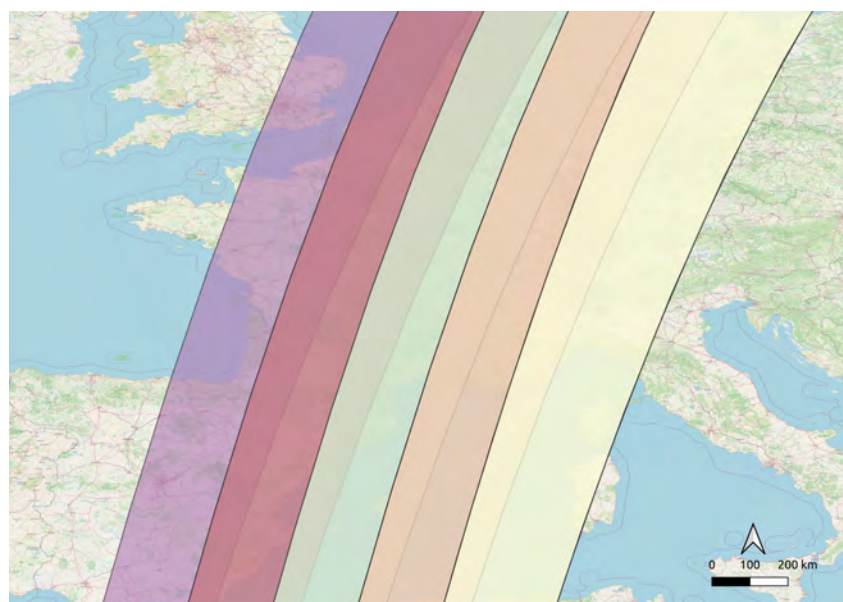


Figure 3.7: Sentinel-2 orbits used for the study area (background map © OpenStreetMap contributors).

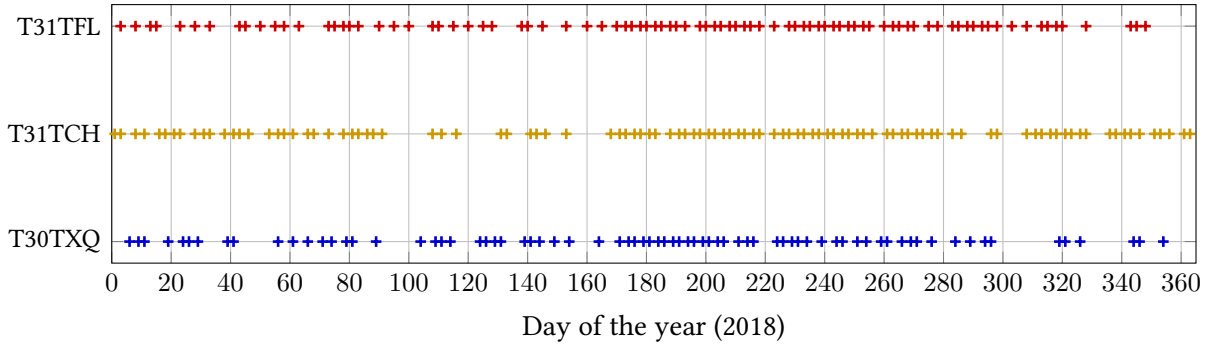


Figure 3.8: Temporal grids for three different tiles: $T30TXQ$, $T31TCH$, $T31TFL$.

As described in Chapter 1, Sentinel-2 pixel time series are unaligned in the temporal domain: observations from two different satellite swaths have different temporal sampling grids. Figure 3.8 represents the temporal grids for three tiles: $T30TXQ$, $T31TCH$, $T31TFL$. Moreover, the time series are irregularly sampled in the temporal domain: observations are not equally spaced in time due to the presence of clouds or cloud shadows. Some gaps are found in the figure because images containing more than 90% of clouds are not processed to the level 2A.

Figure 3.9 represents the NDVI profile for three pixels from these same tiles. Both valid dates and cloudy / shadow dates are represented in this figure. Valid date corresponds to an acquired observation where no cloud or cloud shadow is detected by the level 2A processor. The pixel from the tile $T31TFL$ has long periods without valid dates. The pixel from the tile $T31TCH$ is the least impacted by the cloudy / shadow dates.

The number of valid dates from 230 000 pixels randomly selected in the study area was computed. Figure 3.10 represents the histogram of these valid dates. Valid dates range from 8 to 79 with a mean value around 37. Moreover, three modes are found, they can be explained by the overlap of the orbits. Figure 3.11 represents the averaged number of valid dates for each pixel per month. July, August and September are the months with the largest number of valid dates, with more than 5 dates for each pixel. In contrast, February, November and December are the months with the smallest number of valid acquisitions, with less than 2 dates per pixel.

To cope with the clouds and cloud shadows and different temporal sampling among the tiles, the data can be linearly resampled onto a common set of virtual dates with an interval of ten days, for a total of $T = 37$ dates, as described in [Inglada et al., 2017]. The first date corresponds to the day 1 of the year and the last day corresponds to the day 361 of the year. Figure 3.12 represents the linear interpolation for the three pixels described previously. This processing is optional and is only used in Part II. In Part III, we propose to directly deal with the raw time series.

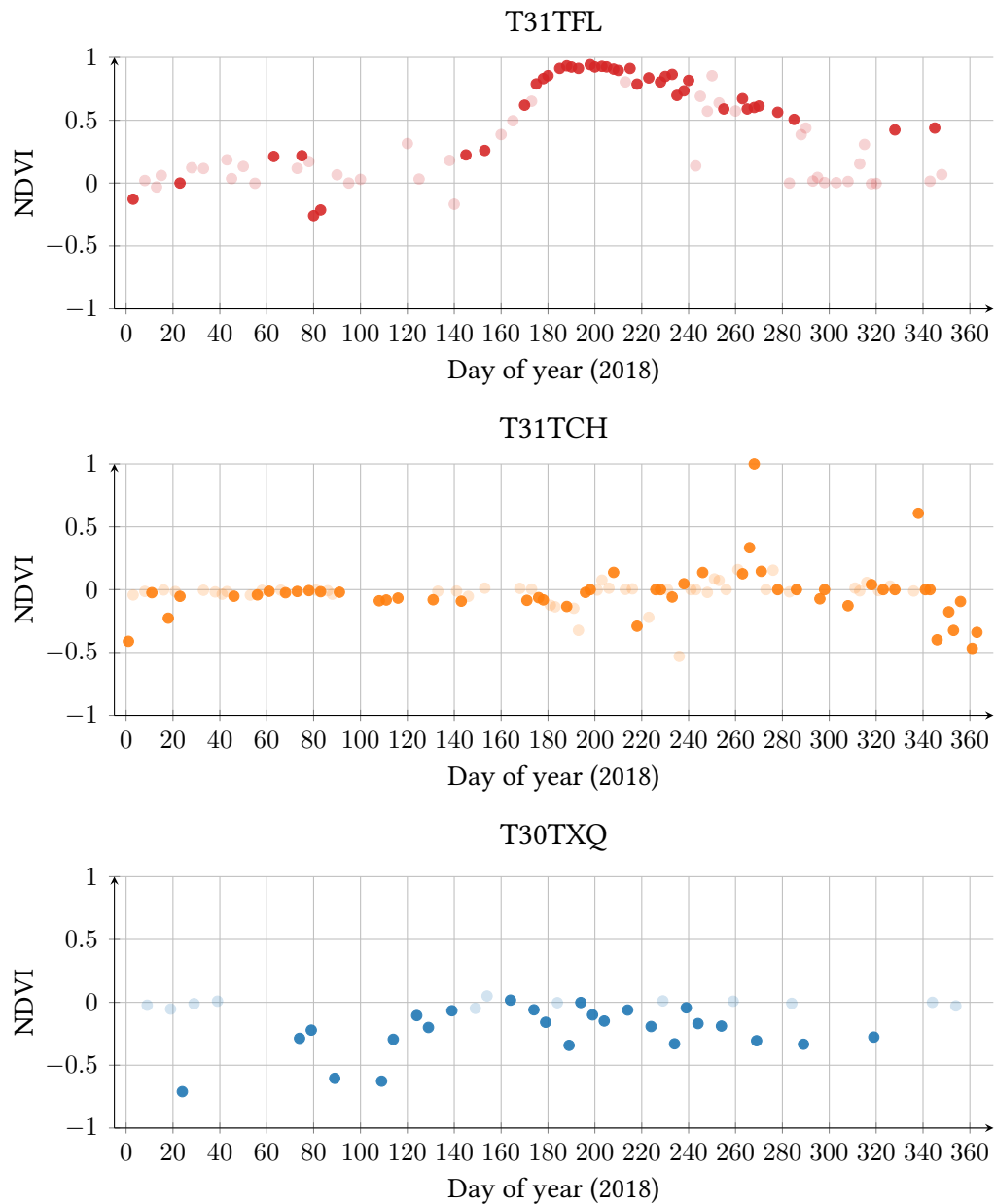


Figure 3.9: NDVI time series for three pixels from different tiles: *T30TXQ*, *T31TCH*, *T31TFL*. Filled dots correspond to valid observations, transparent dots correspond to observations flagged as clouds or cloud shadows in the level 2A masks.

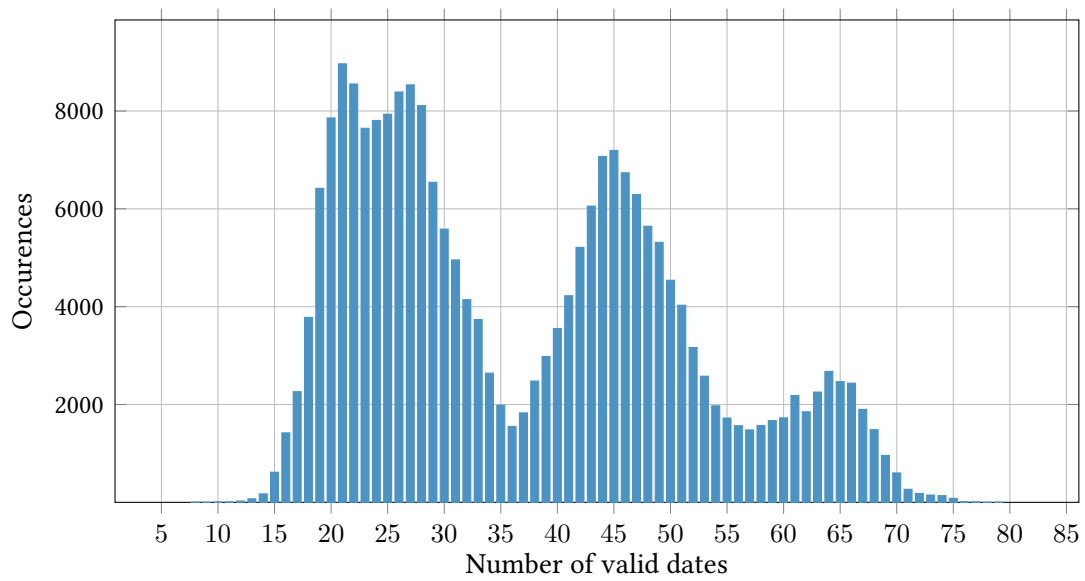


Figure 3.10: Histogram of the number of valid dates for 230 000 pixels in the study area.

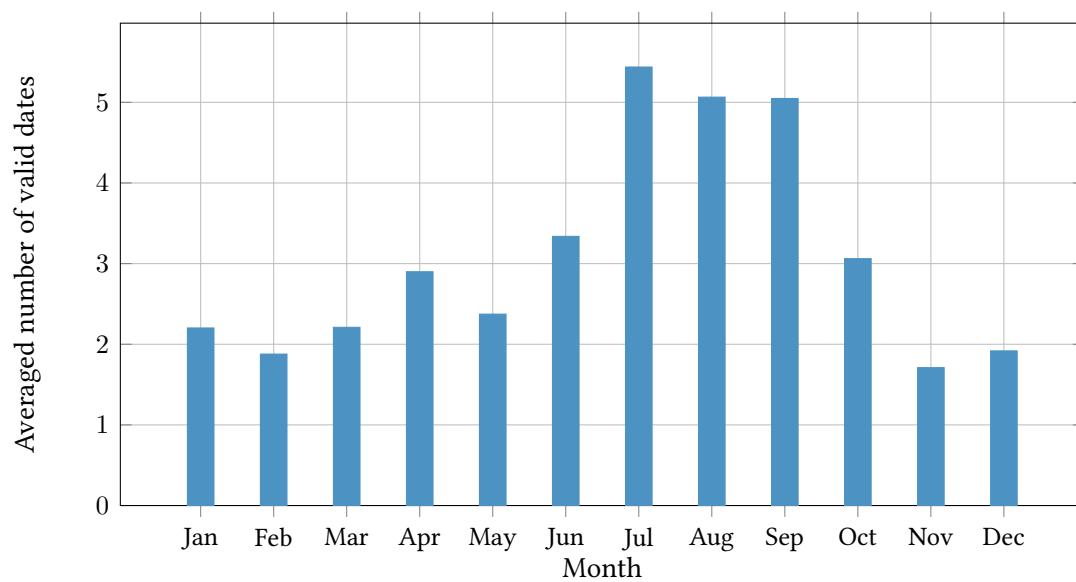


Figure 3.11: Averaged number of valid dates per pixel for each month. The mean was calculated from 230 000 pixels in the study area.

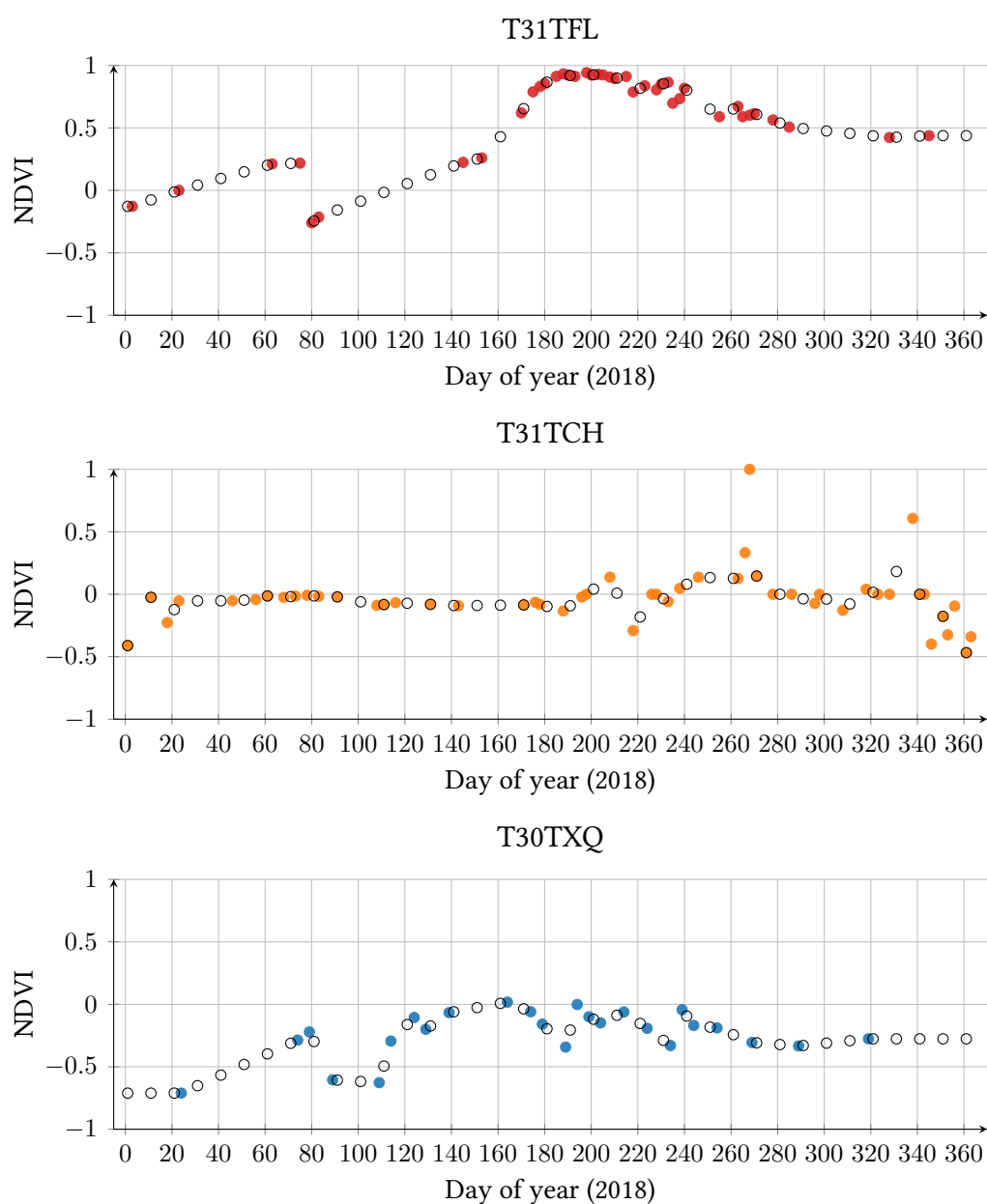


Figure 3.12: NDVI time series for three pixels from different tiles: *T30TXQ*, *T31TCH*, *T31TFL*. The black circled markers correspond to the linear interpolation of the valid dates with an interval of ten days, for a total of 37 dates dots. First interpolated date: day 1 of the year. Last interpolated date: day 361 of the year.

3.3. Reference data

The reference data used in this work is composed of $C = 23$ land cover classes ranging from artificial areas to vegetation and water bodies as described in Table 3.3. It corresponds to the **OSO** Theia Land Cover nomenclature. For the production of the 2016 and 2017 national maps, the **OSO** nomenclature was composed of 17 classes. Since 2018, the nomenclature has been updated to 23 classes. The "summer crop" class was split in five classes: soy, sunflower, corn, rice and tubers/roots. The "winter crop" class was split in three classes: straw cereals, rapeseed and protein crops.

3.3.1. Sources

The **OSO** nomenclature is the result of the fusion of different data sources:

1. CORINE Land Cover (**CLC** 2012). This data source was described in Chapter I. As a reminder, it is an inventory of land cover in 44 classes with a **MMU** of 25ha [Bossard et al., 2000], [Feranec et al., 2016]. It is coordinated by the **EEA**. The different classes are described in Table 3.4. The corresponding classes used to construct the **OSO** nomenclature are shown in bold in this table. **CLC** from 2012 was used to construct the reference data set of 2018. Artificial surfaces are considered to be permanent classes that change only slightly.
2. **UA** (2012). It is a geometrically accurate description of the various artificial cover types with a **MMU** of 25ha [Montero et al., 2014]. This data source was also described in Chapter I. Such as **CLC**, it is coordinated by the **EEA** and produced the same year than **CLC**. **UA** was used to construct the **OSO** class: road surfaces. **UA** from 2012 was used to construct the reference data set of 2018. Road surfaces are considered to be permanent classes that change only slightly.
3. French National Geographic Institute (BD-Topo). It is the national topographical map produced by **Institut Géographique National (IGN)** [Maugeais et al., 2011]. The spatial resolution is in the meter range. Since 2019, it is updated every three years. It was used to construct five different **OSO** classes: road surface, broad-leaved forest, coniferous forest, woody moorlands and water bodies.
4. Agricultural Land Parcel Information System, **Registre Parcellaire Graphique (RPG)** (2018). It is a spatial register of agricultural parcels coordinated by **Agence de Services et de Paiement (ASP)** and **IGN**. The crop type is provided by farmer declarations [Cantelaube and Carles, 2014]. Only the crop fields that obtain subsidies from the European Common Agricultural Policy are present in the database. **RPG** is composed of 24 crop groups as described in Table 3.5, at field level. This table also provides the corresponding classes of the **OSO** nomenclature. All the **OSO** crop classes are based on the **RPG** as shown in Table 3.3.
5. **Randolph Glacier Inventory (RGI)**. It is a global inventory of glacier outlines [Pfeffer et al., 2014]. It is part of the **Global Land Ice Measurements from Space (GLIMS)** initiative. Only the **OSO** class "glaciers and perpetual snows" class uses the **RGI** (version 6 released in 2017).

Table 3.3.: *OSO nomenclature and its corresponding sources [Derksen, 2019].*
























Color	Code	Name	Short description	Source
	CUF	Continuous urban fabric	Buildings, roads and artificially surfaced areas cover more than 80% of the total surface.	CLC 111
	DUF	Discontinuous urban fabric	Buildings, roads and artificially surfaced areas mixed with vegetated areas and bare soil.	CLC 112
	ICU	Industrial and commercial units	Artificially surfaced areas with concrete, asphalt.	CLC 121
	RSF	Road surfaces	Motorway rest areas, parking areas, motorway networks, larger than 50 m.	Urban Atlas, BD Topo
	RAP	Rapeseed	Rapeseed crops.	RPG
	STC	Straw cereals	Mainly wheat, oats, rye, barley, and buckwheat crops.	RPG
	PRO	Protein crops	Mainly peas, beans and lupins crops.	RPG
	SOY	Soy	Soy crops.	RPG
	SUN	Sunflower	Sunflower crops.	RPG
	COR	Corn	Corn crops.	RPG
	RIC	Rice	Rice crops.	RPG
	TUB	Tubers / roots	Mainly potatoes, beetroot and sugar beet crops.	RPG
	GRA	Grasslands	Dense grass cover, of floral composition, not under a rotation system.	RPG
	ORC	Orchards and fruit growing	Parcels planted with fruit trees or shrubs.	RPG
	VIN	Vineyards	Areas planted with vines.	RPG
	BLF	Broad-leaved forest	Forest of broad leaved trees.	BD Topo
	COF	Coniferous forest	Forest of coniferous trees.	BD Topo
	NGL	Natural grasslands	Low productivity grassland. Includes rocky areas, briars and heathland.	CLC 321
	WOM	Woody moorlands	Spontaneous vegetation dominated by woody and semi-woody plants.	BD Topo
	NMS	Natural mineral surfaces	Scree, cliffs, and rock outcrops.	CLC 332
	BDS	Beaches, dunes and sand plains	Beaches, dunes and expanses of sand or pebbles.	CLC 331
	GPS	Glaciers and perpetual snows	Land covered by glaciers or permanent snowfields.	RGI
	WAT	Water bodies	All water bodies longer than 20 m and all water courses larger than 7.5 m.	CLC 523, BD Topo

Table 3.4.: CORINE (CLC 2012) nomenclature [Bossard et al., 2000] [Feranec et al., 2016]. Bold classes correspond to the ones used in the OSO nomenclature.

Code	Description	Code	Description
111	Continuous urban fabric	311	Broad-leaved forest
112	Discontinuous urban fabric	312	Coniferous forest
121	Industrial or commercial units	313	Mixed forest
122	Road and rail networks and associated land	321	Natural grassland
123	Port areas	322	Moors and heathland
124	Airports	323	Sclerophyllous vegetation
131	Mineral extraction sites	324	Transitional woodland/shrub
132	Dump sites	331	Beaches, dunes, sands
133	Construction sites	332	Bare rock
141	Green urban areas	333	Sparsely vegetated areas
142	Sport and leisure facilities	334	Burnt areas
211	Non-irrigated arable land	335	Glaciers and perpetual snow
212	Permanently irrigated land	411	Inland marshes
213	Rice fields	412	Peatbogs
221	Vineyards	421	Salt marshes
222	Fruit trees and berry plantations	422	Salines
223	Olive groves	423	Intertidal flats
231	Pastures	511	Water courses
241	Annual crops associated with permanent crops	512	Water bodies
242	Complex cultivation patterns	521	Coastal lagoons
243	Land principally occupied by agriculture, with significant areas of natural vegetation	522	Estuaries
244	Agro-forestry areas	523	Sea and ocean

Table 3.5.: *RPG nomenclature and the correspondence with the OSO nomenclature [ASP, 2018].*

RPG	OSO
Soft wheat	Straw cereals
Corn	Corn
Barley	Straw cereals
Other cereals	Straw cereals
Rapeseed	Rapeseed
Sunflower	Sunflower
Other oilseeds	Soy
Protein crops	Protein crops
Fiber plants	-
Set-aside areas without production	-
Rice	Rice
Legume	Protein crops (lentils, chickpeas)
Fodder	-
Meadows - moors	-
Permanent pasture	Grasslands
Temporary pastures	Grasslands
Orchards	Orchards and fruit growing
Vineyards	Vineyards
Nut	Orchards and fruit growing
Olive tree	Orchards and fruit growing
Other industrial crops	Tubers/Roots (beet)
Vegetables - flowers	Tubers/Roots (potatoe)
Sugarcane	-
Miscellaneous	-



Figure 3.13: Example of polygons of different shapes and sizes. The colors correspond to the nomenclature defined in Table 3.3 (Background: Sentinel-2 level 2A 20/06/2018 T31TDJ tile).

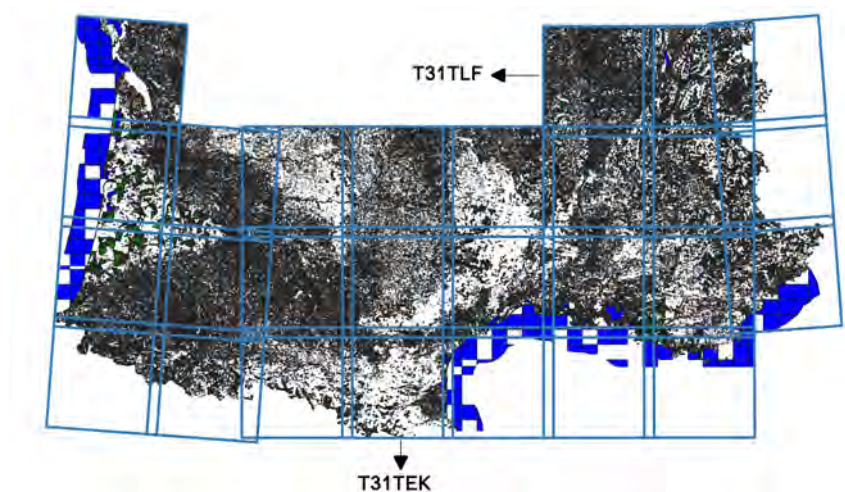


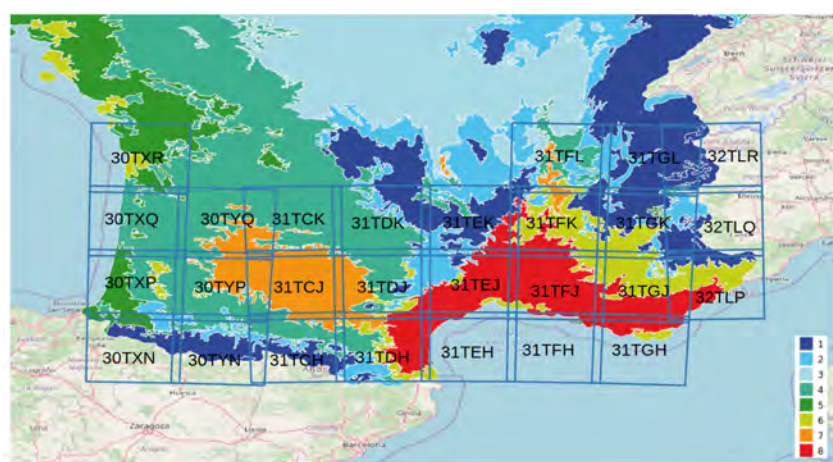
Figure 3.14: Repartition of the polygons in the study area.

3.3.2. Polygons

Following the methodology described in [Inglada et al., 2017], all the information from these different sources has been aggregated, both spatially and semantically, to create the reference data set. It is provided as a set of non-overlapping spatial polygons of different shapes and sizes, as represented in Figure 3.13. The approximately 600 000 polygons are relatively well distributed over the study area, as shown in Figure 3.14. Although there are some areas where there are fewer polygons. For example, the *T31TFL* tile consists of around 50 000 polygons whereas the *T31TEK* is composed of around 20 000 polygons.

Table 3.6.: *Eco-climatic regions nomenclature.*

Color	Code	Name
■	1	Mountainous
■	2	Semi-continental and mountain margins
■	3	Degraded oceanic from central and northern plains
■	4	Altered oceanic
■	5	Straightforward oceanic
■	6	Altered Mediterranean
■	7	South-west basin
■	8	Straightforward Mediterranean

Figure 3.15: *Eco-climatic regions (regions 1-8) for the study area (background map © OpenStreetMap contributors).*

3.3.3. Eco-climatic regions

Eco-climatic regions can be used to stratify the studied area into sub-regions, as proposed in [Joly et al., 2010]. Eight different regions were proposed and their nomenclature is described in Table 3.6. In each eco-climatic region, meteorological and topographical conditions are similar. Inglada *et al.* [Inglada et al., 2017] proposed to divide the training data set and train a model for each eco-climatic region. This allows to reduce the spectro-temporal variability of pixel reflectances. It also enables to reduce the massive training data set which is needed by some machine learning algorithms which do not scale well. In Part II, eco-climatic regions are used to train the different models.

Figure 3.15 presents the eco-climatic regions over our study area. All the eco-climatic regions are represented in the study area, but with varying proportions as shown in Figure 3.16. Region 4, corresponding to "altered oceanic", is the most represented region. Region 3, corresponding to "degraded oceanic from central and northern plains", is the least represented, as our study area covers the south of France. Regions 2, 6 and 7 are roughly equivalent, each covering around 10% of the study area.

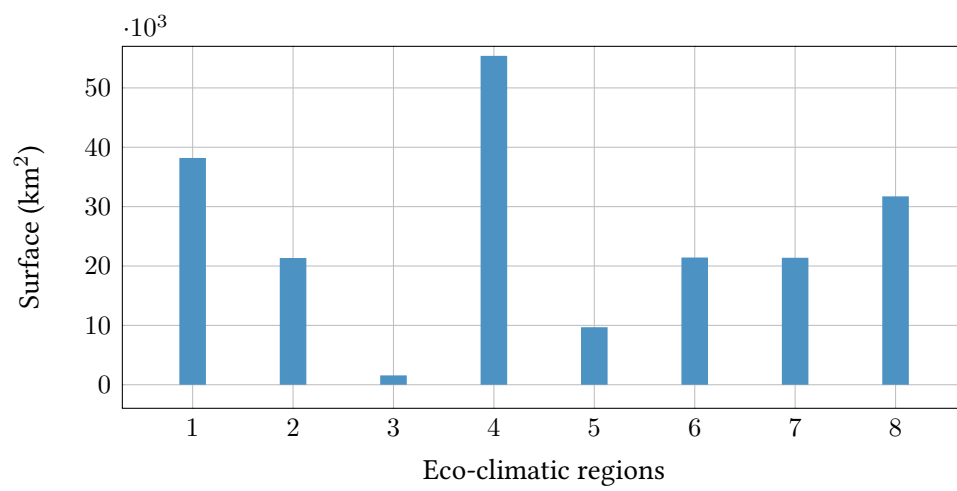


Figure 3.16: Surface (in km²) of each eco-climatic region in the study area.

3.4. Data set selection

The different data sets were produced using the `iota`² software [Inglada et al., 2016]. Figure 3.17 represents the different steps for producing three different datasets (train, validation, test).

3.4.1. Polygon selection

From the approximately 600 000 polygons, one third were randomly selected in order to create our data set used in the following chapters. 80 000, 20 000 and 100 000 polygons were extracted separately and randomly to build the *training*, *validation* and *test* polygons, respectively. Figure 3.17b represents some polygons selected from the initial polygons in Figure 3.17a.

3.4.2. Pixel selection

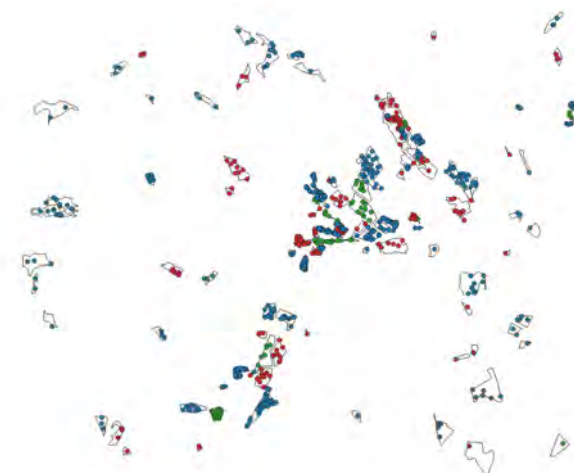
From the polygons described previously, different pixels are randomly extracted in order to form three *spatially disjoint* data subsets: *training*, *validation* and *test*. Indeed, pixels from one polygon fully belong to a unique data subset (either *training*, *validation* and *test*). Figure 3.17c represents the pixels extracted from the polygons. Depending on the polygon, more or fewer pixels are extracted. The number of pixels selected for each experiment is described in chapters 5 and 8.



(a) *Initial polygons*



(b) *Polygon selection*



(c) *Pixel selection*

Figure 3.17: *Data set selection. In Figure 3.17a colors correspond to the nomenclature defined in Table 3.3. Red, green and blue colors in Figures 3.17b and 3.17c respectively represent the training, validation and test data sets.*

Part II.

Gaussian Processes for land cover classification using SITS at large scale

CHAPTER 4

REVIEW ON GAUSSIAN PROCESSES

4.1. Gaussian Distribution	108
4.1.1. Univariate Gaussian Distribution	108
4.1.2. Multivariate Gaussian Distribution	109
4.2. Univariate Gaussian Processes	113
4.2.1. Definition	113
4.2.2. Gaussian Process Regression	118
4.2.3. Binary Classification	124
4.3. Multivariate Gaussian Processes	125
4.3.1. Definition	125
4.3.2. Multi-output regression	128
4.3.3. Multi-class classification	132
4.4. Large scale Gaussian Processes	133
4.4.1. Model Approximation	134
4.4.2. Posterior Approximation by Variational Inference	137

This chapter presents how Gaussian Processes (GP) work, their advantages and limitations and the solutions that allow to overcome the latter.

4.1. Gaussian Distribution

In order to easily introduce GP, a reminder on the Gaussian distribution is proposed.

4.1.1. Univariate Gaussian Distribution

A random variable X follows a Gaussian distribution denoted as:

$$X \sim \mathcal{N}_1(\mu, \sigma^2)$$

with μ its mean and σ^2 its variance if its **Probability Density Function (PDF)** is given by the following equation [Bishop, 2006, Chapter 2.3]:

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right).$$

Figure 4.1 represents the PDF of the standard normal random variable $X_1 \sim \mathcal{N}_1(0, 1)$ and two other random variables $X_2 \sim \mathcal{N}_1(2, 3)$ and $X_3 \sim \mathcal{N}_1(0, 0.2)$.

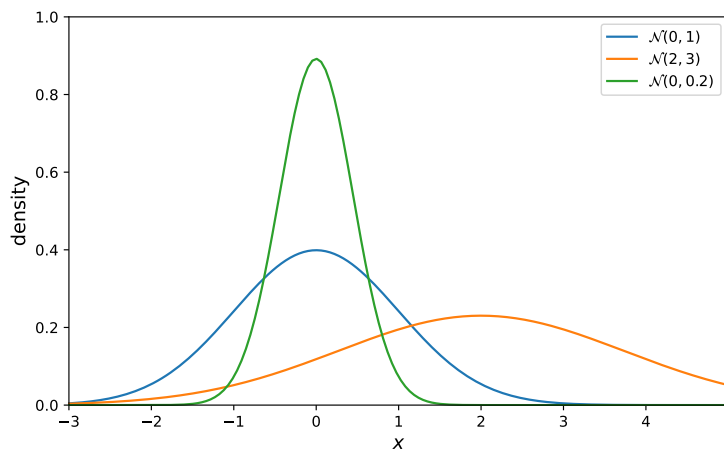


Figure 4.1: Probability density functions of $X_1 \sim \mathcal{N}_1(0, 1)$, $X_2 \sim \mathcal{N}_1(2, 3)$ and $X_3 \sim \mathcal{N}_1(0, 0.2)$.

4.1.2. Multivariate Gaussian Distribution

A d -dimensional random vector $\mathbf{X} = (X_1, \dots, X_d)$, with $X_1 \sim \mathcal{N}_1(\mu_1, \sigma_1^2), \dots, X_d \sim \mathcal{N}_1(\mu_d, \sigma_d^2)$ follows a d dimension Gaussian distribution denoted as:

$$\mathbf{X} \sim \mathcal{N}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

with $\boldsymbol{\mu} = (\mu_1, \dots, \mu_d)$ its mean vector and $\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \dots & \sigma_{1d}^2 \\ \dots & \dots & \dots \\ \sigma_{1d}^2 & \dots & \sigma_d^2 \end{pmatrix}$ its covariance matrix. The joint PDF is defined as:

$$f_{\mathbf{X}}(x_1, \dots, x_d) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} \left((\mathbf{X} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{X} - \boldsymbol{\mu}) \right)\right),$$

with $|\cdot|$ which denotes the determinant, if its covariance matrix is symmetric positive semi-definite (i.e. $\boldsymbol{\Sigma}$ is symmetric and $\mathbf{X}^\top \boldsymbol{\Sigma} \mathbf{X} \geq 0, \forall \mathbf{X} \in \mathbb{R}^d$).

If we assume that \mathbf{X} is split into two parts, such as $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$ where \mathbf{X}_1 is of size c and \mathbf{X}_2 is of size $(d - c)$ with $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)$ and $\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{12} & \boldsymbol{\Sigma}_{22} \end{pmatrix}$, then we have the following results.

Marginalization: Every marginal distribution of a Gaussian distribution is itself a Gaussian distribution. Thus, the c -dimensional marginal distribution of \mathbf{X}_1 is $\mathcal{N}_c(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11})$. In the case of a bivariate Gaussian distribution (i.e. $\mathbf{X} = (X_1, X_2)$), $f_{X_1}(x_1)$ and $f_{X_2}(x_2)$ corresponding to the marginal PDF of X_1 and X_2 , respectively, are defined such as:

$$f_{X_1}(x_1) = \frac{1}{\sigma_1 \sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{x_1 - \mu_1}{\sigma_1}\right)^2\right) \text{ and } f_{X_2}(x_2) = \frac{1}{\sigma_2 \sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{x_2 - \mu_2}{\sigma_2}\right)^2\right).$$

If \mathbf{X}_1 and \mathbf{X}_2 are independent ($\boldsymbol{\Sigma}_{12} = \mathbf{0}$), the joint PDF is equal to the product of the marginal PDF such as $f_{\mathbf{X}_1, \mathbf{X}_2}(x_1, \dots, x_d) = f_{\mathbf{X}_1}(x_1, \dots, x_c) f_{\mathbf{X}_2}(x_{c+1}, \dots, x_d)$.

Conditional distribution: The c -dimensional distribution of \mathbf{X}_1 conditional on \mathbf{X}_2 also follow a Gaussian distribution such as:

$$\mathbf{X}_1 | \mathbf{X}_2 \sim \mathcal{N}_c(\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1} (\mathbf{X}_2 - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{12}^\top). \quad (4.1)$$

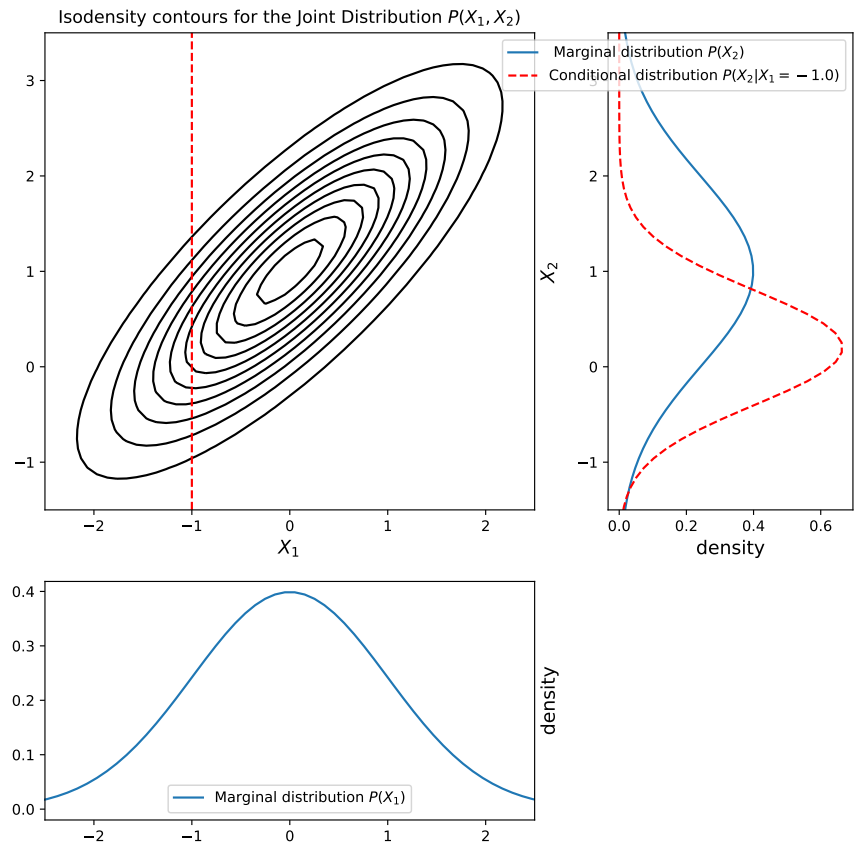
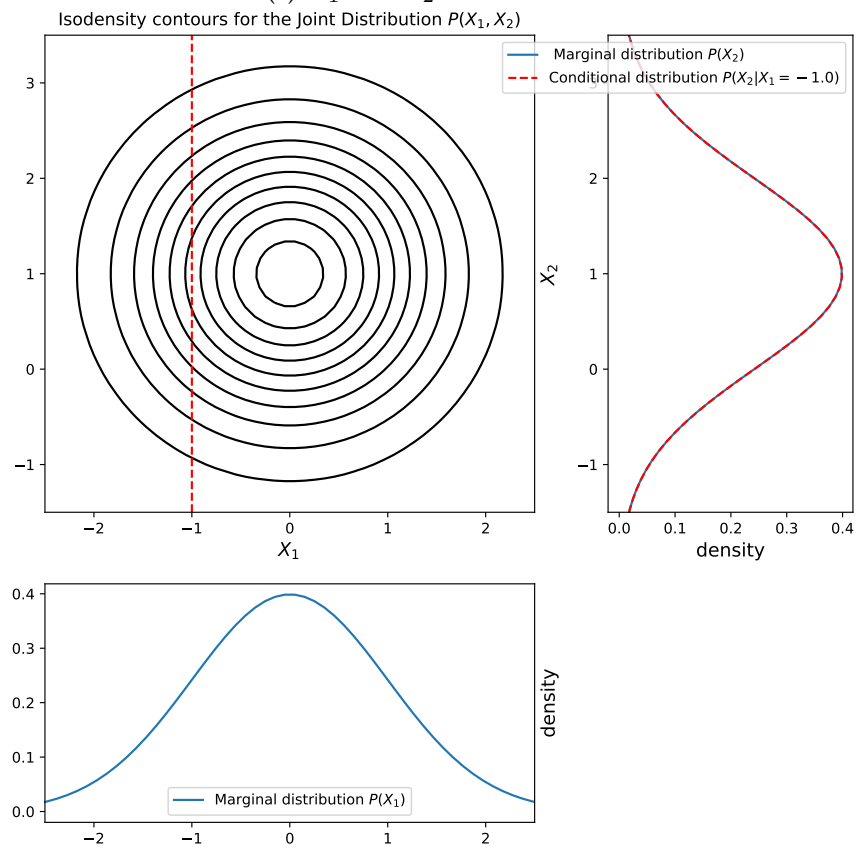
In the case of a bivariate Gaussian distribution, we have:

$$X_1 | X_2 = x_2 \sim \mathcal{N}_1(\mu_{1|2}, \sigma_{1|2}^2) \text{ and } X_2 | X_1 = x_1 \sim \mathcal{N}_1(\mu_{2|1}, \sigma_{2|1}^2)$$

with $\mu_{1|2} = \mu_1 + \sigma_{12}^2 \sigma_2^{-2} (x_2 - \mu_2)$, $\sigma_{1|2}^2 = \sigma_1^2 - \sigma_{12}^2 \sigma_2^{-2} \sigma_{12}^2$, $\mu_{2|1} = \mu_2 + \sigma_{12}^2 \sigma_1^{-2} (x_1 - \mu_1)$ and $\sigma_{2|1}^2 = \sigma_2^2 - \sigma_{12}^2 \sigma_1^{-2} \sigma_{12}^2$.

If \mathbf{X}_1 and \mathbf{X}_2 are independent ($\boldsymbol{\Sigma}_{12} = \mathbf{0}$), the conditional distribution is equal to the marginal such as $f_{\mathbf{X}_1 | \mathbf{X}_2} = f_{\mathbf{X}_1}$. Indeed, knowing the value of \mathbf{X}_2 should not change the distribution of \mathbf{X}_1 and respectively.

In the case of a bivariate Gaussian distribution, Figures 4.2a and 4.2b represent the marginal, conditional and joint PDF of two variables X_1 and X_2 , correlated and independent, respectively. It clearly illustrates the fact that for independent variables the conditional PDF is equal to the marginal PDF.

(a) X_1 and X_2 are correlated(b) X_1 and X_2 are independentFigure 4.2: Representation of marginal, conditional and joint PDFs of two variables X_1 and X_2 .

Kullback-Leibler divergence: The **Kullback–Leibler (KL)** divergence between two probability distributions p and q is defined as:

$$\text{KL}[p||q] = \int_x p(x) \log \frac{p(x)}{q(x)}.$$

If p and q are two multivariate Gaussian distributions, such as $p \sim \mathcal{N}_d(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ and $q \sim \mathcal{N}_d(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$, **KL** divergence can be rewritten as:

$$\text{KL}[p||q] = \frac{1}{2} \left[\log \frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_p|} - d + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_q^{-1} (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) + \text{tr} \left\{ \boldsymbol{\Sigma}_q^{-1} \boldsymbol{\Sigma}_p \right\} \right].$$

Concept of Gaussian Processes: Figure 4.3 represents 4 different realizations of a bivariate Gaussian distribution. Two representations are used to plot these realizations. The first one, in Figure 4.3a, is the most widely used representation, where each variable corresponds to one axis. In second one, in Figure 4.3b, the two variables are represented in the same axis. The latter representation is used to make the concept of Gaussian Processes easier to understand. Indeed, Figure 4.4 represents 4 different realizations selected from several d dimensional Gaussian distribution with the same representation. When the dimension d goes to infinite, it is not anymore a random variable but a random process or a random function. Therefore, a Gaussian Process is defined as an extension of the multivariate Gaussian distribution to infinite dimension.

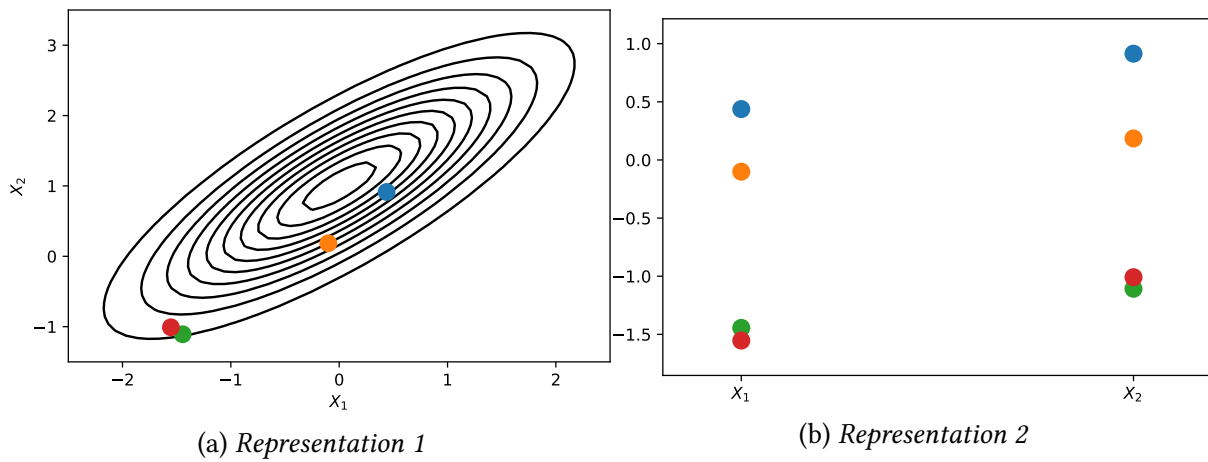


Figure 4.3: Two representations of 4 different realizations of a bivariate Gaussian distribution. In the left figure, each variable corresponds to one axis. In the right figure, the two variables are represented in the same axis (x -axis). In the following, the representation 2 will be used.

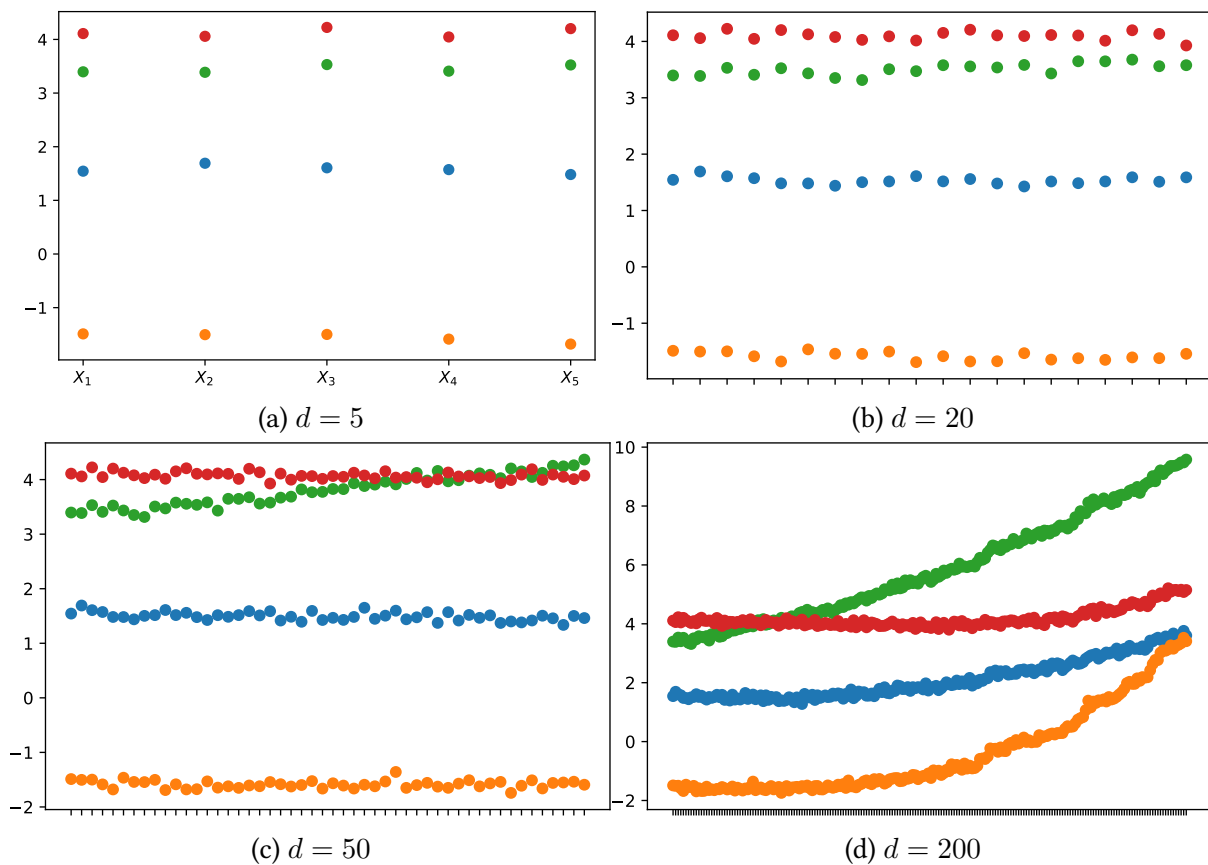


Figure 4.4: Representation of 4 different realizations of a d dimensional Gaussian distribution of increasing size ($d \in \{5, 20, 50, 200\}$).

4.2. Univariate Gaussian Processes

4.2.1. Definition

An univariate GP f is specified by its real-valued mean function m and its covariance function k : $f \sim \mathcal{GP}(m, k)$ [Rasmussen and Williams, 2005]. f is defined as a (potentially infinite) collection of random variables, any finite number of which have joint Gaussian distribution. Noting $f(\mathbf{X})$ the random vector defined as $f(\mathbf{X}) = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)]^\top$ with N the number of observations, $f(\mathbf{X})$ follows a multivariate Gaussian distribution: $f(\mathbf{X}) \sim \mathcal{N}_N(\boldsymbol{\mu}, \mathbf{K})$ with $\boldsymbol{\mu} = [m(\mathbf{x}_1), \dots, m(\mathbf{x}_N)]^\top$ and \mathbf{K} such as $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $\forall i, j \in \{1, \dots, N\}^2$. The parameters m and k , functions of \mathbf{x}_i , are usually modeled by parametric functions with hyper-parameters $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_k$, respectively. The choice of the parametric functions can be made by the user and depend on the application. The hyper-parameters $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_k$ can be optimized by gradient descent, as explained in Section 4.2.2. Figure 4.5 represents four realizations of two different GP. The mean of the GP is also represented. Different parametric functions were selected for these two GP leading to different representations. In the following, the most common mean and covariance functions are described.

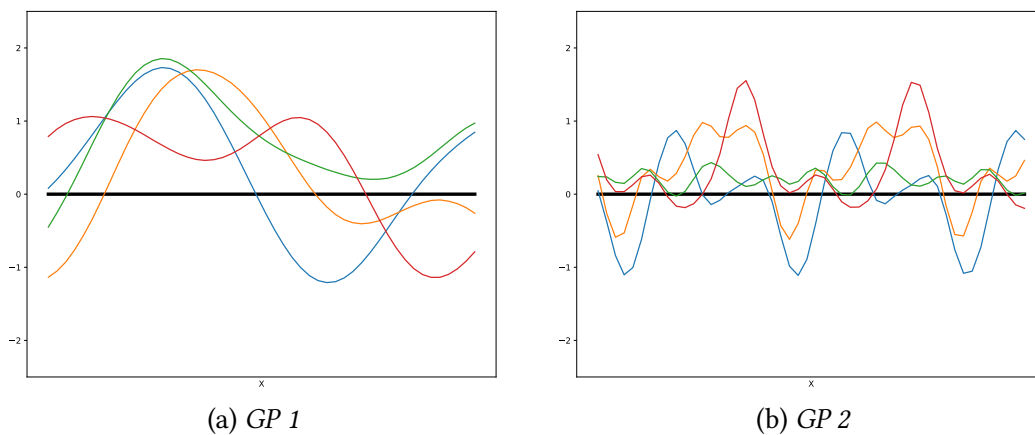


Figure 4.5: Representation of two different Gaussian Processes: GP 1 and GP 2. The colored curves correspond to 4 different realizations of the GP. The black curve corresponds to the mean of the GP here $\boldsymbol{\mu} = 0$. The GP 1 is composed of a mean function equal to zero and a covariance function corresponding to a RBF function defined in Table 4.2 with $\sigma = 1$ and $\ell = 1$. The GP 2 is composed of a mean function equal to zero and a covariance function corresponding to a periodic function defined in Table 4.2 with $\sigma = 1$ and $\ell = 2$.

Mean function

In statistics, the mean function is typically referred as the "prior mean" and is able to incorporate all the prior information. In contrast, in machine learning, the mean function is seldom used and it is the covariance function which incorporates most of the prior information. The most common mean functions m used in ML are represented in Table 4.1. Typically, for machine learning, the constant mean is usually used. It simplifies the GP model compared to the linear mean function or to more complex ones.

Table 4.1.: Description of the most common mean functions m .

Name	Formula $m(\mathbf{x})$	Hyperparameters θ_m
Zero	0	None
Constant	b	$\{b\}$
Linear	$a\mathbf{x} + b$	$\{a, b\}$

Covariance function

$k(\mathbf{x}, \mathbf{x}')$ indicates the correlation between \mathbf{x} and \mathbf{x}' , e.g. how they are close. k is constrained to be a symmetric positive semi-definite function [Rasmussen and Williams, 2005, Chapter 4] such as:

$$\sum_{i,j}^N k(\mathbf{x}_i, \mathbf{x}_j) \mathbf{x}_i \mathbf{x}_j = \sum_{i,j}^N K_{ij} \mathbf{x}_i \mathbf{x}_j = \mathbf{x}^T \mathbf{K} \mathbf{x} \geq 0, \quad \forall \mathbf{x} \in \mathbb{R}^N.$$

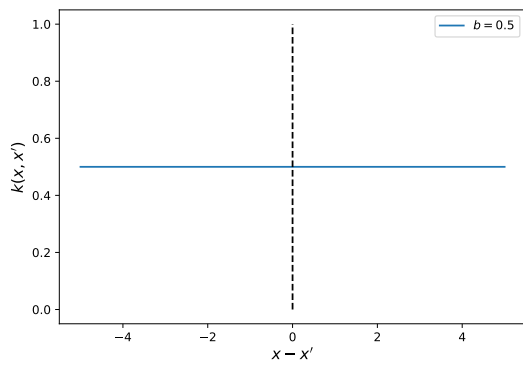
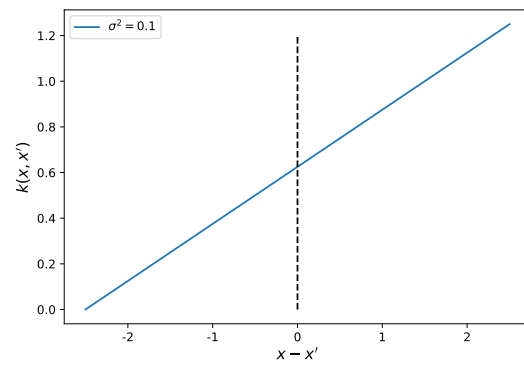
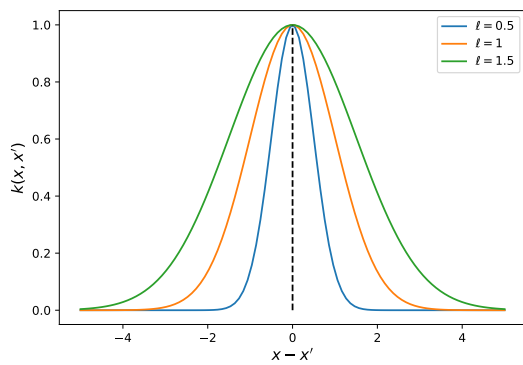
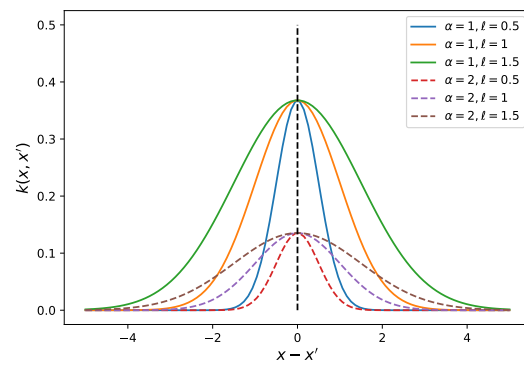
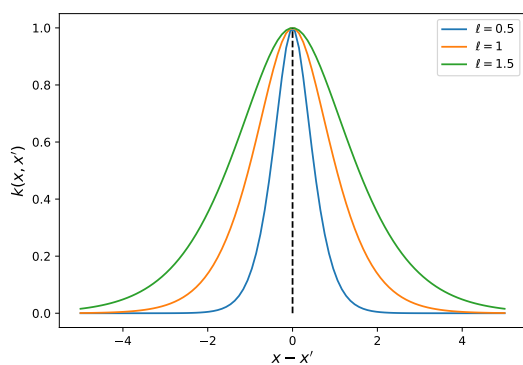
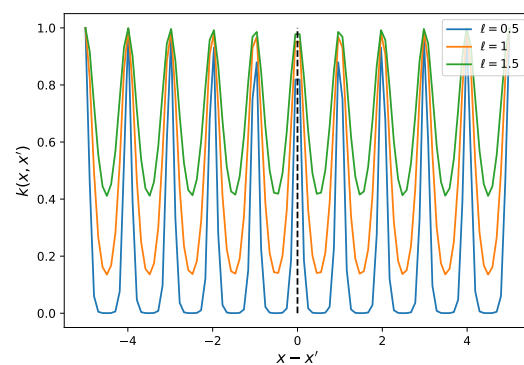
The most common covariance functions are detailed in Table 4.2 and their illustrations are provided in Figure 4.6. The choice of the covariance function allows to introduce prior knowledge and to infer properties of GP posteriors [Scholkopf and Smola, 2001], [Rasmussen and Williams, 2005].

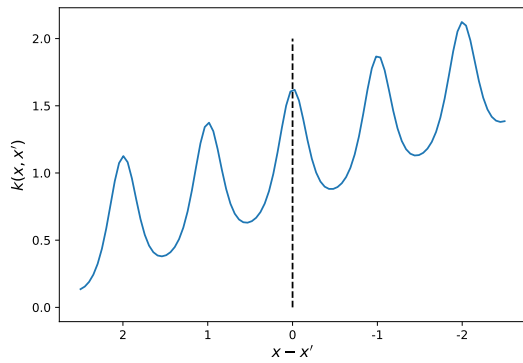
The most popular function used for GP is the Squared Exponential, also called **Radial Basis Function (RBF)** or Exponential Quadratic. This covariance function is isotropic (invariant under rotation) and stationary (invariant under translations). This operator can model smooth transitions for two "close" pixels, as represented in Figure 4.6c. **Rational Quadratic (RQ)** and Matérn 5/2 kernels are generalizations of the **Radial Basis Function (RBF)** kernel, as illustrated in Figures 4.6d and 4.6e. The **RQ** kernel can be considered as an infinite sum of RBF kernels with different length-scales ℓ [Duvenaud, 2014]. Finally, by using the Periodic kernel, the periodicity of the data can be taken into account, as illustrated in Figure 4.6f.

Sometimes, a covariance function alone does not meet all the requirements. One solution is to combine the covariance functions together. The covariance functions can either be added or multiplied together [Duvenaud, 2014]. Figures 4.6g, 4.6h, 4.6i and 4.6j illustrate different combinations of covariance functions. A periodic covariance function added to a linear covariance function corresponds to a periodic function with an increasing mean, as represented in Figure 4.6g. A periodic covariance function times a linear covariance function corresponds to a periodic function with an increasing amplitude, as represented in Figure 4.6h. A linear times a linear covariance function results in a quadratic function, as represented in Figure 4.6j.

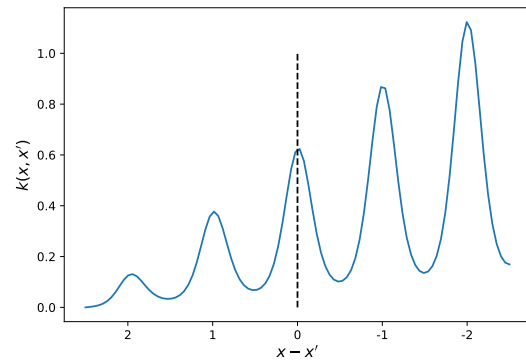
Table 4.2.: Description of the most common covariance functions k .

Name	Formula $k(\mathbf{x}, \mathbf{x}')$	Hyperparameters θ_k
Constant	b	$\{b\}$
Linear	$\sigma^2 \mathbf{x}^\top \mathbf{x}'$	$\{\sigma\}$
Squared Exponential (RBF)	$\sigma^2 \exp\left(-\frac{\ \mathbf{x}-\mathbf{x}'\ ^2}{2\ell^2}\right)$	$\{\sigma, \ell\}$
Rational Quadratic (RQ)	$\sigma^2 \exp\left(1 + \frac{\ \mathbf{x}-\mathbf{x}'\ ^2}{2\alpha\ell^2}\right)^{-\alpha}$	$\{\sigma, \ell, \alpha\}$
Matérn 5/2	$\sigma^2 \left(1 + \frac{\sqrt{5}\ \mathbf{x}-\mathbf{x}'\ }{\ell} + \frac{5\ \mathbf{x}-\mathbf{x}'\ ^2}{3\ell^2}\right) \exp\left(-\frac{\sqrt{5}\ \mathbf{x}-\mathbf{x}'\ }{\ell}\right)$	$\{\sigma, \ell\}$
Periodic	$\sigma^2 \exp\left(-\frac{2 \sin^2\left(\frac{\mathbf{x}-\mathbf{x}'}{2}\right)}{\ell^2}\right)$	$\{\sigma, \ell\}$

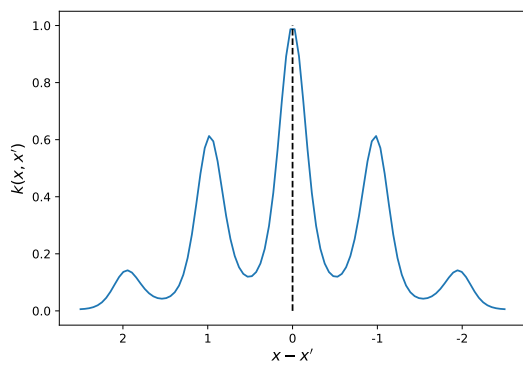
(a) Constant with $b = 0.5$.(b) Linear with $\sigma^2 = 0.1$.(c) RBF with $\sigma = 1$ and $l \in \{0.5, 1, 1.5\}$.(d) RQ with $\sigma = 1$ and $\alpha, l \in \{\{1, 0.5\}, \{1, 1\}, \{1, 1.5\}, \{2, 0.5\}, \{2, 1\}, \{2, 1.5\}\}$.(e) Matérn 5/2 with $\sigma = 1$ and $l \in \{0.5, 1, 1.5\}$.(f) Periodic with $\sigma = 1$ and $l \in \{0.5, 1, 1.5\}$.



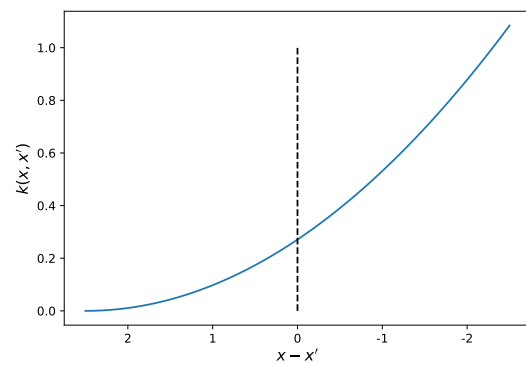
(g) *Periodic + Linear* corresponds to:
 $\exp\left(-\frac{2 \sin^2\left(\frac{x-x'}{2}\right)}{\ell^2}\right) + \sigma^2 \mathbf{x}^\top \mathbf{x}'$ with
 $\ell = 1$ and $\sigma^2 = 0.1$.



(h) *Periodic \times Linear* corresponds to:
 $\exp\left(-\frac{2 \sin^2\left(\frac{x-x'}{2}\right)}{\ell^2}\right) \times \sigma^2 \mathbf{x}^\top \mathbf{x}'$ with
 $\ell = 1$ and $\sigma^2 = 0.1$.



(i) *Periodic \times RBF* corresponds
to:
 $\exp\left(-\frac{2 \sin^2\left(\frac{x-x'}{2}\right)}{\ell^2}\right)$
 $\times \sigma^2 \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\ell^2}\right)$ with $\ell = 1$.



(j) *Linear \times Linear* corresponds to:
 $\sigma^2 \mathbf{x}^\top \mathbf{x}' \times \sigma^2 \mathbf{x}^\top \mathbf{x}'$ with $\sigma^2 = 0.1$.

Figure 4.6: Illustration of different covariance functions k .

4.2.2. Gaussian Process Regression

Univariate **GP** are commonly used to regress a scalar target value ($y_i \in \mathbb{R}$) through a link function ψ that relates the univariate latent variable $f(\mathbf{x}_i)$ to the observed y_i . We denote $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^\top$ and $\mathbf{y} = [y_1, \dots, y_N]^\top$.

To model realistic situations, an usual approach is to consider a noisy version of the function value such as

$$y_i = \psi(f(\mathbf{x}_i)) = f(\mathbf{x}_i) + \epsilon_i \quad (4.2)$$

with $\epsilon_i \sim \mathcal{N}_1(0, \sigma^2)$ and σ the noise level. By definition of a Gaussian Process, we have:

$$f(\mathbf{X}) \sim \mathcal{N}_N(\boldsymbol{\mu}, \mathbf{K}).$$

Assuming additive independent identically distributed Gaussian noise, \mathbf{y} is a sum of two independent multivariate Gaussian variables, thus we have:

$$\mathbf{y} \sim \mathcal{N}_N(\boldsymbol{\mu}, \mathbf{K} + \sigma^2 I_N).$$

Inference

The joint distribution of the observed values \mathbf{y} and the function values for a new input \mathbf{x}_* is expressed as:

$$\begin{pmatrix} \mathbf{y} \\ f(\mathbf{x}_*) \end{pmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \boldsymbol{\mu} \\ m(\mathbf{x}_*) \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \sigma^2 I_N & \mathbf{k}_* \\ \mathbf{k}_*^\top & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix} \right) \quad (4.3)$$

with $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \dots, k(\mathbf{x}_N, \mathbf{x}_*)]^\top$. Using Equation (4.1), the conditional posterior prediction is given by [Bishop, 2006, Chapter 2.3.2 and 2.3.3]:

$$p(f(\mathbf{x}_*) | \mathbf{y}, \mathbf{X}, \mathbf{x}_*) = \mathcal{N}_N(f(\mathbf{x}_*) | \mu_*, \sigma_*^2), \quad (4.4)$$

with

$$\mu_* = m(\mathbf{x}_*) + \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 I_N)^{-1} (\mathbf{y} - \boldsymbol{\mu}), \quad (4.5)$$

$$\sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 I_N)^{-1} \mathbf{k}_*, \quad (4.6)$$

We can also write:

$$p(y_* | \mathbf{y}, \mathbf{X}, \mathbf{x}_*) = \mathcal{N}_N(y_* | \mu_*, \sigma_*^2),$$

with

$$\mu_* = m(\mathbf{x}_*) + \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 I_N)^{-1} (\mathbf{y} - \boldsymbol{\mu}),$$

$$\sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) + \sigma^2 - \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 I_N)^{-1} \mathbf{k}_*,$$

Given a new input \mathbf{x}_* the prediction is done by taking the *maximum a posteriori (MAP)* [MacKay, 1996] of this predictive distribution. For a Gaussian distribution, the **MAP** is given by the mean of the distribution, i.e., $\hat{y}_* = \mu_*$. Furthermore, the **GP** framework allows to estimate the uncertainty of the prediction through the variance of the posterior distribution σ_*^2 . This variance

does not depend on the output \mathbf{y} but only on the inputs \mathbf{X} and \mathbf{x}_* . The 95% confidence interval is calculated by taking $\mu_* \pm 2\sigma_*^2$.

To illustrate this section, we define the following example of univariate regression:

$$y = \sin(2\pi x) + 10 + \frac{x}{2} + \epsilon \quad (4.7)$$

with $\epsilon \sim \mathcal{N}_1(0, 0.2)$. The noise-free function we would like to find is defined as: $h(x) = \sin(2\pi x) + 10 + \frac{x}{2}$. Figure 4.7 represents the noisy observations and the noise-free function $h(x)$. The observations are defined on the interval $[0, 4]$. We will try to fit these data with a Gaussian Process.

We define $f \sim \mathcal{GP}(m, k)$, a GP over the latent noise-free functions $h(x)$. From the Equation (4.7), we define $m(x) = 10$ and k a periodic covariance function added to a linear function, as illustrated in Figure 4.6g.

Predictions were computed inside and outside the definition interval using Equations (4.5) and (4.6). In addition to the prediction value (μ_*), the 95% confidence interval is provided ($\mu_* \pm 2\sigma_*^2$). For both predictions inside or outside the definition interval, the confidence interval is similar. Indeed, GP seems to be quite good to extrapolate values. Predictions comparison with several regression models are provided in Figure 4.8. Gaussian Process gives a better prediction outside the definition interval than RF with 200 trees or a MLP.

In this example, the mean function and the covariance function were selected with values fitting correctly our model. The choice of a correct covariance function has a huge influence on the prediction. Figure 4.9 represents GP predictions on the interval $[0, 8]$ with different covariance functions: linear, periodic, RBF and periodic added with a linear. It is clear that the periodic covariance function added with a linear covariance function gives the best prediction. It is interesting to note that the confidence interval is very large when the prediction is very far from the true function and narrow when the prediction is near to the true function.

The values of parameters of the mean function and the covariance function were also selected in order to fit correctly the model. Unsuitable values can lead to under fitting or over fitting. Under fitting is when a model did not learn correctly the patterns in the training data. In contrast, a model is over fitted if it performs very well with the training data but is not able to generalize with the test data. Figure 4.10 represents the prediction with different values of the length-scale ℓ from the periodic covariance function. In this example, a too small value for ℓ leads to over fitting and a too big value leads to under fitting. It is quite easy to choose the form of the covariance function because we have a priori knowledge of our data. Otherwise, it is almost impossible to choose correctly the values of the covariance function parameters. It is necessary to learn them, as explained in the following section.

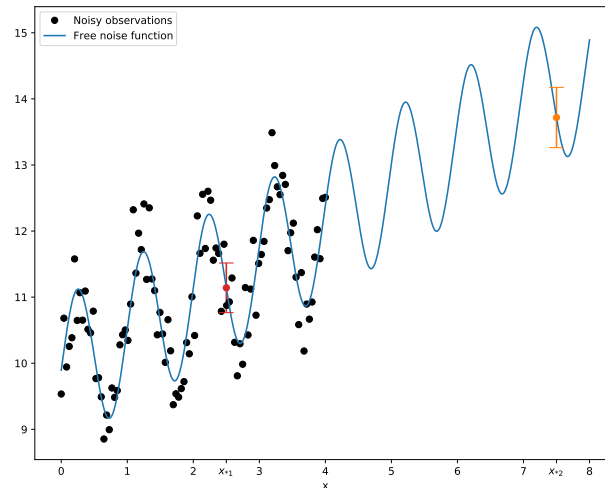


Figure 4.7: Representation of data used for the univariate regression example defined in Equation (4.7). The noisy observations are represented by black circles and the noise-free function $h(x)$ is represented by a blue curve. The observations are defined on the interval $[0, 4]$. The GP is defined with a mean function equal to ten and a covariance function corresponding to a periodic covariance function added to a linear function. The mean prediction of the GP for the input inside the definition interval: $x_{*1} \in [0, 4]$ is represented by a red dot. The mean prediction of the GP for the input outside the definition interval: $x_{*2} \in [4, 8]$ is represented in an orange dot. The error bar corresponds to the 95% confidence interval ($\mu_* \pm 2\sigma_*^2$).

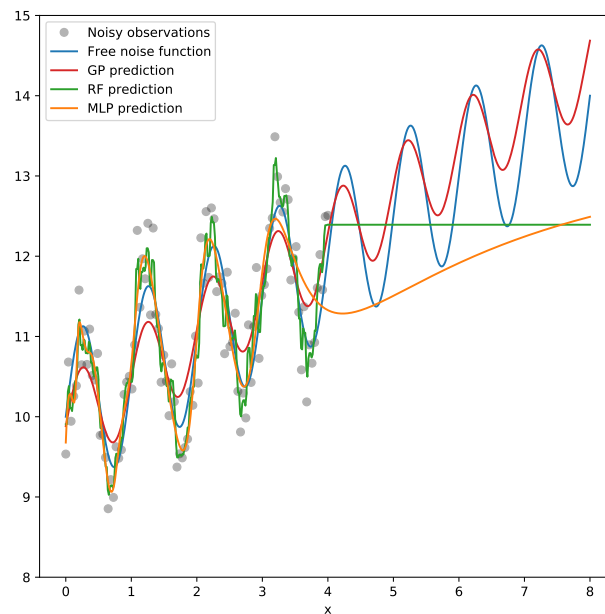


Figure 4.8: Comparison of predictions on the interval $[0, 8]$ with different regressors: Gaussian Process (GP), Random Forest (RF) and Multilayer Perceptron (MLP). The noisy observations are represented by black circles and the noise-free function $g(x)$ is represented by a blue curve. The GP is defined with a mean function equal to ten and a covariance function corresponding to a periodic covariance function added to a linear function. The prediction of the GP is represented by a red curve. A Random Forest with 300 trees and a Multilayer Perceptron with two hidden layers are considered. Their prediction are represented by a green and a yellow curves, respectively.

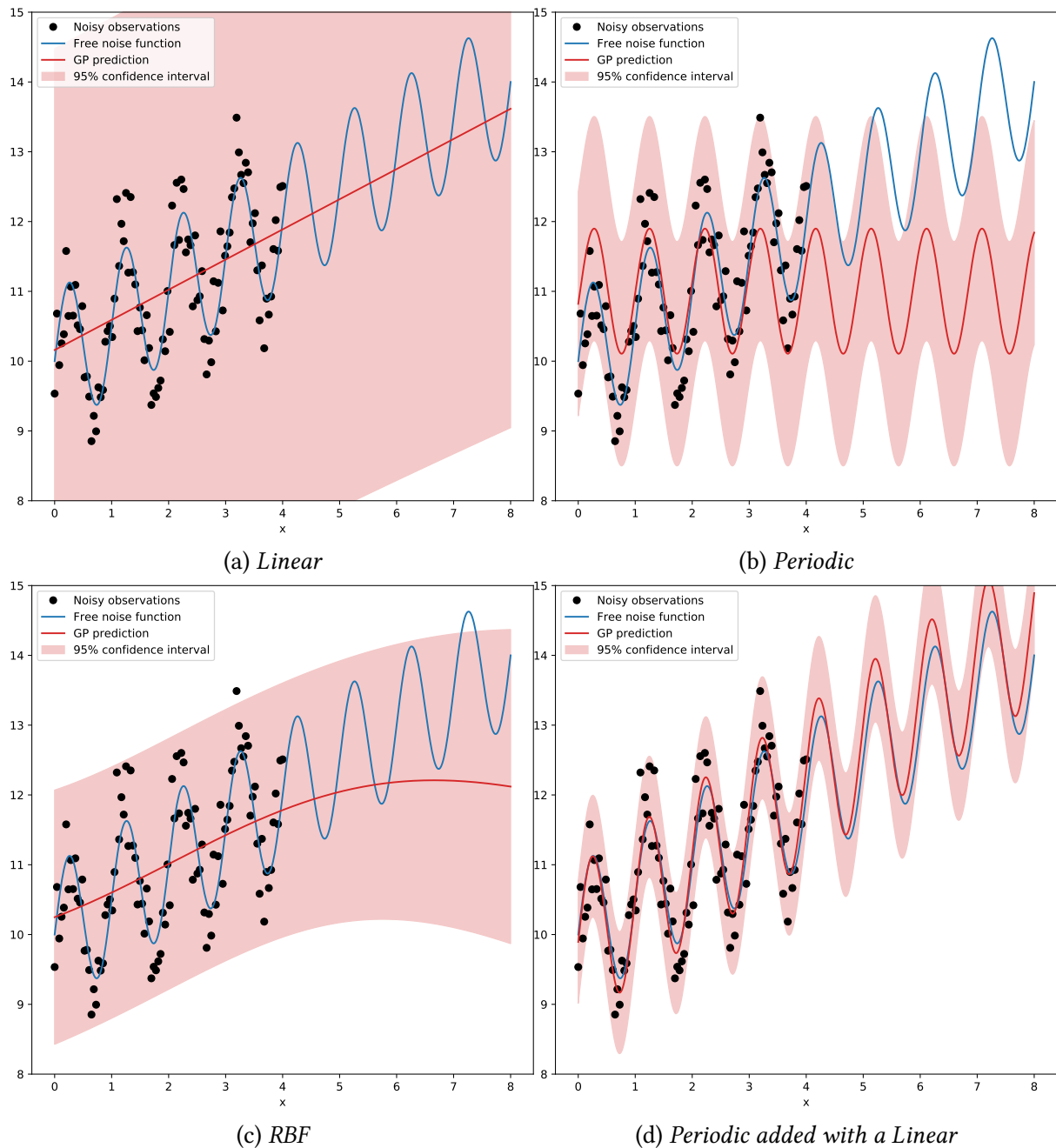


Figure 4.9: Comparison of predictions with different covariance functions: linear, periodic, RBF and periodic added to linear. The noisy observations are represented by black circles and the noise-free function $h(x)$ is represented by a blue curve. The GP is defined with a mean function equal to ten and different covariance functions. The prediction of the GP is represented by a red curve. The red area corresponds to the 95% confidence interval.

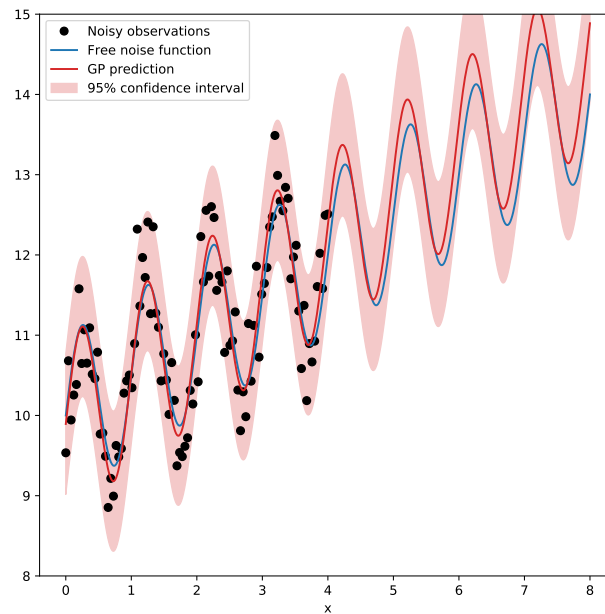
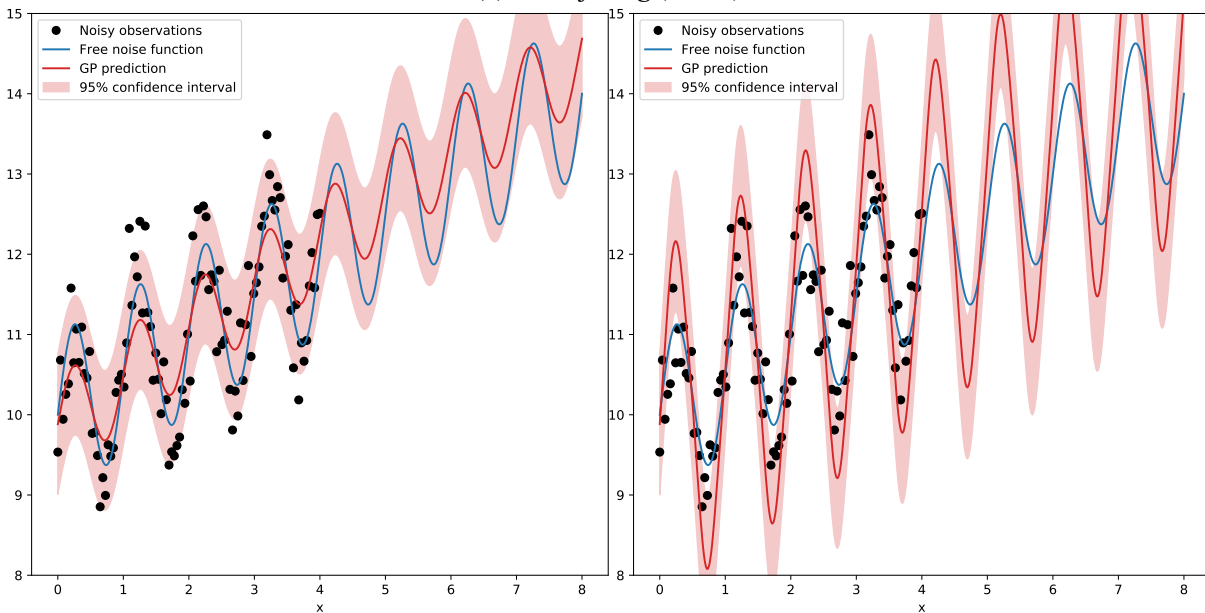
(a) Good fitting ($\ell = 5$)(b) Underfitting ($\ell = 10$)(c) Overfitting ($\ell = 2$)

Figure 4.10: Comparison of predictions with different length-scale values: $\ell = \{2, 5, 10\}$. The noisy observations are represented by black circles and the noise-free function $h(x)$ is represented by a blue curve. The GP is defined with a mean function equal to ten and a covariance function corresponding to a periodic covariance function added to a linear function. The prediction of the GP is represented by a red curve. The red area corresponds to the 95% confidence interval. A too small value for ℓ leads to over fitting: the mean prediction is too close to the noisy observations. A too big value for ℓ leads to under fitting: the mean prediction is far from the noisy observations. By taking the right value for ℓ , the mean prediction is very close to the noise-free function.

Training

The hyper-parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_m, \boldsymbol{\theta}_k, \sigma^2\}$ strongly influence the prediction since they appear in Equations (4.5) and (4.6). Compared with other techniques, such as SVMs, hyper-parameters can be optimized. They are not fixed based on expert knowledge or found by cross-validation, but are learned. Therefore, they can be called parameters. They are usually optimized by maximizing the log-marginal likelihood of the model on the training set \mathcal{S} [Rasmussen and Williams, 2005, Chapter 2]:

$$\arg \max_{\boldsymbol{\theta}} \log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}). \quad (4.8)$$

The marginal likelihood corresponds to the integral of the likelihood times the prior, defined as:

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \int p(\mathbf{y}|f(\mathbf{X}), \boldsymbol{\theta})p(f|\mathbf{X}, \boldsymbol{\theta})df. \quad (4.9)$$

The likelihood can be written as

$$p(y_i|f(\mathbf{x}_i)) = \mathcal{N}_1(y_i|f(\mathbf{x}_i), \sigma^2). \quad (4.10)$$

Assuming i.i.d. samples, the full likelihood can be factorized and is given by

$$p(\mathbf{y}|f(\mathbf{X})) = \prod_{i=1}^N p(y_i|f(\mathbf{x}_i)) = \mathcal{N}_N(\mathbf{y}|f(\mathbf{X}), \sigma^2 I_N) \quad (4.11)$$

and the prior is written as $p(f|\mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}_N(\boldsymbol{\mu}, \mathbf{K})$. Thus, the log marginal likelihood is defined as:

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) &= \log \left(\int \mathcal{N}_N(\mathbf{y}|f(\mathbf{X}), \sigma^2 I_N) \mathcal{N}_N(\boldsymbol{\mu}, \mathbf{K}) \right) \\ &= -\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^\top (\mathbf{K} + \sigma^2 I_N)^{-1} (\mathbf{y} - \boldsymbol{\mu}) \\ &\quad - \frac{1}{2} \log (|\mathbf{K} + \sigma^2 I_N|) - \frac{N}{2} \log(2\pi). \end{aligned} \quad (4.12)$$

The derivatives of Equation (4.12) are analytically tractable and the optimization of $\boldsymbol{\theta}$ can be done using constrained gradient descent [Rasmussen and Williams, 2005, Chapter 5 and Appendix A.3]. In comparison to other non-linear prediction algorithms, such as SVM or kernel ridge regression, GP offer the possibility to automatically tune their hyper-parameters $\boldsymbol{\theta}$.

Complexity

GP scale poorly w.r.t. the number of training samples N . The main bottleneck comes from the computational cost of the matrix inversion $(\mathbf{K} + \sigma^2 I_N)^{-1}$ and the computation of the determinant $(|\mathbf{K} + \sigma^2 I_N|)$ in Equation (4.12). These operations scale cubically with the number of training pixels, $\mathcal{O}(N^3)$. Moreover, the storage complexity is $\mathcal{O}(N^2)$. That is why GP are limited to around 10 000 points. Approximation methods will be discussed in Section 4.4.

To conclude, GP have very interesting properties. Such as SVM, by using a proper kernel function, they can take into account prior information of the data. Besides, they have advantages over SVM as they provide probabilistic outputs. Moreover, the hyper-parameters can be tuned by gradient descent. For all these reasons, univariate GP regression was widely used in the remote sensing community. They were successfully applied for biophysical parameter estimation (e.g. chlorophyll, LAI, etc.) [Furfaro et al., 2006], [Pasolli et al., 2010], [Verrelst et al., 2011], [Bazi et al., 2012], [Verrelst et al., 2012b], [Verrelst et al., 2012a], [Verrelst et al., 2013]. However, due to complexity issues, all these works were applied to small data sets, a few thousand pixels [Camps-Valls et al., 2016]. In the following, we will focus on the binary classification case.

4.2.3. Binary Classification

In the case of binary classification, univariate GP are used to predict a discrete target value ($y_i \in \{0, 1\}$) from an input value \mathbf{x}_i . The target y_i follows a Bernoulli distribution: it takes the value 1 with probability p and the value 0 with probability $1 - p$. A logistic function σ is used as link function to relate the univariate GP and the probability p :

$$p = p(y_i = 1 | f(\mathbf{x}_i)) = \sigma(f(\mathbf{x}_i)). \quad (4.13)$$

Different logistic functions can be used such as the sigmoid function:

$$\sigma(f(\mathbf{x}_i)) = \frac{1}{1 + \exp(-f(\mathbf{x}_i))}. \quad (4.14)$$

The targets are Bernoulli distributed and independent random variables, thus the full likelihood can be written as [Nickisch and Rasmussen, 2008]:

$$p(\mathbf{y} | f(\mathbf{X})) = \prod_{i=1}^N p(y_i | f(\mathbf{x}_i)) = \prod_{i=1}^N \sigma(f(\mathbf{x}_i)). \quad (4.15)$$

Contrary to the univariate regression case, the posterior is non Gaussian due to the non Gaussian likelihood in Equation (4.15). Therefore, analytic expressions of the marginal and predictive distributions are not available explicitly. Different solutions are proposed either based on sampling algorithms or based on Gaussian approximations of the posterior [Nickisch and Rasmussen, 2008]. Sampling methods, such as Markov Chain Monte Carlo (MCMC) [Neal, 1997], provide exact computation but at prohibitive computational costs. Concerning Gaussian approximations, two different approximation methods are usually used: Expectation Propagation (EP) [Minka, 2001] and Laplace Approximation (LA) [Williams and Barber, 1998], [Rasmussen and Williams, 2005].

In remote sensing, Bazi *et al.* [Bazi and Melgani, 2008], [Bazi and Melgani, 2010] studied both approximation methods, EP and LA, with mono date hyper-spectral and multi-spectral data sets. Both methods show comparable classification accuracies. However, LA required less computational time than EP. For these reasons, LA was applied for hyper-spectral datasets in [Yang et al., 2015b]. Like conventional GP, approximated GP are still limited to a few thousand of training inputs.

4.3. Multivariate Gaussian Processes

Section 4.2 described univariate GP that are used for univariate univariable regression (i.e. $x \in \mathbb{R}$ and $y \in \mathbb{R}$), univariate multivariable regression (i.e. $\mathbf{x} \in \mathbb{R}^D$ and $y \in \mathbb{R}$) or binary classification (i.e. $x \in \mathbb{R}$ or $\mathbf{x} \in \mathbb{R}^D$ and $y \in \{0, 1\}$). In this section, we will present multivariate GP are also known as multi-output or multi-task GP [Bonilla et al., 2007]. They are used to regress multivariate variables, such as multi-output regression ($x \in \mathbb{R}$ or $\mathbf{x} \in \mathbb{R}^D$ and $\mathbf{y} \in \mathbb{R}^P$) (c.f. Section 4.3.2) or multi-class classification ($x \in \mathbb{R}$ or $\mathbf{x} \in \mathbb{R}^D$ and $\mathbf{y} \in \{0, 1\}^C$) (c.f. Section 4.3.3).

4.3.1. Definition

Like univariate GP, a P -variate GP \mathbf{f} , with $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), \dots, f_p(\mathbf{x}), \dots, f_P(\mathbf{x})]^\top$, is specified by its vector-valued mean function $\mathbf{m} \in \mathbb{R}^P$ and its positive definite matrix-valued covariance function $\mathcal{K} \in \mathbb{R}^{P \times P}$. We have $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$ with:

$$\mathbf{m}(\mathbf{x}) = [m_1(\mathbf{x}) \quad \dots \quad m_P(\mathbf{x})]^\top,$$

$$\mathcal{K}(\mathbf{x}, \mathbf{x}) = \begin{bmatrix} k_{11}(\mathbf{x}, \mathbf{x}) & \dots & k_{1P}(\mathbf{x}, \mathbf{x}) \\ \dots & k_{pp'}(\mathbf{x}, \mathbf{x}) & \dots \\ k_{P1}(\mathbf{x}, \mathbf{x}) & \dots & k_{PP}(\mathbf{x}, \mathbf{x}) \end{bmatrix},$$

where $k_{pp'}(\mathbf{x}, \mathbf{x})$ is the covariance between two univariate GP: $f_p(\mathbf{x})$ and $f_{p'}(\mathbf{x})$ with $p, p' \in \{1, \dots, P\}$ and \mathcal{K} of size $P \times P$. Similarly to univariate GP, all marginals follow a Gaussian distribution, noting

$$\mathbf{f}(\mathbf{X}) = [f_1(\mathbf{x}_1), \dots, f_P(\mathbf{x}_1), \dots, f_1(\mathbf{x}_N), \dots, f_P(\mathbf{x}_N)]^\top$$

the random vector of size NP , then $\mathbf{f}(\mathbf{X}) \sim \mathcal{N}_{NP}(\boldsymbol{\mu}_o, \mathbf{K}_o)$ with $\boldsymbol{\mu}_o = [\mathbf{m}(\mathbf{x}_1), \dots, \mathbf{m}(\mathbf{x}_N)]^\top$ and

$$\mathbf{K}_o = \begin{bmatrix} \mathcal{K}(\mathbf{x}_1, \mathbf{x}_1) & \dots & \mathcal{K}(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ \mathcal{K}(\mathbf{x}_N, \mathbf{x}_1) & \dots & \mathcal{K}(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}.$$

In the following, the construction of the mean and covariance functions is described.

Mean function

As the univariate GP, in ML, the mean function has little influence and all the prior information is given by the covariance function. Different vector-valued mean functions \mathbf{m} can be used. Usually, a vector of constant values is used such as: $\mathbf{m} = [b_1 \quad \dots \quad b_P]^\top$ with b_p different constant values and $\mathbf{m} \in \mathbb{R}^P$.

Covariance function

In the matrix-valued covariance function \mathcal{K} , the term $k_{pp'}$ corresponds to the cross covariance function between two univariate GP f_p and $f_{p'}$. The main challenge is to build and optimize this cross-covariance function $k_{pp'}$ that:

- leads to a valid covariance function \mathcal{K} (positive semi-definite),
- exploits the multivariate structure of the problem (e.g. exploits the correlation between variables),
- can lead to efficient computation because \mathbf{K}_o is of size $NP \times NP$.

Independent Gaussian Processes: A simple approach is to use independent GP. All the diagonal terms of \mathcal{K} are equal to the covariance function of the univariate GP ($k_{pp} = k_p$ with $f_p \sim \mathcal{GP}(m_p, k_p)$) and all the off-diagonal terms of \mathcal{K} are set to zero such as:

$$\mathcal{K}(\mathbf{x}, \mathbf{x}) = \begin{bmatrix} k_1(\mathbf{x}, \mathbf{x}) & \dots & 0 \\ \dots & k_p(\mathbf{x}, \mathbf{x}) & \dots \\ 0 & \dots & k_P(\mathbf{x}, \mathbf{x}) \end{bmatrix} \quad (4.16)$$

This corresponds to a diagonal matrix which is definite semi-positive and thus a valid covariance function. However, with independent GP, it is not possible to capture any cross-correlation between the outputs.

Separable kernels: A common approach is to consider separable kernels where one kernel, $k(\mathbf{x}, \mathbf{x}')$, acts on the input sample and another kernel, $k_T(p, p')$, models the interaction between the outputs, as defined in the following equation [Álvarez et al., 2012]:

$$k_{pp'}(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}')k_T(p, p'). \quad (4.17)$$

The **Linear Model of Co-regionalization (LMC)** exploits the formulation defined in Equation (4.17) [Journal and Huijbregts, 1976], [Goovaerts and Goovaerts, 1997]. In LMC, Alvarez et al. [Álvarez et al., 2012] defined each marginal f_p as

$$f_p(\mathbf{x}) = \sum_{l=1}^L \sum_{r=1}^R a_{p,l}^r g_l^r(\mathbf{x}), \quad (4.18)$$

with L groups of R latent functions g_l^r where g_l^r is an univariate GP such as $g_l^r \sim \mathcal{GP}(m_l, k_l)$ and $a_{p,l}^r$ is a scalar coefficient. Figure 4.11 represents the LMC configuration for P marginal GP, f_p , combined linearly with $L \times R$ latent univariate GP, g_l^r , as defined in Equation (4.18). From Equation (4.17), the kernel can be rewritten as

$$k_{pp'}(\mathbf{x}, \mathbf{x}') = \sum_{l=1}^L b_{p,p'}^l k_l(\mathbf{x}, \mathbf{x}') \quad (4.19)$$

with $b_{p,p'}^l = \sum_{r=1}^R a_{p,l}^r a_{p',l}^r$.

The **Intrinsic Co-regionalization Model (ICM)** is a simplified version of **LMC** [Goovaerts, 1997] where $L = 1$ and $R \neq 1$. In this case, the R latent functions share the same covariance function k_1 but they are independent from each other for $r \neq r'$. The **Semiparametric Latent Factor Model (SLFM)** is also a simplified version of **LMC** [Teh et al., 2005] where $R = 1$ and $L \neq 1$. In this case, the L latent functions have different covariance functions k_l and are independent from each other for $l \neq l'$. We can write:

$$f_p(\mathbf{x}) = \sum_{l=1}^L a_{p,l} g_l(\mathbf{x}) \quad (4.20)$$

or in matrix form, with $\mathbf{f} = [f_1(\mathbf{X}), \dots, f_P(\mathbf{X})]^\top$:

$$\mathbf{f} = \mathbf{A}\mathbf{g} \quad (4.21)$$

with $\mathbf{A} \in \mathbb{R}^{P \times L}$.

Many multivariate **GP** models from the literature are particular cases of the **LMC**, see for instance [Durrande et al., 2010], [Álvarez et al., 2012]. Unlike with independent **GP**, with **LMC**, the cross-correlation between the outputs are taken into account.

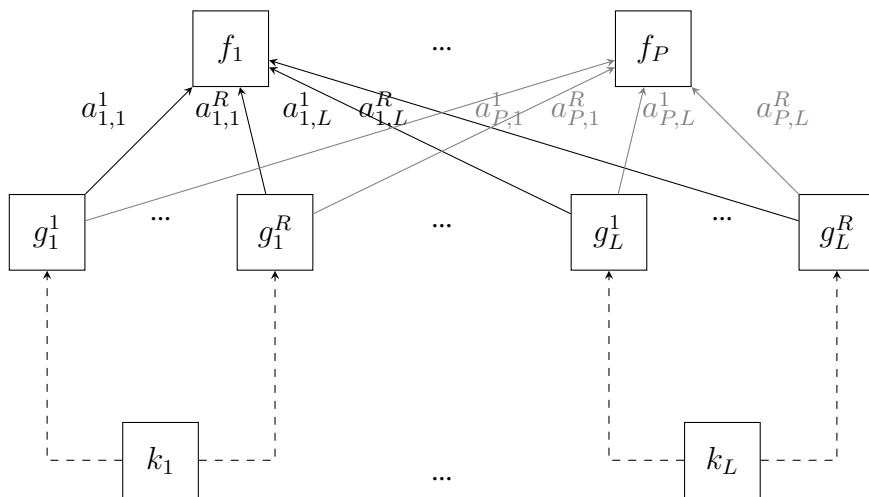


Figure 4.11: *LMC configuration for P marginal GP, f_p , combined with L groups of R latent functions g_l^r . Inside each l group, all the latent functions $\{g_l^r\}_{r=1}^R$ share the same covariance function k_l but are independent from other latent functions for $r \neq r'$. Each latent function g_l has different covariance function for $l \neq l'$. All the g_l^r latent univariate GP are combined linearly to form the f_p marginal GP as described in Equation (4.18).*

Non-separable kernels: Another common approach is to remove the separable assumption, with non separable-kernels, by using convolution processes [Higdon, 2002], [Boyle and Frean, 2004], [van der Wilk et al., 2017]. Convolution processes can capture more dependence between outputs than **LMC** (e.g. translation between outputs), but they lack a formulation that scales well with the number of training samples N [van der Wilk et al., 2020].

4.3.2. Multi-output regression

In the case of multi-output regression, P scalar target values ($\mathbf{y}_i \in \mathbb{R}^P, \mathbf{y}_i = [y_1^i, \dots, y_P^i]$) are regressed from the input \mathbf{x}_i . We denote $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]^\top$. We observed a noisy version of each function such as

$$y_p^i = \psi(f_p(\mathbf{x}_i)) = f_p(\mathbf{x}_i) + \epsilon_p \quad (4.22)$$

with $\{\epsilon_p\}_{p=1}^P$ independent white noises with variance σ_p^2 . We have:

$$\mathbf{y}_i = \psi(\mathbf{f}(\mathbf{x}_i)) = \mathbf{f}(\mathbf{x}_i) + \epsilon \quad (4.23)$$

with $\epsilon \sim \mathcal{N}(0, \Sigma)$ with $\Sigma \in \mathbb{R}^{P \times P}$ a diagonal matrix with element $\{\sigma_p^2\}_{p=1}^P$. In the case of independent latent GP, we have:

$$\mathbf{y}_i = I_p \mathbf{g}(\mathbf{x}_i) + \epsilon.$$

In the case of the **SLFM** (i.e. the simplified version of **LMC**), we have:

$$\mathbf{y}_i = \mathbf{A} \mathbf{g}(\mathbf{x}_i) + \epsilon.$$

For one pixel \mathbf{x}_i , the likelihood is given by

$$p(\mathbf{y}_i | \mathbf{f}(\mathbf{x}_i)) = \mathcal{N}_P(\mathbf{y}_i | \mathbf{m}(\mathbf{x}_i), \mathcal{K}(\mathbf{x}_i, \mathbf{x}_i) + \Sigma). \quad (4.24)$$

with $\mathbf{f}(\mathbf{x}_i) = \mathbf{A} \mathbf{g}(\mathbf{x}_i)$ or $\mathbf{f}(\mathbf{x}_i) = I_p \mathbf{g}(\mathbf{x}_i)$.

Assuming i.i.d. samples, the full likelihood is given by

$$p(\mathbf{Y} | \mathbf{f}(\mathbf{X})) = \prod_{i=1}^N p(\mathbf{y}_i | \mathbf{f}(\mathbf{x}_i)) = \mathcal{N}_{NP}(\mathbf{Y} | \boldsymbol{\mu}_o, \mathbf{K}_o + \Sigma \otimes I_N). \quad (4.25)$$

with \otimes the Kronecker product.

Inference

As for the univariate case, in the multivariate regression, the predictive distribution and the marginal likelihood can be derived analytically. The predictive distribution for a new input \mathbf{x}_* is

$$p(\mathbf{f}(\mathbf{x}_*) | \mathbf{Y}, \mathbf{X}, \mathbf{x}_*) = \mathcal{N}_{NP}(\mathbf{f}(\mathbf{x}_*) | \boldsymbol{\mu}_*, \mathbf{K}_*), \quad (4.26)$$

with

$$\boldsymbol{\mu}_* = \mathbf{m}(\mathbf{x}_*) + \mathcal{K}_*^\top (\mathbf{K}_o + \Sigma \otimes I_N)^{-1} (\mathbf{Y} - \boldsymbol{\mu}_o), \quad (4.27)$$

$$\mathbf{K}_* = \mathcal{K}(\mathbf{x}_*, \mathbf{x}_*) - \mathcal{K}_*^\top (\mathbf{K}_o + \Sigma \otimes I_N)^{-1} \mathcal{K}_*, \quad (4.28)$$

and with $\mathcal{K}_* \in \mathbb{R}^{P \times NP}$, $\mathcal{K}_* = [\mathcal{K}(\mathbf{x}_*, \mathbf{x}_1), \dots, \mathcal{K}(\mathbf{x}_*, \mathbf{x}_N)]^\top$. We can also write:

$$p(\mathbf{y}_* | \mathbf{Y}, \mathbf{X}, \mathbf{x}_*) = \mathcal{N}_{NP}(\mathbf{y}_* | \boldsymbol{\mu}_*, \mathbf{K}_*),$$

with

$$\begin{aligned}\boldsymbol{\mu}_* &= \mathbf{m}(\mathbf{x}_*) + \mathcal{K}_*^\top (\mathbf{K}_o + \boldsymbol{\Sigma} \otimes I_N)^{-1} (\mathbf{Y} - \boldsymbol{\mu}_o), \\ \mathbf{K}_* &= \mathcal{K}(\mathbf{x}_*, \mathbf{x}_*) + \boldsymbol{\Sigma} - \mathcal{K}_*^\top (\mathbf{K}_o + \boldsymbol{\Sigma} \otimes I_N)^{-1} \mathcal{K}_*,\end{aligned}$$

To illustrate this section, we define an example of multi-output regression. The multi-output regression is described as the regression of two scalar values y_1 and y_2 from an input in one dimension ($\mathbf{x} = x \in \mathbb{R}$). The outputs are defined by the following function

$$y_1 = h_1(x) + \epsilon_1, \quad y_2 = h_2(x) + \epsilon_2 \quad (4.29)$$

with the following noise-free functions:

$$\begin{aligned}h_1(x) &= 1.5(x + 2.5) \times \sqrt{((6x - 2)^2 \times \sin(12x - 4) + 10)} \\ h_2(x) &= (6x - 2)^2 \times \sin(12x - 4) + 10.\end{aligned}$$

They are defined for $x \in [0, 1]$ and with $\epsilon_1, \epsilon_2 \sim \mathcal{N}_1(0, 0.2)$. Figure 4.12 represents the noisy observations y_1 and y_2 and the noise-free function $h_1(x)$ and $h_2(x)$. h_1 is a nonlinear transformation of h_2 , they are highly correlated, the Pearson correlation coefficient is $r = 0.95$ [Liu et al., 2018b].

We define $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$, a GP over the latent noise-free functions $h_1(x)$ and $h_2(x)$. In this example, two different methods were used to build the cross-covariance function k_{pp} : independent GP and simplified version of LMC. Figure 4.13 represents the comparisons of predictions with these two methods. With the independent GP configuration, the Root Mean Squared Error (RMSE) is equal to 1.47 for the first output y_1 and is equal to 1.90 for the second output y_2 . With the simplified LMC configuration, the performances are increased: RMSE is equal to 0.49 for y_1 and is equal to 0.72 for y_2 . By taking into account the correlation between the outputs y_1 and y_2 with the simplified version of LMC, the performances are increased for the predictions on both outputs.

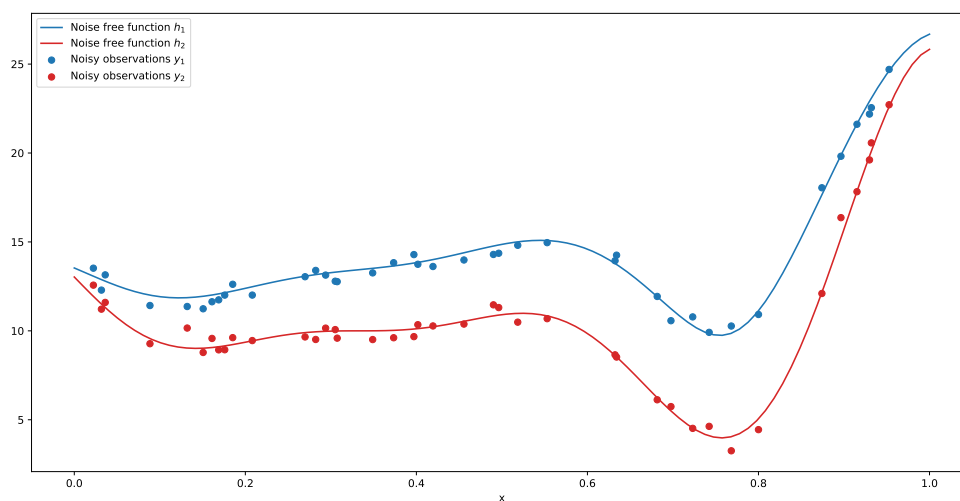
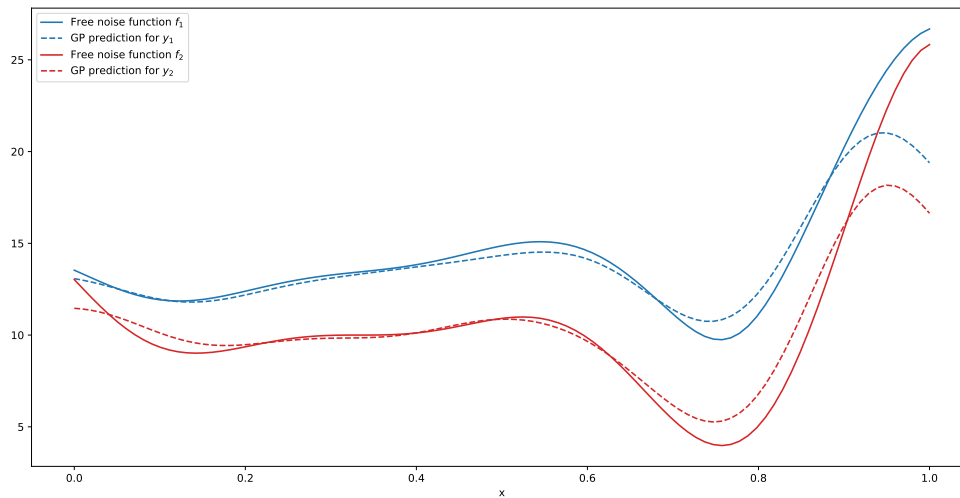
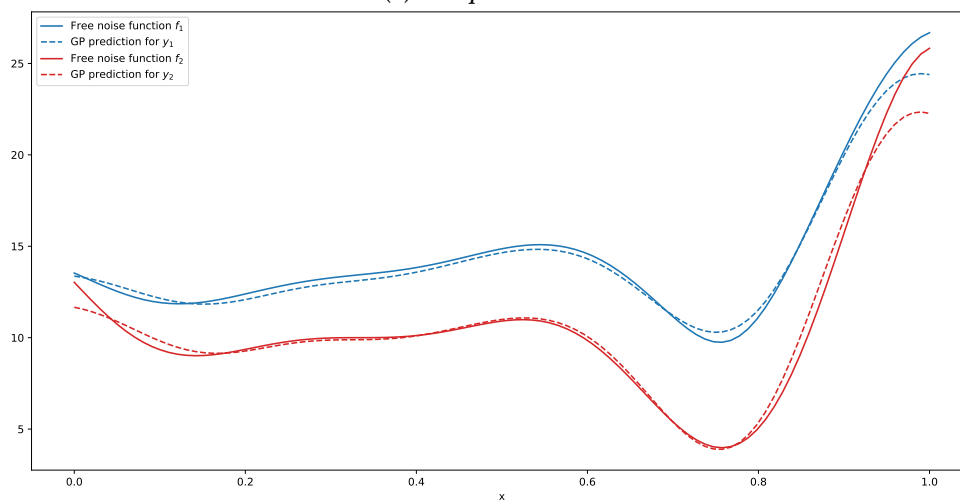


Figure 4.12: Representation of data used for the multi-output regression example defined in Equation (4.29). The noisy observations are represented by circles, blue circles for y_1 and red circles for y_2 . The noise-free functions are represented by curves, blue curve for h_1 and red curve for h_2 .



(a) Independent GP



(b) LMC

Figure 4.13: Comparison of predictions with two configurations of covariance functions: covariance function computed with independent GP and covariance function computed with simplified version of LMC. The noise-free functions are represented by curves, blue curve for h_1 and red curve for h_2 . The latent GP are defined with a constant mean function and with a RBF covariance function. The mean prediction of the GP for respectively, the output y_1 and y_2 are represented by blue and red dashed lines, respectively.

Training

In the multivariate case, the log marginal likelihood is very similar to the univariate case and is defined as:

$$\begin{aligned} \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}) = & -\frac{1}{2}(\mathbf{Y} - \boldsymbol{\mu}_o)^\top (\mathbf{K}_o + \boldsymbol{\Sigma} \otimes I_N)^{-1} (\mathbf{Y} - \boldsymbol{\mu}_o) \\ & -\frac{1}{2} \log (|\mathbf{K}_o + \boldsymbol{\Sigma} \otimes I_N|) - \frac{N}{2} \log(2\pi). \end{aligned} \quad (4.30)$$

As defined in Section 4.2.2, the hyper-parameters $\boldsymbol{\theta}$ are usually optimized by maximizing the log-marginal likelihood of the model on the training set \mathcal{S} .

Complexity

\mathbf{K}_o is of size $NP \times NP$, thus, in the worst case, the complexity is $\mathcal{O}((PN)^3)$ and the associated storage complexity is $\mathcal{O}((PN)^2)$. For independent GP, if we simplify by taking the same covariance function, we have: $\mathcal{K}(\mathbf{x}, \mathbf{x}) = k(\mathbf{x}, \mathbf{x})I_p$, thus \mathbf{K}_o is defined as a block diagonal. In particular if the input points are the same, all the blocks are equal and the problem reduces to invert a matrix of size $N \times N$. Moreover, the use of separable kernels reduces the computational complexity [Baldassarre et al., 2012]. Indeed, the LMC configuration permits to reduce the complexity from $\mathcal{O}((NP)^3)$ to $\mathcal{O}(P^3) + \mathcal{O}(N^3)$ [Álvarez et al., 2012]. More recently, efficient optimization procedures were proposed in the literature [Wilson et al., 2016], [Moreno-Muñoz et al., 2018], [van der Wilk et al., 2020]. However, if N is huge, the complexity remains high and approximation methods are needed, as described in Section 4.4.

In remote sensing, LMC was used to regress biophysical variables in [Mateo-Sanchis et al., 2018] using MODIS time-series. SLFM, a simplified version of LMC, was used to produce cloud-free LAI and Radar Vegetation Index (RVI) from optical and radar time series [Pipia et al., 2019]. Convolution processes were used to produce cloud-free fusion of Sentinel-1 and Sentinel-2 time series [Caballero et al., 2023]. Independent GP were used for interpolated multiple time series from the CubeSats [Ruan et al., 2017]. All these works were applied to small data sets or to small study areas. We want to use these methods for large datasets, approximation methods are required as described in Section 4.4.

4.3.3. Multi-class classification

In the case of classification with C classes, the target is such as $\mathbf{y}_i \in \{0, 1\}^C$ with all its values set to zero except for the element $y_{ic} = 1$ for \mathbf{x}_i of class c . In the classification case, we denote $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^\top$ and $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]^\top$. A softmax function $\boldsymbol{\sigma}$ is used as link function to relate the multivariate latent variable $\mathbf{f}(\mathbf{x}_i) = [f_1(\mathbf{x}_i), \dots, f_C(\mathbf{x}_i)]^\top$ and the observation \mathbf{y}_i :

$$\begin{aligned} \mathbf{y}_i = & \boldsymbol{\sigma}(\mathbf{f}(\mathbf{x}_i)) \\ = & \frac{1}{\sum_{c'=1}^C \exp(f_{c'}(\mathbf{x}_i))} \times \begin{bmatrix} \exp(f_1(\mathbf{x}_i)) \\ \vdots \\ \exp(f_C(\mathbf{x}_i)) \end{bmatrix}. \end{aligned} \quad (4.31)$$

The associated likelihood for the sample i is written:

$$\begin{aligned} p(\mathbf{y}_i | \mathbf{f}(\mathbf{x}_i)) &= \prod_{c=1}^C \left[\frac{\exp(f_c(\mathbf{x}_i))}{\sum_{c'=1}^C \exp(f_{c'}(\mathbf{x}_i))} \right]^{y_{ic}} \\ &= \frac{\exp(\mathbf{y}_i^\top \mathbf{f}(\mathbf{x}_i))}{\sum_{c'=1}^C \exp(f_{c'}(\mathbf{x}_i))}, \end{aligned} \quad (4.32)$$

Such as in binary classification defined in Section 4.2.3, the likelihood in Equation (4.32) is not conjugate to the Gaussian distribution and thus analytic expressions of the marginal and predictive distributions are not available. As defined in Section 4.2.3, sampling methods (i.e. MCMC) or approximations methods (i.e. EP or LA) can be used to deal with this non Gaussian likelihood [Villacampa-Calvo et al., 2021].

In remote sensing, Constantin *et al.* [Constantin et al., 2021], [Constantin et al., 2022] proposed to use a mixture of independent multivariate Gaussian Processes for land cover classification from Sentinel-2 time-series. The authors proposed a particular LMC configuration in order to separately model the temporal and spectral correlation. This constrained spectro-temporal structure permits to reduce the complexity of the model and to apply the model for large scale applications.

Like in multi-output regression, the LMC configuration in multi-class permits to reduce the complexity from $\mathcal{O}((NC)^3)$ to $\mathcal{O}(C^3) + \mathcal{O}(N^3)$ [Álvarez et al., 2012]. However, if N is huge, the complexity remains high and approximation methods are needed. In the following section, advances that alleviate the computational cost of GP are presented.

4.4. Large scale Gaussian Processes

For simplification, in the following, we will focus on univariate GP regression defined in Section 4.2.2. In training, as defined in Equation (4.12), we need to calculate the inverse and the determinant of the kernel matrix of size $N \times N$. It leads to a huge computational complexity: $\mathcal{O}(N^3)$ and also to an important storage complexity: $\mathcal{O}(N^2)$. In prediction, as defined in Equations (4.5) and (4.6), only the inverse of the kernel matrix needs to be calculated. Figure 4.14 represents the computation time of the log-marginal likelihood from the example defined in Equation (4.7) for several amounts of training samples. After a certain number of samples, the calculation is no longer possible. It is clear that this time is increasing significantly. Two different approaches are usually used to reduce the computational cost of GP: model approximation and posterior approximation [van der Wilk et al., 2020]. These two approaches are presented in the following.

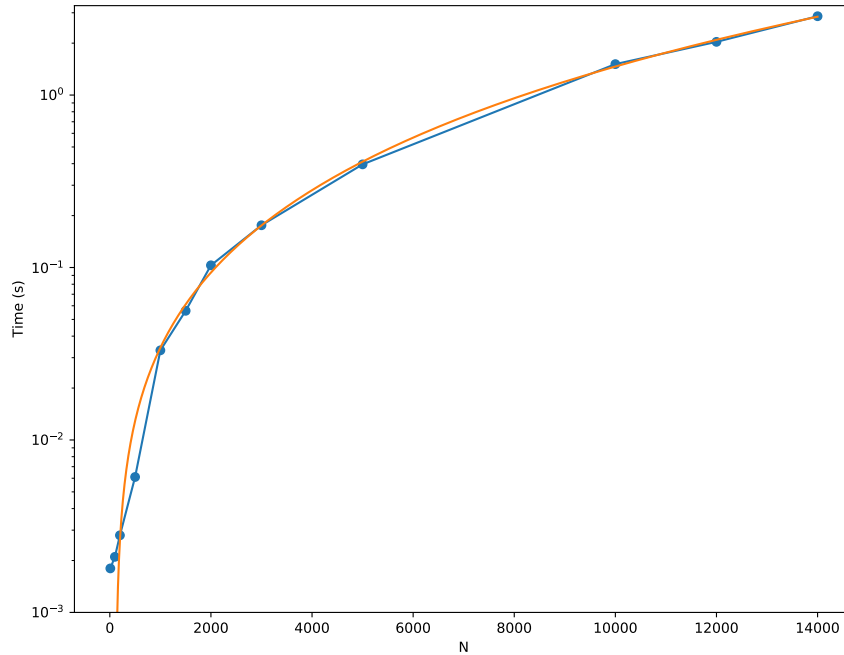


Figure 4.14: The blue curve correspond to the computation times of the log-marginal likelihood (Equation (4.12)) for several number of training samples: $N \in \{10, 100, 200, 500, 1000, 1500, 2000, 3000, 5000, 10000, 12000, 14000\}$ from the univariate regression example defined in Equation (4.7). For more than 15 000 samples, it is no longer possible to calculate the log-marginal likelihood, i.e. memory problem. The function `gpytorch.mlls.ExactMarginalLogLikelihood` from the library `Gpytorch` was used to compute the log-marginal likelihood with a laptop (8 CPU cores with 15GB of RAM). The orange curve was computed using the function `polyfit` from `Scipy`. It corresponds to a polynomial function of degree 3 such as: $1.6 \times 10^{-13}N^3 + 1.03 \times 10^{-8}N^2 + 2.7 \times 10^{-5}N - 3.2 \times 10^{-3}$. The computational time is exponential with the number of training samples.

4.4.1. Model Approximation

Approximation of a Gaussian process model consists in reducing the computational complexity when computing the prior $p(f(\mathbf{X}))$ or the joint prior $p(f(\mathbf{x}_*)|f(\mathbf{X}))$ [Quiñonero-Candela and Rasmussen, 2005]. The main idea is to reduce the complexity of the calculation of the inverse and the determinant of the kernel matrix of size $N \times N$.

Kronecker methods

The data structure can be taken into account to speed-up the inversion of \mathbf{K} , such as in [Saatci, 2011, Chapter 5] and [Wilson et al., 2014] where \mathbf{K} is decomposed into a Kronecker product of smaller matrices such as

$$\mathbf{K} = \mathbf{K}_1 \otimes \dots \otimes \mathbf{K}_D \quad (4.33)$$

with D the number of matrices.

Noting \mathbf{K}_1 a square matrix of size $n \times n$ and \mathbf{K}_2 a square matrix of size $m \times m$. The Kronecker product of these two matrices $\mathbf{K}_1 \otimes \mathbf{K}_2$ is of size $mn \times mn$. The computation of the inverse of a matrix of size $mn \times mn$ becomes impossible to do efficiently even if m and n

are not too big. One interesting property is that the inverse of the Kronecker product of two matrices is the Kronecker product of the inverse of the matrices:

$$(\mathbf{K}_1 \otimes \mathbf{K}_2)^{-1} = \mathbf{K}_1^{-1} \otimes \mathbf{K}_2^{-1}.$$

Taking the inverse of two matrices: one $n \times n$ and another $m \times m$ is much easier than inverting a big matrix of size $mn \times mn$. Moreover, noting \mathbf{K}_1 a square matrix of size $n \times n$ and \mathbf{K}_2 a square matrix of size $m \times m$, we have:

$$\det(\mathbf{K}_1 \otimes \mathbf{K}_2) = \det(\mathbf{K}_1)^m \times \det(\mathbf{K}_2)^n.$$

This property simplifies the calculation of the determinant.

Decomposing the matrix \mathbf{K} of size $N \times N$ with the properties of the Kronecker product still involves operations in $\mathcal{O}(N)$. Using a large number of training inputs N still involves high computation complexity. By using the Kronecker product, the correlation between features can be lost, as they are treated independently. Indeed, by using a Kronecker product, Constantin *et al.* [Constantin et al., 2021] lost the correlation between spectral and temporal features.

Random projection

Random projection involves mapping the high-dimensional input space onto a lower-dimensional subspace using a random matrix \mathbf{B} [Pérez-Suay et al., 2017], such as

$$\mathbf{K} \times \mathbf{B} = \mathbf{K}' \quad (4.34)$$

with \mathbf{K} of size $N \times N$, \mathbf{B} of size $N \times M$ and \mathbf{K}' of size $N \times M$ with $M \ll N$. The values for the random projection matrix \mathbf{B} are drawn i.i.d. from a Gaussian distribution.

In remote sensing, random projections were mainly used with SVM on hyperspectral images [Li et al., 2013], [Menon et al., 2016]. We did not find works with Gaussian Processes.

Nyström approximation

The main idea is to approximate the covariance matrix \mathbf{K} of size $N \times N$ by $\tilde{\mathbf{K}}_{NN}$ also of size $N \times N$ but defined as

$$\tilde{\mathbf{K}}_{NN} = \mathbf{K}_{NM} \mathbf{K}_{MM}^{-1} \mathbf{K}_{NM}^\top \quad (4.35)$$

with $M \ll N$ [Williams and Seeger, 2000]. By using the approximated matrix $\tilde{\mathbf{K}}_{NN}$, we can approximate the log-marginal likelihood such as

$$\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \log \mathcal{N}_N(\mathbf{y}|\boldsymbol{\mu}, \mathbf{K} + \sigma^2 I_N) \approx \log \mathcal{N}_N(\mathbf{y}|\boldsymbol{\mu}, \tilde{\mathbf{K}}_{NN} + \sigma^2 I_N). \quad (4.36)$$

Using the Woodbury formula¹ [Max, 1950] in Equation (4.12), we have:

$$(\tilde{\mathbf{K}}_{NN} + \sigma^2 I_N)^{-1} = (\mathbf{K}_{NM} \mathbf{K}_{MM}^{-1} \mathbf{K}_{NM}^\top + \sigma^2 I_N)^{-1} \quad (4.37)$$

$$= (\sigma^2 I_N)^{-1} - (\sigma^2 I_N)^{-1} \mathbf{K}_{NM} (\mathbf{K}_{MM} + \mathbf{K}_{NM}^\top (\sigma^2 I_N)^{-1} \mathbf{K}_{NM})^{-1} \mathbf{K}_{NM}^\top (\sigma^2 I_N)^{-1} \quad (4.38)$$

$$= \sigma^{-2} I_N - \sigma^{-4} \mathbf{K}_{NM} (\mathbf{K}_{MM} + \sigma^{-2} \mathbf{K}_{NM}^\top \mathbf{K}_{NM})^{-1} \mathbf{K}_{NM}^\top. \quad (4.39)$$

¹ $(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{U} (\mathbf{C}^{-1} + \mathbf{VA}^{-1} \mathbf{U})^{-1} \mathbf{VA}^{-1}$

Thus, instead of calculating the inverse of a matrix $N \times N$, only the inverse of a matrix $M \times M$ is calculated. The computation complexity is reduced from $\mathcal{O}(N^3)$ to $\mathcal{O}(NM^2)$. However, the storage complexity is still $\mathcal{O}(N^2)$, as we still need to store the matrix \mathbf{K}_{NN} .

In remote sensing, the Nyström approximation was used for crop yield estimation with GP [Martínez-Ferrer et al., 2021]. However, we are still limited by the number of samples N .

Sparse methods

Sparse methods are methods based on **Inducing Points (IP)**. IP, also called "support points", "active set" or "pseudo inputs", correspond to a set of M latent variables $f(\mathbf{Z})$ with $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^M$. Depending on the algorithm used, different methods can be used to select these inducing points (e.g. subset of the training data set). Taking this set of inducing points \mathbf{Z} , the same GP prior is assumed: $f(\mathbf{Z}) \sim \mathcal{N}_M(0, \mathbf{K}_{MM})$ and for simplicity, we note $\mathbf{u} = f(\mathbf{Z})$. As a reminder, we have $\mathbf{y} = f(\mathbf{X}) + \epsilon$. By using the marginalization property, we have

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \int_{\mathbf{u}} p(\mathbf{y}, \mathbf{u}|\mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) d\mathbf{u} \quad (4.40)$$

$$= \int_{\mathbf{u}} p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z}) d\mathbf{u} \text{ (joint property)} \quad (4.41)$$

where

$$p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) = \mathcal{N}_N(\mathbf{y}|\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{u}, \mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top + \sigma^2 I_N)$$

provided by Equation (4.1) and where

$$p(\mathbf{u}|\mathbf{Z}) = \mathcal{N}_M(\mathbf{u}|0, \mathbf{K}_{MM}).$$

Computing $\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$ still involves computing \mathbf{K}_{NN} of size $N \times N$. Different approximations are made for $p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z})$ in order to reduce the complexity. For example, **Deterministic Training Conditional (DTC)** approximation [Seeger, 2003], assumes that $p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z})$ can be approximated by:

$$p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) \approx \mathcal{N}_N(\mathbf{y}|\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{u}, \sigma^2 I_N), \quad (4.42)$$

Fully Independent Training Conditional (FITC) approximation [Snelson and Ghahramani, 2005] assumes that $\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top$ is diagonal, such as:

$$p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) \approx \mathcal{N}_N(\mathbf{y}|\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{u}, \text{diag}(\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top) + \sigma^2 I_N), \quad (4.43)$$

Partially Independent Training Conditional (PITC) approximation [Snelson and Ghahramani, 2007] assumes that $\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top$ is block diagonal, such as:

$$p(\mathbf{y}|\mathbf{u}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) \approx \mathcal{N}_N(\mathbf{y}|\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{u}, \text{blockdiag}(\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top) + \sigma^2 I_N). \quad (4.44)$$

Focusing on FITC, by marginalizing \mathbf{u} , the log marginal likelihood can be approximated by

$$\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}, \mathbf{Z}) \approx \log \mathcal{N}_N(\mathbf{y}|0, \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top + \text{diag}(\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^\top) + \sigma^2 I_N). \quad (4.45)$$

Using the Woodbury formula, we have

$$(\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^{\top} + \mathbf{C})^{-1} = \mathbf{C}^{-1} - \mathbf{C}^{-1}\mathbf{K}_{NM}(\mathbf{K}_{MM} + \mathbf{K}_{NM}^{\top}\mathbf{C}^{-1}\mathbf{K}_{NM})^{-1}\mathbf{K}_{NM}^{\top}\mathbf{C}^{-1} \quad (4.46)$$

with $\mathbf{C} = \text{diag}(\mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{NM}^{\top}) + \sigma^2\mathbf{I}_N$.

Such as in Nyström approximation, the computation complexity is reduced from $\mathcal{O}(N^3)$ to $\mathcal{O}(NM^2)$. However, the storage complexity is still $\mathcal{O}(N^2)$, as we still need to store the matrix \mathbf{K}_{NN} .

In remote sensing, model approximation was widely used for large data sets. In most works, **IP** are used to reduce the complexity [Martínez-Ferrer et al., 2021], [Camps-Valls et al., 2016]. Usually, the inducing points \mathbf{Z} are a subset of the training inputs. The choice of subset is very important for the quality of the estimate. The literature has shown that using only subsets did not work well because they might not represent the data correctly [Quiñero-Candela and Rasmussen, 2005]. An effective approach is to consider the optimization of the **IP** during the learning step, in complement to the mean and covariance function parameters, as proposed in [Snelson and Ghahramani, 2005], [Hensman et al., 2015]. This approach considers a variational approximation of the posterior (instead of model approximation) which gives superior results in large scale scenarios. This aspect is discussed in the following part.

4.4.2. Posterior Approximation by Variational Inference

In model approximation, the main idea was to approximate the model with a simpler one in order to have a tractable inference. In approximate inference, the original model is kept (i.e. no modification of the prior) but instead the posterior is approximated. In the following, we will focus on the posterior approximation with **Variational Inference (VI)**.

Sparse Variational Gaussian Processes

By using the marginalization property, the marginal likelihood can be written as

$$p(\mathbf{y}|\mathbf{X}) = \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{y}|\mathbf{u}, \mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z}) d\mathbf{f} d\mathbf{u} \quad (4.47)$$

$$= \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z}) d\mathbf{f} d\mathbf{u} \quad (4.48)$$

with $p(\mathbf{y}|\mathbf{f}) = \mathcal{N}_N(\mathbf{y}|\mathbf{f}, \sigma^2\mathbf{I}_N)$, $p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z}) = \mathcal{N}_N(\mathbf{f}|\mathbf{K}_{MN}^{\top}\mathbf{K}_{MM}^{-1}\mathbf{u}, \mathbf{K}_{NN} - \mathbf{K}_{MN}^{\top}\mathbf{K}_{MM}^{-1}\mathbf{K}_{MN})$ and $p(\mathbf{u}|\mathbf{Z}) = \mathcal{N}_M(\mathbf{u}|0, \mathbf{K}_{MM})$. As a reminder, we note $\mathbf{u} = f(\mathbf{Z})$ and $\mathbf{f} = f(\mathbf{X})$.

Instead of using approximated $p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$, we will use Sparse Variational **GP** introduced by Titsias [Titsias, 2009]. A variational lower bound of the log marginal likelihood is defined

as:

$$\begin{aligned}
\log p(\mathbf{y}|\mathbf{X}) &= \log \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z}) \, d\mathbf{f} \, d\mathbf{u} \\
&= \log \int_{\mathbf{f}, \mathbf{u}} \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z}) \, d\mathbf{f} \, d\mathbf{u} \\
&= \log \mathbb{E}_{q(\mathbf{f}, \mathbf{u})} \left[\frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{f}, \mathbf{u})} \right] \\
&\geq \mathbb{E}_{q(\mathbf{f}, \mathbf{u})} \left[\log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{f}, \mathbf{u})} \right] \text{ (Jensen's inequality [Jensen, 1906])} \\
&= \int_{\mathbf{f}, \mathbf{u}} q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{f}, \mathbf{u})} \, d\mathbf{f} \, d\mathbf{u}.
\end{aligned}$$

Titsias proposed to define $q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u})$ where $q(\mathbf{u}) \sim \mathcal{N}_M(\mathbf{m}, \mathbf{S})$ is the variational distribution with $\mathbf{m} \in \mathbb{R}^M$ and $\mathbf{S} \in \mathbb{R}^{M \times M}$. We denote $\boldsymbol{\theta}^v = \{\mathbf{m}, \mathbf{S}\}$ the parameters of the variational distribution. The log marginal likelihood can be maximized by:

$$\begin{aligned}
\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) &\geq \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z})}{p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u})} \, d\mathbf{f} \, d\mathbf{u} \\
&= \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{u})} \, d\mathbf{f} \, d\mathbf{u} \\
&= \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}) \left(\log p(\mathbf{y}|\mathbf{f}) - \log \frac{p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{u})} \right) \, d\mathbf{f} \, d\mathbf{u} \\
&= \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}) \log p(\mathbf{y}|\mathbf{f}) \, d\mathbf{f} \, d\mathbf{u} - \int_{\mathbf{u}} q(\mathbf{u}) \log \frac{p(\mathbf{u}|\mathbf{Z})}{q(\mathbf{u})} \, d\mathbf{u} \\
&= \mathbb{E}_{p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u})} \left[\log p(\mathbf{y}|\mathbf{f}) \right] - \text{KL} \left[q(\mathbf{u}) \parallel p(\mathbf{u}|\mathbf{Z}) \right] \\
&= \mathcal{E}(q)
\end{aligned}$$

with $\mathcal{E}(q)$ the **Evidence Lower Bound (ELBO)**. $\boldsymbol{\theta}^v$ and $\boldsymbol{\theta}$ are optimized by maximizing the **ELBO**:

$$\mathcal{E}(q) = \mathbb{E}_{p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u})} \left[\log p(\mathbf{y}|\mathbf{f}) \right] - \text{KL} \left[q(\mathbf{u}|\boldsymbol{\theta}^v, \boldsymbol{\theta}) \parallel p(\mathbf{u}|\mathbf{Z}, \boldsymbol{\theta}) \right].$$

Having

$$q(\mathbf{f}|\mathbf{X}, \mathbf{Z}) = \int_{\mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}) \, d\mathbf{u}$$

with $p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z}) = \mathcal{N}_N(\mathbf{f}|\mathbf{K}_{MN}^\top \mathbf{K}_{MM}^{-1} \mathbf{u}, \mathbf{K}_{NN} - \mathbf{K}_{MN}^\top \mathbf{K}_{MM}^{-1} \mathbf{K}_{MN})$ and $q(\mathbf{u}) \sim \mathcal{N}_M(\mathbf{m}, \mathbf{S})$. We can rewrite it, using [Bishop, 2006, Chapter 2.3.3], as:

$$q(\mathbf{f}|\mathbf{X}, \mathbf{Z}) \sim \mathcal{N}_N \left(\mathbf{f} \mid \mathbf{K}_{MN}^\top \mathbf{K}_{MM}^{-1} \mathbf{m}, \mathbf{K}_{NN} - \mathbf{K}_{MN}^\top \mathbf{K}_{MM}^{-1} (\mathbf{K}_{MM} - \mathbf{S}) \mathbf{K}_{MM}^{-1} \mathbf{K}_{MN} \right).$$

The **ELBO** can be rewritten as:

$$\begin{aligned}\mathcal{E}(q) &= \int_{\mathbf{f}, \mathbf{u}} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z}) q(\mathbf{u}) \log p(\mathbf{y}|\mathbf{f}) \, d\mathbf{f} \, d\mathbf{u} - \text{KL} \left[q(\mathbf{u}|\boldsymbol{\theta}^v, \boldsymbol{\theta}) \parallel p(\mathbf{u}|\mathbf{Z}, \boldsymbol{\theta}) \right] \\ &= \int_{\mathbf{f}} q(\mathbf{f}|\mathbf{X}, \mathbf{Z}) \log p(\mathbf{y}|\mathbf{f}) \, d\mathbf{f} - \text{KL} \left[q(\mathbf{u}|\boldsymbol{\theta}^v, \boldsymbol{\theta}) \parallel p(\mathbf{u}|\mathbf{Z}, \boldsymbol{\theta}) \right] \\ &= \mathbb{E}_{q(\mathbf{f}|\mathbf{X}, \mathbf{Z})} \left[\log p(\mathbf{y}|\mathbf{f}) \right] - \text{KL} \left[q(\mathbf{u}|\boldsymbol{\theta}^v, \boldsymbol{\theta}) \parallel p(\mathbf{u}|\mathbf{Z}, \boldsymbol{\theta}) \right].\end{aligned}$$

Stochastic Variational Gaussian Processes (SVGP)

Hensman *et al.* [Hensman et al., 2013] proposed to factor the expectation term over data points. By noting:

$$\log p(\mathbf{y}|\mathbf{f}) = \sum_{i=1}^N \log p(y_i|f(\mathbf{x}_i)), \quad (4.49)$$

the expectation term can be rewritten as:

$$\mathbb{E}_{q(\mathbf{f}|\mathbf{X}, \mathbf{Z})} \left[\log p(\mathbf{y}|\mathbf{f}) \right] = \mathbb{E}_{q(\mathbf{f}|\mathbf{X}, \mathbf{Z})} \left[\sum_{i=1}^N \log p(y_i|f(\mathbf{x}_i)) \right] \quad (4.50)$$

$$= \sum_{i=1}^N \mathbb{E}_{q(f(\mathbf{x}_i)|\mathbf{x}_i, \mathbf{Z})} \left[\log p(y_i|f(\mathbf{x}_i)) \right] \quad (4.51)$$

Finally, the **ELBO** is equal to:

$$\begin{aligned}\mathcal{E}(q) &= \sum_{i=1}^N \mathbb{E}_{q(f(\mathbf{x}_i)|\boldsymbol{\theta}^v, \boldsymbol{\theta})} \left[\log p(y_i|f(\mathbf{x}_i)) \right] \\ &\quad - \text{KL} \left[q(f(\mathbf{Z})|\boldsymbol{\theta}^v, \boldsymbol{\theta}) \parallel p(f(\mathbf{Z})|\boldsymbol{\theta}) \right],\end{aligned} \quad (4.52)$$

with

$$\begin{aligned}q(f(\mathbf{x}_i)|\boldsymbol{\theta}^v, \boldsymbol{\theta}) &\sim \mathcal{N}_1 \left(f(\mathbf{x}_i) \mid \mathbf{k}_{M_i}^\top \mathbf{K}_{MM}^{-1} \mathbf{m}, \right. \\ &\quad \left. k(\mathbf{x}_i, \mathbf{x}_i) - \mathbf{k}_{M_i}^\top \mathbf{K}_{MM}^{-1} (\mathbf{K}_{MM} - \mathbf{S}) \mathbf{K}_{MM}^{-1} \mathbf{k}_{M_i} \right).\end{aligned} \quad (4.53)$$

The first term, the expectation term, can be computed analytically as $p(\mathbf{y}|\mathbf{f})$ is Gaussian, as we are working with univariate **GP** regression. It is not the case for the classification. Indeed, the first can not be calculated analytically as the likelihood is not Gaussian (c.f. Equation (4.32)). However, it can be estimated using Gauss-Hermite quadrature (for binary problems) or by **Monte Carlo (MC)** sampling (for multi-class problems) [Hensman et al., 2015]. The latter is discussed in Chapter 5.

The second term, the KL term, is the Kullback-Leibler divergence between two Gaussian distributions. It can be computed and derived analytically, as described in Section 4.1.2 [Rasmussen and Williams, 2005, Chapter A.3.1].

From the **ELBO** defined in the previous section, only the expectation term is rewritten, as the second term is not modified as it does not depend on the data. The **ELBO** can be optimized using stochastic optimization [Bottou et al., 2018] and more precisely mini batch learning with

stochastic gradient descent. This method is called **Stochastic Variational Gaussian Processes (SVGP)**. Finally, it leads to a computational complexity of $\mathcal{O}(M^3)$ instead of $\mathcal{O}(NM^2)$ without mini-batch. Using such strategy, Hensman *et al.* [Hensman et al., 2015] optimized the whole model, i.e. $\{\boldsymbol{\theta}, \boldsymbol{\theta}^v, \mathbf{Z}\}$, on 700 000 points for a regression problem on a mono-CPU computer. Table 4.3 summarizes the time and storage complexities for the main approximation methods previously defined: Sparse GP described in Section 4.4.1, Sparse Variational GP described in Section 4.4.2 and finally the Stochastic Variational GP described in Section 4.4.2.

In remote sensing, VI was used to model heteroscedastic noise in GP regression [Moreno-Muñoz et al., 2018] and for binary classification with model approximation [Morales-Alvarez et al., 2018]. Moreover, Svendsen *et al.* [Svendsen et al., 2020] used Variational GP with inducing points for the estimation of surface temperature from infrared sounding data, and for biophysical parameter estimation from Sentinel-3 time series. Thanks to the approximation, huge training and testing data sets were used: 250 000 and $\sim 10^6$ points, respectively.

Table 4.3.: *Time and storage complexities for the main GP approximation methods described in this chapter.*

Method	Approximation	Time complexity	Storage complexity	Optimization
Full GP	No approximation	$\mathcal{O}(N^3)$	$\mathcal{O}(N^2)$	Full Gradient Descent (GD)
Sparse GP	Handcrafted IP	$\mathcal{O}(NM^2)$	$\mathcal{O}(N^2)$	Full GD
Sparse Variational GP	Optimized IP	$\mathcal{O}(NM^2)$	$\mathcal{O}(NM)$	Full GD
Stochastic Variational GP	Optimized IP	$\mathcal{O}(M^3)$	$\mathcal{O}(BM)$	Mini-batch stochastic GD

CHAPTER 5

SVGP CLASSIFICATION: METHOD AND EXPERIMENTAL SET-UP

5.1. Large scale multi-class GP land cover classification	142
5.1.1. Training	142
5.1.2. Inference	144
5.1.3. Hyper-parameters	145
5.1.4. Model complexity	147
5.2. Experimental set-up	149
5.2.1. Configuration	149
5.2.2. Data set generation	149
5.2.3. Method set-up	153
5.2.4. Map production	156
5.2.5. Feature reduction	156

5.1. Large scale multi-class GP land cover classification

For a C -classes classification problem, we define as $\mathbf{x}_i = [\mathbf{x}_i(t_1), \dots, \mathbf{x}_i(t_T)] \in \mathbb{R}^d$, the raw feature vector made of the concatenation of all spectral measurements for every observation. We assume that a class membership one-hot vector $\mathbf{y}_i \in \{0, 1\}^C$ is associated to each \mathbf{x}_i . Thus, the training set is denoted $\mathcal{S} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$. In the following, we define a multi-class GP from **Stochastic Variational Gaussian Processes (SVGP)**, defined in Section 4.4.2. The nomenclature used in this chapter and the following is defined in Table 5.1.

5.1.1. Training

The simplified version of **LMC** (see Equation (4.21)) is used in order to extend the univariate case of **SVGP**, defined in Section 4.4.2, to the multivariate case [Hensman et al., 2015].

During the training, the parameters are learned by optimizing the posterior using variational inference, which corresponds to maximizing the **ELBO**, as described in the previous chapter. In the case of multi-class classification, the variational distribution is defined independently for each latent GP g_l such as $q(g_l(\mathbf{Z}_l)) \sim \mathcal{N}_M(\mathbf{m}_l, \mathbf{S}_l)$ with $l \in \{1, \dots, L\}$, L corresponding to the number of latent GP and with \mathbf{Z}_l the M inducing points for each latent GP g_l . Therefore, the trainable parameters correspond to:

- the parameters of the mean function m_l , the parameters of the covariance function k_l and the M values of the inducing points \mathbf{Z}_l for each latent GP g_l , denoted as $\boldsymbol{\theta}_l$,
- the values of the mixing matrix \mathbf{A} defined in Equation (4.21) and
- the parameters of the mean function \mathbf{m}_l and of the covariance function \mathbf{S}_l for each variational distribution $q(g_l(\mathbf{Z}_l))$, denoted as $\boldsymbol{\theta}_l^v$.

We denote $\mathbf{g} = [g_1, \dots, g_l, \dots, g_L]^\top$, the L -dimensional vector corresponding to all the latent functions. Therefore, we denote $\mathbf{g}(\mathbf{Z})$ the ML -dimensional random vector such as $\mathbf{g}(\mathbf{Z}) = [g_1(\mathbf{Z}_1), \dots, g_L(\mathbf{Z}_L)]^\top$. From the **LMC** definition in Section 4.3.1, it follows that

$$p(\mathbf{g}(\mathbf{Z})|\boldsymbol{\Theta}) = \prod_{l=1}^L p(g_l(\mathbf{Z}_l)|\boldsymbol{\theta}_l)$$

with $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_L\}$ and $p(g_l(\mathbf{Z}_l)|\boldsymbol{\theta}_l)$ Gaussian. Similarly, the same independence assumption is assumed for $q(\mathbf{g}(\mathbf{Z}))$ and we have:

$$q(\mathbf{g}(\mathbf{Z})) = \prod_{l=1}^L q(g_l(\mathbf{Z}_l)).$$

With these assumptions, the **ELBO**, from Equation (4.52), can be rewritten as

$$\mathcal{E}(q) = \sum_{i=1}^N \mathbb{E}_{q(\mathbf{g}(\mathbf{x}_i)|\boldsymbol{\Theta}^v, \boldsymbol{\Theta})} \left[\log p(\mathbf{y}_i|\mathbf{g}(\mathbf{x}_i), \mathbf{A}) \right] - \text{KL} \left[q(\mathbf{g}(\mathbf{Z})|\boldsymbol{\Theta}^v, \boldsymbol{\Theta}) \parallel p(\mathbf{g}(\mathbf{Z})|\boldsymbol{\Theta}) \right] \quad (5.1)$$

$$= \sum_{i=1}^N \mathbb{E}_{q(\mathbf{g}(\mathbf{x}_i)|\boldsymbol{\Theta}^v, \boldsymbol{\Theta})} \left[\log p(\mathbf{y}_i|\mathbf{g}(\mathbf{x}_i), \mathbf{A}) \right] - \sum_{l=1}^L \text{KL} \left[q(g_l(\mathbf{Z}_l)|\boldsymbol{\theta}_l^v, \boldsymbol{\theta}_l) \parallel p(g_l(\mathbf{Z}_l)|\boldsymbol{\theta}_l) \right] \quad (5.2)$$

Table 5.1.: Nomenclature used in Chapters 5 and 6

Symbol	Meaning
\mathbf{A}	Mixing matrix, $\mathbf{A} \in \mathbb{R}^{C \times L}$
\hat{c}	Class predicted
C	Number of classes, $c \in \{1, \dots, C\}$
d, d'	Number of spectro-temporal, spatial features
D	Number of spectral measurements
\mathcal{E}	ELBO
$\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$	C -multivariate GP such as $\mathbf{f} = \mathbf{A}\mathbf{g}$ with mean function \mathbf{m} and covariance function \mathcal{K}
$g_l \sim \mathcal{GP}(m_l, k_l)$	Univariate GP, the l^{th} latent GP with mean function m_l and covariance function k_l
\mathbf{g}	Vector of L independent univariate GP, $\mathbf{g} = [g_1, \dots, g_L]$
$k_{l\phi}$	Spatial covariance function
$k_{l\lambda t}$	Spectro-temporal covariance function
k_l^P	Covariance function of the model $\phi\lambda t$ -GPPC
k_l^S	Covariance function of the model $\phi\lambda t$ -GPSC
\mathbf{K}_{MM}^l	Covariance matrix of the distribution $p(g_l(\mathbf{Z}_l) \mathbf{Z}_l) = \mathcal{N}_M(0, \mathbf{K}_{MM}^l)$.
\mathcal{K}^v	Covariance matrix of the L -dimensional distribution $q(\mathbf{g}(\mathbf{x}_i) \boldsymbol{\theta}^v, \boldsymbol{\theta}) \sim \mathcal{N}_L(\mathbf{g}(\mathbf{x}_i) \mathbf{m}^v, \mathcal{K}^v)$
\mathcal{K}_{ll}^v	The diagonal l^{th} element of diagonal covariance matrix \mathcal{K}^v
L	Number of latent processes, $l \in \{1, \dots, L\}$
$\ell_{l\lambda t}, \ell_{l\phi\lambda t}$	Length-scales of the covariance functions $k_{l\phi}$ and $k_{l\lambda t}$, respectively.
\mathbf{m}_l	Mean vector of the distribution $q(g_l(\mathbf{Z}_l)) \sim \mathcal{N}_M(\mathbf{m}_l, \mathbf{S}_l)$
\mathbf{m}^v	Mean matrix of the L -dimensional distribution $q(\mathbf{g}(\mathbf{x}_i) \boldsymbol{\theta}^v, \boldsymbol{\theta}) \sim \mathcal{N}_L(\mathbf{g}(\mathbf{x}_i) \mathbf{m}^v, \mathcal{K}^v)$
M	Number of inducing points
N	Number of training inputs
S	Number of realizations for the MC sampling
\mathbf{S}_l	Covariance matrix of the distribution $q(g_l(\mathbf{Z}_l)) \sim \mathcal{N}_M(\mathbf{m}_l, \mathbf{S}_l)$
$\sigma_{l\lambda t}, \sigma_{l\phi\lambda t}$	Spectro-temporal, spatial output-scales from the covariance function k_l^S
T	Number of observations
$\boldsymbol{\theta}_l$	Hyper-parameters of the latent process g_l
$\boldsymbol{\Theta}$	Hyper-parameters of \mathbf{g} , $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_L\}$
$\boldsymbol{\theta}_l^V$	Parameters of the variational distribution q , $\boldsymbol{\theta}_l^V = \{\mathbf{m}_l, \mathbf{S}_l\}$
$\boldsymbol{\Theta}^V$	Parameters of all the variational distributions $\boldsymbol{\Theta}^V = \{\boldsymbol{\theta}_1^V, \dots, \boldsymbol{\theta}_L^V\}$
$\mathbf{x}_i, \mathbf{y}_i$	The i^{th} training input, target
$\mathbf{x}_{i\phi}, \mathbf{x}_{i\lambda t}$	Spatial, spectro-temporal features of the i^{th} pixel
$\mathbf{x}_*, \mathbf{y}_*$	New input, target
\mathbf{Z}_l	Set of inducing points for the latent process g_l

with $\Theta^v = \{\theta_1^v, \dots, \theta_L^v\}$. From Equation (4.52), $f(\mathbf{x}_i)$ is replaced by $\mathbf{g}(\mathbf{x}_i)$ as we have a multivariate GP instead of an univariate GP. A simplified version of LMC is used to extend the univariate to the multivariate, therefore, the matrix \mathbf{A} is optimized during the training process. $q(\mathbf{g}(\mathbf{x}_i)|\Theta^v, \Theta)$ is a L -dimensional Gaussian distribution with diagonal covariance matrix

$$q(\mathbf{g}(\mathbf{x}_i)|\Theta^v, \Theta) \sim \mathcal{N}_L(\mathbf{g}(\mathbf{x}_i)|\mathbf{m}^v, \mathcal{K}^v). \quad (5.3)$$

Each marginal is given by Equation (4.53), a consequence of the LMC: the latent processes become dependent on one another only during the computation of the likelihood. Specifically, the l^{th} element of the mean vector and of the diagonal of the covariance matrix are totally specified by the l^{th} latent process:

$$\mathbf{m}_l^v = \mathbf{k}_{Mi}^{l\top} \mathbf{K}_{MM}^{l-1} \mathbf{m}_l, \quad (5.4)$$

$$\mathcal{K}_{ll}^v = k_l(\mathbf{x}_i, \mathbf{x}_i) - \mathbf{k}_{Mi}^{l\top} \mathbf{K}_{MM}^{l-1} (\mathbf{K}_{MM}^l - \mathbf{S}_l) \mathbf{K}_{MM}^{l-1} \mathbf{k}_{Mi}^l. \quad (5.5)$$

As for the regression case, the KL terms can be computed and derived in closed-form. The expectation term needs to be approximated. Indeed, the likelihood defined in Equation (5.2) is not Gaussian. MC sampling is used, similar to [Hensman et al., 2015], [Wilson et al., 2016]. It is combined with the so-called *reparametrization trick* from Variational Auto-encoder (VAE) to compute the derivative of the expectation during the stochastic gradient descent [Kingma and Welling, 2019, section 2.4]. With the reparametrization trick, the ELBO can be rewritten as

$$\begin{aligned} \mathcal{E}(q) &= \sum_{i=1}^N \mathbb{E}_{p(\epsilon)} \left[\log p(\mathbf{y}_i | \mathbf{g}(\mathbf{x}_i), \mathbf{A}) \right] \\ &\quad - \sum_{l=1}^L \text{KL} \left[q(g_l(\mathbf{Z}_l) | \theta_l^v, \theta_l) \parallel p(g_l(\mathbf{Z}_l) | \theta_l) \right] \end{aligned} \quad (5.6)$$

with $p(\epsilon) = \mathcal{N}_L(0, I_L)$ and $\mathbf{g}(\mathbf{x}_i) = \mathbf{m}^v + \mathcal{K}^v \epsilon$ with $\epsilon \sim p(\epsilon)$. Thus, using MC sampling, we have:

$$\sum_{i=1}^N \mathbb{E}_{p(\epsilon)} \left[\log p(\mathbf{y}_i | \mathbf{g}(\mathbf{x}_i), \mathbf{A}) \right] \approx \sum_{i=1}^N \left(\frac{1}{S} \sum_{s=1}^S \left[\log p(\mathbf{y}_i | \mathbf{g}(\mathbf{x}_i)^{(s)}, \mathbf{A}) \right] \right)$$

with $\mathbf{g}(\mathbf{x}_i)^{(s)}$ the s th sample of MC sampling. In practice, one realization ($S = 1$) is enough for the MC sampler during the training, as found in VAE [Kingma and Welling, 2014], [Hensman et al., 2015].

5.1.2. Inference

The prediction for a new input \mathbf{x}_* uses the same variational approximation for the joint prior than in the marginal likelihood, and reduces to:

$$p(\mathbf{y}_* | \mathbf{Y}, \mathbf{X}, \mathbf{x}_*) = \mathbb{E}_{q(\mathbf{g}(\mathbf{x}_*) | \Theta^v, \Theta)} \left[p(\mathbf{y}_* | \mathbf{g}(\mathbf{x}_*), \mathbf{A}) \right] \quad (5.7)$$

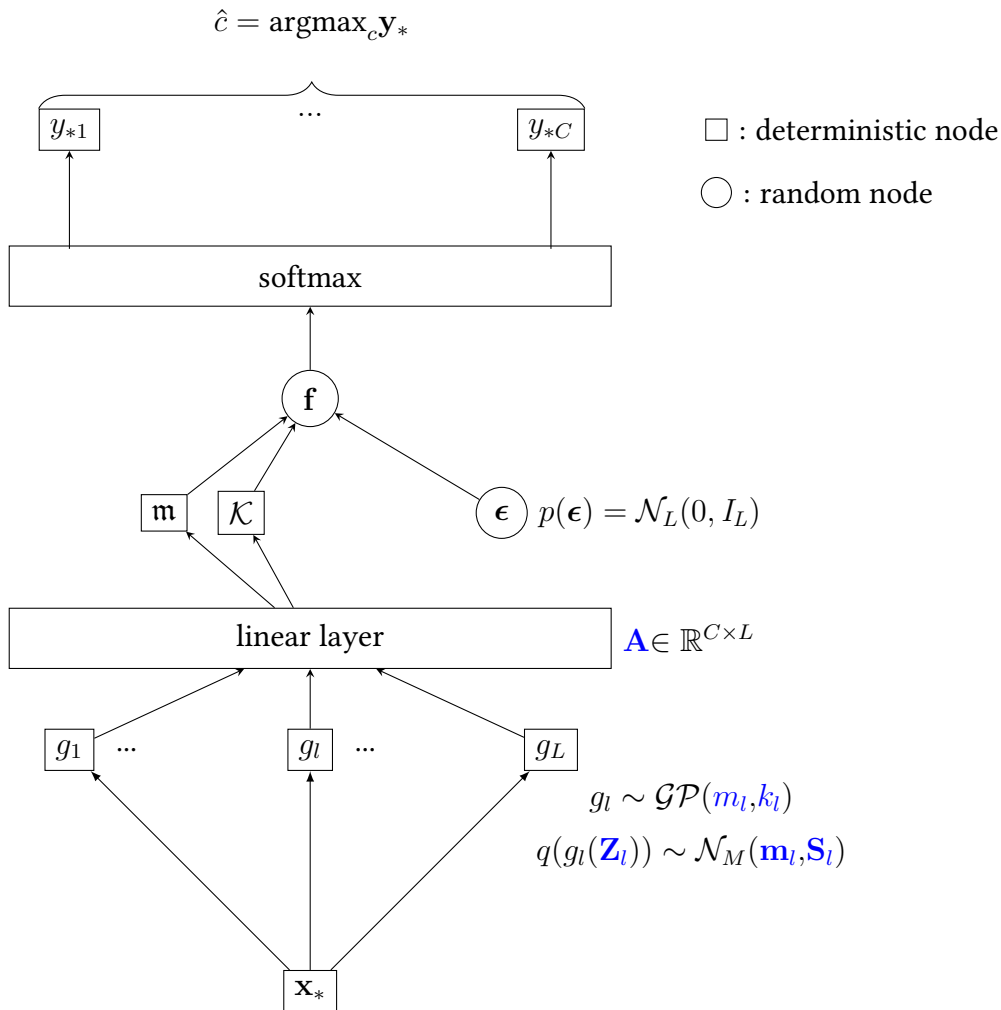


Figure 5.1: Model for the prediction on one new input \mathbf{x}_* . The predicted class corresponds to \hat{c} . The trainable parameters are written in blue.

with $q(\mathbf{g}(\mathbf{x}_*) | \Theta^v, \Theta)$ given by Equation (5.3). Again, the expectation is not analytically tractable: the approximation is obtained with MC sampling. In this work, 10 realizations were used for the inference. The influence on the number of draws for the inference is studied in Section 6.3.1. Figure 5.1 represents this model for the prediction on one new input \mathbf{x}_* . The class is estimated by taking $\hat{c} = \operatorname{argmax}_c \mathbf{y}_*$.

5.1.3. Hyper-parameters

Different choices were made for the model in order to implement multi-class GP for land cover classification in large scale. The definition of the main hyper-parameters (mean function, covariance function and number of inducing points) is presented in the following. Influence of their parametrization as well as their initialization is discussed in Section 6.3.

Mean function

For each latent function g_l , we proposed to define the mean function m_l as a constant:

$$m_l(\mathbf{x}_i) = \mu_l. \quad (5.8)$$

The trainable parameter for each univariate GP is μ_l .

Covariance function

We proposed to use composite covariance functions in order to exploit the spatio-spectro-temporal structure of the data through the covariance function. Thus, for each latent function g_l , we proposed to define the covariance function $k_l(\mathbf{x}_i, \mathbf{x}_{i'})$ as a composition of a spatial covariance function $k_{l\phi}(\mathbf{x}_{i,\phi}, \mathbf{x}_{i',\phi})$ and a spectro-temporal covariance function $k_{l\lambda t}(\mathbf{x}_{i,\lambda t}, \mathbf{x}_{i',\lambda t})$. Where $\mathbf{x}_{i,\phi}$ and $\mathbf{x}_{i,\lambda t}$ are composed of d' spatial features and d spectro-temporal features, respectively. Indeed, distant pixels from different classes can have a similar vegetation phenology because of latitudinal and topographical effects on the biological cycle and cannot be discriminated with the spectro-temporal information only. Such modeling takes into account both the phenology and the spatial location in the studied area. Thus, this configuration prevents two spatially distant pixels to be correlated even if they share a similar spectro-temporal profile.

The RBF kernel, defined in Table 4.2, is used for both $k_{l\phi}$ and $k_{l\lambda t}$. It is composed of two parameters: the output-scale $\sigma > 0$ and the length-scale $\ell > 0$. This kernel uses isotropic distance between pixels in the spatial and spectro-temporal domain and the proximity between two pixels is controlled by the length-scale parameter ℓ : a small value tends to make all pixels uncorrelated ($k(\mathbf{x}_i, \mathbf{x}_{i'}) \approx 0$) and a high value tends to increase the correlation between pixels ($k(\mathbf{x}_i, \mathbf{x}_{i'}) \approx 1$).

In this work, we propose to study two different combinations of kernels. The first combination is the *sum of kernels*:

$$\begin{aligned} k_l^S(\mathbf{x}_i, \mathbf{x}_{i'}) &= \sigma_{l\phi}^2 \times k_{l\phi}(\mathbf{x}_{i,\phi}, \mathbf{x}_{i',\phi}) + \sigma_{l\lambda t}^2 \times k_{l\lambda t}(\mathbf{x}_{i,\lambda t}, \mathbf{x}_{i',\lambda t}) \\ &= \sigma_{l\phi}^2 \exp\left(-\frac{\|\mathbf{x}_{i,\phi} - \mathbf{x}_{i',\phi}\|_2^2}{2\ell_{l\phi}^2}\right) + \sigma_{l\lambda t}^2 \exp\left(-\frac{\|\mathbf{x}_{i,\lambda t} - \mathbf{x}_{i',\lambda t}\|_2^2}{2\ell_{l\lambda t}^2}\right). \end{aligned} \quad (5.9)$$

For each covariance function k_l^S , the trainable parameters are: $\{\sigma_{l\phi}, \sigma_{l\lambda t}, \ell_{l\phi}, \ell_{l\lambda t}\}$. The scaling parameters $\sigma_{l\phi}$ and $\sigma_{l\lambda t}$ allow to give different weights to either spatial or spectro-temporal features. The second combination is the *product of kernels*:

$$\begin{aligned} k_l^P(\mathbf{x}_i, \mathbf{x}_{i'}) &= k_{l\phi}(\mathbf{x}_{i,\phi}, \mathbf{x}_{i',\phi}) \times k_{l\lambda t}(\mathbf{x}_{i,\lambda t}, \mathbf{x}_{i',\lambda t}) \\ &= \exp\left(-\frac{\|\mathbf{x}_{i,\phi} - \mathbf{x}_{i',\phi}\|_2^2}{2\ell_{l\phi}^2}\right) \times \exp\left(-\frac{\|\mathbf{x}_{i,\lambda t} - \mathbf{x}_{i',\lambda t}\|_2^2}{2\ell_{l\lambda t}^2}\right). \end{aligned} \quad (5.10)$$

For each covariance function k_l^P , the trainable parameters are: $\{\ell_{l\phi}, \ell_{l\lambda t}\}$. In this covariance function, the output-scale is not used as the scale of the kernel function is handled by the mixing matrix \mathbf{A} .

Inducing points (IP)

A set of IP \mathbf{Z}_l of size M is associated to each latent process g_l . The number M of IP is the same for each latent GP g_l . During the optimization, these IP are independently learned for each latent GP g_l .

5.1.4. Model complexity

The model has a computational complexity of $\mathcal{O}((CM)^3)$ and its storage complexity is $\mathcal{O}((CN)^2)$. Three different GP models are defined:

1. λt -GP: GP model trained using only spectro-temporal features $\mathbf{x}_{\lambda t}$. For each g_l , its covariance function is: $k_{l\lambda t}(\mathbf{x}_{i,\lambda t}, \mathbf{x}_{i',\lambda t})$.
2. $\phi\lambda t$ -GPSC (GP Sum Covariance): GP model trained using spectro-temporal features $\mathbf{x}_{\lambda t}$ and spatial features \mathbf{x}_ϕ . Its covariance function, $k_l^S(\mathbf{x}, \mathbf{x}')$, is defined in Equation (5.9).
3. $\phi\lambda t$ -GPPC (GP Product Covariance): GP model trained using spectro-temporal features $\mathbf{x}_{\lambda t}$ and spatial features \mathbf{x}_ϕ . Its covariance function, $k_l^P(\mathbf{x}, \mathbf{x}')$, is defined in Equation (5.10).

Different parameters need to be optimized during the training. Focusing on the λt -GP model, one parameter (length-scale) needs to be learned for each covariance function k_l and also one parameter (mean constant) for each mean function m_l . Concerning the inducing points \mathbf{Z}_l , $M \times d$ values need to be learned for each latent GP g_l . M constant values need to be learned for each \mathbf{m}_l and $(M(M+1))/2$ for each \mathbf{S}_l (i.e. symmetric matrix). The values of the mixing matrix \mathbf{A} of size $L \times C$ need also to be learned. Finally, the total number of trainable parameters for λt -GP is

$$L \left(1 + 1 + Md + M + \frac{M(M+1)}{2} \right) + LC.$$

Three additional parameters for each k_l are learned for $\phi\lambda t$ -GPSC. Only one additional parameter for each k_l is learned for $\phi\lambda t$ -GPPC. Moreover, $M(d+d')$ values are computed, instead of Md for each \mathbf{Z}_l . These are the only trainable parameters that differ between GP models. Table 5.2 summarizes the different trainable parameters and their respective sizes for each model: λt -GP, $\phi\lambda t$ -GPSC and $\phi\lambda t$ -GPPC.

Focusing on the λt -GP model, LdM correspond to the number of parameters to learn for the IP and $L \left(2 + M + \frac{M(M+1)}{2} \right) + LC$ correspond to all the other trainable parameters except the IP. In our experimental setting, $L = C = 23$ and $M = 50$, the inducing points represent a large proportion of the trainable parameters. Therefore, the number of trainable parameters is mostly controlled by the number of inducing points M , as illustrated in Figure 5.2. For $d = 481$, the inducing points represent 95% of trainable parameters.

A reduction of the number of spectro-temporal features will decrease drastically the model parameters number. In Section 5.2.5, an experimental set-up is proposed in order to study the influence on the reduction of the number of spectro-temporal features.

Table 5.2.: Description of the trainable parameters and their corresponding sizes. The last line corresponds to the total number of trainable parameters for each model. For each latent function g_l , the same form of the mean and kernel functions were chosen as well as the same number M of inducing points.

	λt -GP	$\phi \lambda t$ -GPSC	$\phi \lambda t$ -GPPC
k_l	1	4	2
m_l	1	1	1
\mathbf{Z}_l	Md	$M(d + d')$	$M(d + d')$
\mathbf{m}_l	M	M	M
\mathbf{S}_l	$(M(M + 1))/2$	$(M(M + 1))/2$	$(M(M + 1))/2$
\mathbf{A}	LC	LC	LC
Total	$L(1 + 1 + Md + M + \frac{M(M+1)}{2}) + LC$	$L(4 + 1 + M(d + d') + M + \frac{M(M+1)}{2}) + LC$	$L(2 + 1 + M(d + d') + M + \frac{M(M+1)}{2}) + LC$

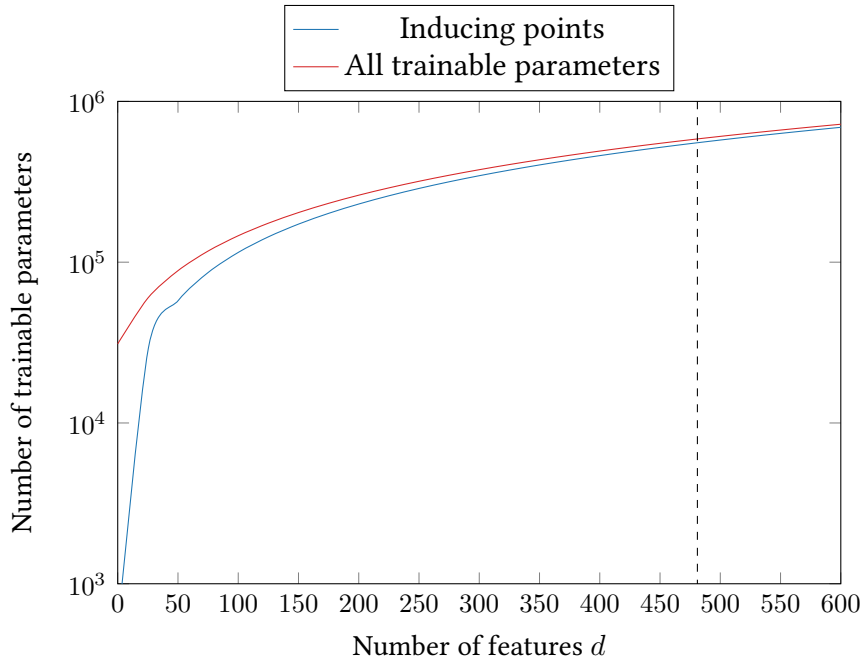


Figure 5.2: Number of trainable parameters for the λt -GP model as a function of the number of spectro-temporal features. The blue curve represents only the inducing points. The red curve represents all the trainable parameters (including the inducing points). In this case, we defined $L = C = 23$ and $M = 50$. The vertical dotted line corresponds to $d = 481$.

5.2. Experimental set-up

This section describes the experimental set-up implemented, the results associated are presented in the next chapter, Chapter 6. The methods used to prepare the training/validation/test sets and to measure the classification accuracy are presented in Appendix A.

5.2.1. Configuration

Two learning scenarios were considered: with and without spatial stratification. As defined in Section 3.3.3, spatial stratification with eco-climatic regions allows to reduce the spectro-temporal variability of pixel reflectances. It also enables to reduce the massive training data set. For the first scenario, with spatial stratification, *stratification* configuration, a dedicated learning model was fit on each eco-climatic region, and global predictions were obtained by joining per-region model predictions over the full area. For the second scenario, *global* configuration, only one model was learned using pixels over the full area. Figure 5.3 represents pixels used for the training in *stratification* and *global* configurations.

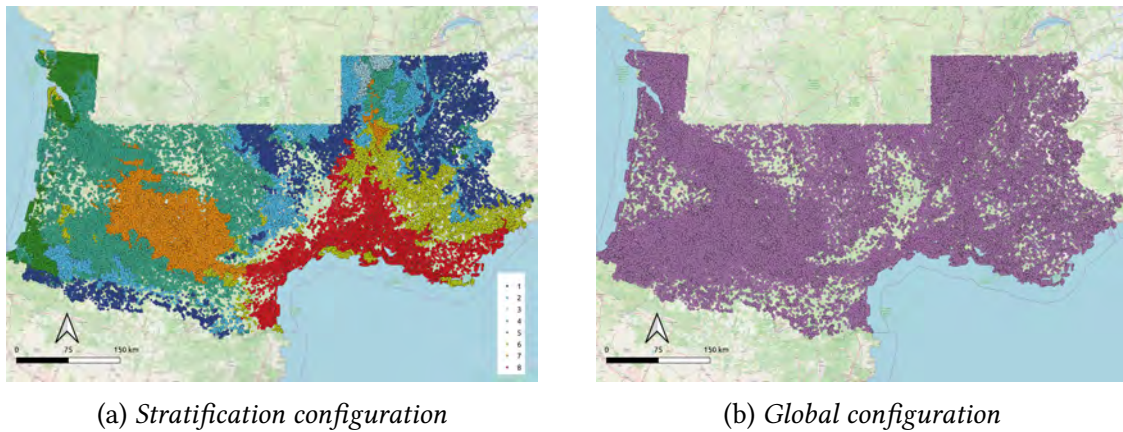


Figure 5.3: Pixels used for the training in stratification and global configurations (background map © OpenStreetMap contributors).

5.2.2. Data set generation

The data used is described in Chapter 3. In the following, only specific pre-processing for Chapters 5 and 6 is described. A total of $D = 13$ spectral features were extracted for each pixel \mathbf{x}_i at time t_k . Moreover, two spatial features describe each pixel. Temporal resampling is applied and interpolated time series are used. A set of 483 features describes each pixel \mathbf{x}_i as $d + d'$ with:

- $d = 481$ which corresponds to 37 interpolated dates \times 13 spectral features,
- $d' = 2$ spatial features.

Two different data sets are generated: a *classification* data set and a *boundary* data set. The *classification* data set is used to train and evaluate the model in large scale setting. Concerning RF models, the *stratification* configuration has shown an improvement in performances [Inglada et al., 2017]. However, some discontinuities in prediction for RF models can

be found at the boundaries between eco-climatic regions, as illustrated in Figure 2.10. Thus, the *boundary* data set is used to evaluate the spatial continuity or discontinuity of the class membership prediction between two eco-climatic regions.

Classification data set

The *classification* data set is used to train and validate the model. Different pixels were extracted randomly from the ground truth polygons in order to form three *spatially disjoint* data subsets: *training*, *validation* and *test* for each eco-climatic region. The global data set is composed of sets from all eco-climatic regions.

Two sizes for the *training-validation* subsets have been investigated for the learning step: (4 000, 1 000) and (16 000, 4 000) pixels per class, respectively called data set DS-A and data set DS-B. 10 000 pixels per class were extracted for the *test* set (except for the classes with fewer pixels, for which all were selected). Two data sets DS-A and DS-B were generated in order to evaluate the learning capabilities and performance in two different large scale configurations.

To estimate the classification metrics, 11 runs with different random pixel samplings were done. Table 5.3 provides the average number of pixels for each class and each eco-climatic region for the 11 *training-validation-test* pixels subsets. In *global* configuration, with the data set DS-A, the *training* data set contains around 646 000 pixels, and around 2 348 000 pixels, for DS-B. Moreover, in average, the *test* data set has 1 551 904 pixels.

Table 5.3.: Average number of pixels per class and regions for the classification data set. For a given class, the two first rows (data set DS-A and B) indicate the number of training-validation pixels per region and the third rows indicates the number of test pixels per region. The nomenclature of the 23 land cover classes can be found in Table 3.3.

Class	Regions								Global
	1	2	3	4	5	6	7	8	
CUF	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	6569 - 1727	16000 - 4000	16000 - 4000	16000 - 4000	12011 - 2676	10802 - 2657	16000 - 4000	16000 - 4000	109382 - 27061
	7286	10000	10000	10000	10000	10000	10000	10000	77286
DUF	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
ICU	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
RSF	3939 - 966	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	2562 - 658	4000 - 1000	4000 - 1000	30501 - 7624
	5191 - 2104	16000 - 4000	7642 - 4000	16000 - 4000	9148 - 2769	2562 - 658	16000 - 4000	16000 - 4000	88543 - 23457
	6622	10000	10000	10000	10000	5360	10000	10000	71982
RAP	4000 - 987	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 941	32000 - 7928
	5942 - 1424	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	7261 - 2125	109204 - 27549
	4551	10000	10000	10000	10000	10000	10000	10000	74551
STC	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
PRO	1073 - 340	4000 - 1000	1188 - 363	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	26261 - 6704
	1073 - 340	9748 - 2596	1188 - 363	16000 - 4000	16000 - 4000	11945 - 2709	16000 - 4000	13154 - 3243	85110 - 21253
	1222	10000	3120	10000	10000	10000	10000	10000	64342
SOY	3998 - 902	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	31998 - 7902
	4362 - 1122	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 3959	16000 - 4000	16000 - 4000	16000 - 344	116362 - 28525
	7098	10000	10000	10000	10000	10000	10000	10000	77098
SUN	1316 - 437	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	29316 - 7437
	1437 - 1122	16000 - 4000	16000 - 3757	16000 - 4000	16000 - 3959	16000 - 4000	16000 - 4000	16000 - 344	113316 - 28194
	3492	10000	10000	10000	10000	10000	10000	10000	73492
COR	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
RIC	0 - 0	0 - 0	0 - 0	0 - 0	0 - 0	4000 - 1000	0 - 0	4000 - 1000	8000 - 2000
	0 - 0	0 - 0	0 - 0	0 - 0	0 - 0	16000 - 4000	0 - 0	16000 - 4000	32000 - 8000
	0	0	0	0	0	10000	0	10000	20000
TUB	1604 - 411	3836 - 912	2757 - 676	4000 - 1000	4000 - 988	4000 - 1000	4000 - 1000	4000 - 1000	28199 - 6988
	1604 - 411	3928 - 1078	2757 - 676	16000 - 4000	8688 - 2563	11518 - 3296	16000 - 4000	16000 - 3985	76497 - 20011
	1816	5185	5864	10000	10000	10000	10000	10000	62865
GRA	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
ORC	844 - 173	4000 - 1000	1175 - 343	4000 - 1000	3236 - 800	4000 - 1000	4000 - 1000	4000 - 1000	25256 - 6317
	844 - 173	15967 - 3930	1175 - 343	16000 - 4000	3236 - 965	16000 - 4000	16000 - 4000	16000 - 4000	85223 - 21412
	657	10000	3026	10000	3590	10000	10000	10000	57273
VIN	672 - 207	4000 - 987	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	28672 - 7194
	672 - 207	5399 - 1545	6255 - 1649	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	92327 - 23402
	574	5115	9200	10000	10000	10000	10000	10000	64889
BLF	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 32000
	10000	10000	10000	10000	10000	10000	10000	10000	80000
COF	4000 - 1000	4000 - 1000	2598 - 648	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	30598 - 7648
	16000 - 4000	16000 - 4000	2598 - 717	16000 - 4000	16000 - 3896	16000 - 4000	16000 - 4000	16000 - 4000	114598 - 28614
	10000	10000	5317	10000	10000	10000	10000	10000	75317
NGL	4000 - 1000	4000 - 1000	0 - 0	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	28000 - 7000
	16000 - 4000	16000 - 4000	0 - 0	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	112000 - 28000
	10000	10000	0	10000	10000	10000	10000	10000	70000
WOM	4000 - 1000	4000 - 1000	3983 - 925	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	31983 - 7925
	16000 - 4000	16000 - 4000	4920 - 1401	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	16000 - 4000	116920 - 29401
	10000	10000	6189	10000	10000	10000	10000	10000	76189
NMS	4000 - 1000	4000 - 1000	0 - 0	4000 - 1000	3437 - 768	4000 - 1000	0 - 0	4000 - 1000	23437 - 5768
	16000 - 4000	16000 - 4000	0 - 0	16000 - 3773	7654 - 1795	16000 - 4000	0 - 0	16000 - 3932	87654 - 21500
	10000	10000	0	10000	3140	10000	0	10000	53140
BDS	4000 - 1000	3990 - 748	0 - 0	4000 - 931	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	27990 - 6679
	15713 - 3853	5274 - 1194	0 - 0	16000 - 2137	16000 - 3972	16000 - 4000	6817 - 4000	16000 - 4000	91805 - 23157
	10000	9097	0	10000	10000	10000	0	10000	59097
GPS	4000 - 1000	0 - 0	0 - 0	0 - 0	3715 - 818	0 - 0	0 - 0	0 - 0	7715 - 1818
	16000 - 4000	0 - 0	0 - 0	0 - 0	4773 - 2114	0 - 0	0 - 0	0 - 0	20773 - 6114
	10000	0	0	0	4383	0	0	0	14383
WAT	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	4000 - 1000	32000 - 8000
	16000 - 4000	16000 - 3915	16000 - 4000	16000 - 3957	16000 - 3586	16000 - 4000	16000 - 4000	16000 - 4000	128000 - 31459
	10000	10000	10000	10000	10000	10000	10000	10000	80000
Total	73450 - 18427	83828 - 20649	63703 - 15957	84000 - 20931	86390 - 21376	86563 - 21658	80000 - 20000	88000 - 21941	645934 - 160939
	235292 - 59801	296318 - 74180	202537 - 50854	336000 - 81868	301513 - 76300	324829 - 81382	310817 - 80000	340416 - 84730	2347722 - 589115
	163318	199397	152716	210000	201113	215360	190000	220000	1551904

Boundary data set

The *boundary* data set is used to evaluate the spatial continuity of the model predictions at the border between two eco-climatic regions. A synthetic example of a *boundary* data set is represented in Figure 5.4. The *boundary* data set is composed of labeled and unlabeled pixels in a buffered zone around the boundary between two regions. Unlabeled pixels are used in order to increase the number of pixels and because the study of continuity (i.e. computation of agreement) does not require labeled pixels.

Several buffer sizes B have been investigated: $B \in \{100, 200, 500, 1000, 1500, 2000\}$ meters, the total width of the buffer being equal to $2 \times B$. A buffered zone with real data between two eco-climatic regions is given in Figure 5.5. All available labeled pixels were selected except those included in the *training* and *validation* data sets. From the available unlabeled pixels, approximately 1% were selected. Table 5.4 summarizes the number of labeled and unlabeled pixels for each buffer size.

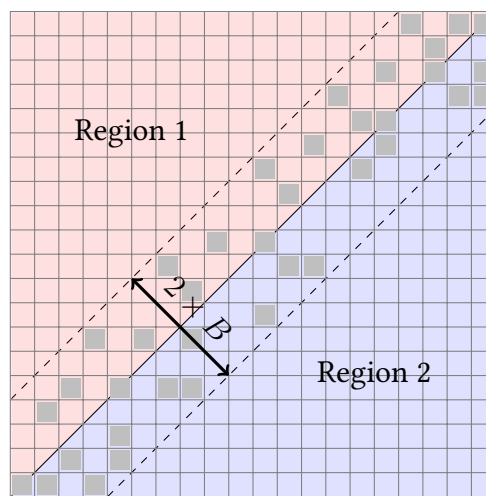


Figure 5.4: Synthetic representation of a buffered zone: the solid line represents the boundary between two eco-climatic regions and the area inside the dotted lines corresponds to the buffered zone of size $2 \times B$. Gray pixels are selected to compose the boundary data set.

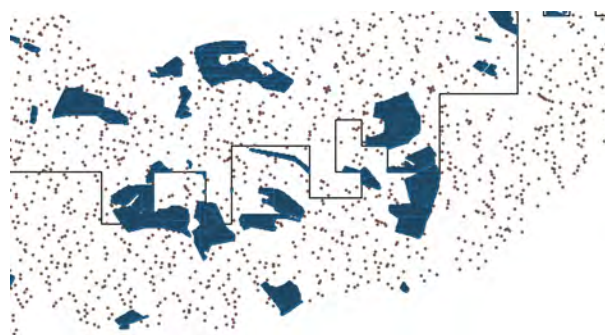


Figure 5.5: Example with real data: pixels are extracted from the $2 \times B = 2000m$ buffered zone between regions 4 and 7 in the T31TCJ tile. Labeled pixels are represented with \bullet and unlabeled pixels are represented with \bullet .

Table 5.4.: Number of extracted pixels in the boundary data set for each buffer size. Labeled pixels for each class and also unlabeled pixels are represented.

Class	Buffer size $2 \times B$ (in meters)					
	200	400	1 000	2 000	3 000	4 000
CUF	13 210	24 795	54 063	89 637	120 212	145 055
DUF	69 801	129 865	290 381	551 337	793 284	985 294
ICU	37 873	76 091	175 984	345 776	499 108	632 258
RSF	4 413	8 319	20 344	42 039	63 063	77 133
RAP	39 251	73 323	149 778	250 106	329 672	408 112
STC	62 048	119 209	250 463	440 889	583 845	710 629
PRO	13 729	27 975	68 267	124 310	158 918	196 369
SOY	54 631	107 367	243 272	404 260	536 731	667 757
SUN	140 271	262 218	574 634	987 998	1 315 013	1 597 642
COR	139 962	261 293	583 261	1 019 811	1 360 352	1 651 259
RIC	7 952	14 465	32 304	63 066	82 738	95 780
TUB	4 479	10 608	21 697	41 108	57 657	74 043
GRA	151 587	289 454	636 485	1 141 138	1 551 963	1 892 411
ORC	10 512	20 144	46 956	81 462	109 277	133 584
VIN	29 979	56 131	129 244	239 707	323 826	403 441
BLF	334 754	634 454	1 430 734	2 480 683	3 323 765	3 974 349
COF	623 400	1 175 363	2 615 784	4 755 157	6 669 116	8 524 143
NGL	458 962	881 752	1 977 349	3 410 308	4 606 858	5 621 974
WOM	236 179	443 113	944 710	1 542 605	2 040 969	2 511 469
NMS	81 900	155 856	324 391	483 110	618 084	785 524
BDS	8 480	16 246	47 651	69 107	91 400	112 524
GPS	7	7	608	2 887	5 311	5 390
WAT	262 745	507 158	1 170 362	2 177 128	3 098 221	3 910 482
Total	2 786 125	5 295 206	11 788 722	20 743 629	28 339 383	35 116 622
Unlabeled	466 238	887 200	1 966 564	3 427 563	4 639 251	5 710 571
Labeled + Unlabeled	3 252 363	6 182 406	13 755 286	24 171 192	32 978 634	40 827 193

Finally, feature scaling was performed for each data set. Mean and standard deviation were estimated for each feature on the training data set from the *classification* data set and then used to standardize the data on the different data sets (*training*, *validation*, *test* and *boundary*) [Kuhn and Johnson, 2019]. The standardization was performed with the Scikit-Learn function *StandardScaler* [Pedregosa et al., 2011].

5.2.3. Method set-up

Model implementation

The GP model was implemented using the GPyTorch library [Gardner et al., 2018]. We chose GPyTorch because it has several key advantages. First, Gpytorch is based on PyTorch [Paszke et al., 2019], a very popular deep learning framework, and inherits of the advantages of PyTorch. More precisely, it provides easy access to GPU acceleration. Second, it implements several approximation algorithms, such as the variational strategy defined by [Hensman et al., 2015] and presented in Chapter 4. It is also possible to use LMC proposed in Chapter 4 and fast kernel methods such as Toeplitz and Kronecker structure within the covariance matrix. Finally, GPytorch is quite easy to install and use. Other libraries such as GPy [The-GPyOpt-authors, 2016] or GPFlow [Matthews et al., 2017] can be used for large scale GP. However, they

do not have the same advantages as GpyTorch. Indeed, GPy uses Python and Numpy for all computations, thus there is no GPU acceleration. Besides, variational inference requires gradient propagation (autograd) which is not implemented in Numpy. GPflow, on the other hand, can perform GPU acceleration. However, it relies on TensorFlow which offers less flexibility than Pytorch.

Hyper-parameters are selected at the initialization of the GP model. Moreover, an appropriate initialization of the parameters can facilitate the optimization and help the model to converge faster. The influence of hyper-parameters selection and of parameters initialization on model performance is provided in Section 6.3. A summary of the selected values is given (as a reminder we have $d = 481$ and $d' = 2$):

- The number of g_l latent functions was selected with $L = C = 23$.
- The mean function was selected as a constant. For each latent GP g_l , the mean function was initialized with $\mu_l = 0$.
- Covariance function:
 - for λt -GP, one RBF function for each latent GP g_l with the following initialization for the length-scale: $\ell_{l\lambda t} = \sqrt{d}$.
 - for $\phi\lambda t$ -GPSC, $k_i^S(\mathbf{x}, \mathbf{x}')$ each latent GP g_l with the following initialization for the lengths-scales: $\ell_{l\lambda t} = \sqrt{d}$, $\ell_{l\phi} = \sqrt{d'}$. Concerning the output-scale, we have: $\sigma_{l\lambda t} = \ln(1 + \exp(\tilde{\sigma}_{l\lambda t}))$ and $\sigma_{l\phi\lambda t} = \ln(1 + \exp(\tilde{\sigma}_{l\phi\lambda t}))$ with $\tilde{\sigma}_{l\lambda t} = \tilde{\sigma}_{l\phi\lambda t} = 0$. $\tilde{\sigma}_{l\lambda t}$ and $\tilde{\sigma}_{l\phi\lambda t}$ are optimized instead of $\sigma_{l\lambda t}$ and $\sigma_{l\phi\lambda t}$, respectively (i.e. parameterization in log-scale to enforce positivity constraints during the learning step).
 - for $\phi\lambda t$ -GPPC, $k_i^P(\mathbf{x}, \mathbf{x}')$ each latent GP g_l with the following initialization: $\ell_{l\lambda t} = \sqrt{d}$ and $\ell_{l\phi} = \sqrt{d'}$.
- The same number of inducing points $M = 50$ was selected for each g_l . They were initialized with a random selection with the same set of inducing points.
- For each latent GP g_l , the variational distribution was defined as $q(g_l(\mathbf{Z}_l)) \sim \mathcal{N}_M(\mathbf{m}_l, \mathbf{S}_l)$ with the following initialization: $\mathbf{m}_l = \mathbf{0}$ and $\mathbf{S}_l = \mathbf{I}_M$.
- The mixing matrix \mathbf{A} of size $C \times C$ was initialized with random values drawn from a standard Gaussian distribution, $\mathbf{A} \sim \mathcal{N}(0, 1)$.

Competitive methods

Three different classification methods were defined as competitive methods:

1. **Random Forest (RF):** The RF Classifier from the Scikit-Learn library [Pedregosa et al., 2011] was used to train the RF model. Standard parameter settings were used: 100 trees with no maximum depth and the number of features considered for splitting at each leaf node was equal to the square root of the total number of features, as defined in [Inglada et al., 2017] and [Inglada et al., 2018].

Table 5.5.: Number of trainable parameters for each model in the global configuration classification.

Model	# of parameters
λt -GP	584 200
$\phi\lambda t$ -GPSC	586 569
$\phi\lambda t$ -GPPC	586 523
λt -MLP	143 579
$\phi\lambda t$ -MLP	144 612
λt -LTAE	239 521
$\phi\lambda t$ -LTAE	240 005

- Multi-layer Perceptron (MLP):** The **MLP** model was built with four hidden layers. The number of neurons in the first layer was the number of features divided by two ($d/2$ or $(d + d')/2$ i.e. 240 or 241) and in the last three layers: the number of classes multiplied by three ($C \times 3$ i.e. 69). The activation function used was the **Rectified Linear Unit (ReLU)**.
- Lightweight Temporal Self-Attention (LTAE):** In **LTAE**, temporal inputs were divided in channels distributed among several compact attention heads. Each head operated in parallel and extracted highly-specialized temporal features. These features were concatenated to create a single representation. A more detailed description and the parameters used from the **LTAE** model are given in [Garnot and Landrieu, 2020]. The implementation was based on the Pytorch library and was extracted from the `iota`² repository¹.

The first two methods do not take into account the spectro-temporal structure of the data, e.g., modifying the order of the temporal acquisitions would not change the behavior of the algorithm. The last one takes the temporal structure into account to process the SITS by using temporal positional encoding and attention mechanisms.

For each classification method previously defined, two different versions of each model were trained: λt -model and $\phi\lambda t$ -model. A λt -version was trained using only spectro-temporal features $\mathbf{x}_{\lambda t}$. A $\phi\lambda t$ -version was trained using spectro-temporal features $\mathbf{x}_{\lambda t}$ and spatial features \mathbf{x}_{ϕ} . The number of trainable parameters for each method in the *global* configuration classification is summarized in Table 5.5.

For **GP** and neural networks, the Adam optimizer was used. Optimizer parameters are given in Table 5.6. They were found by trial and error. The performance of each model in terms of classification accuracy for the two scenarios was computed using the **Overall Accuracy (OA)** and F-score, described in Appendix A.2.2. The results are provided in Chapter 6.

¹<https://framagit.org/iota2-project/iota2/-/tree/develop>

Table 5.6.: Parameter values for the Adam optimizer for GP, MLP and LTAE.

	GP	MLP	LTAE
Number of epochs E	100	300	100
Batch size β	1024	1000	1000
Learning rate η	1×10^{-3}	1×10^{-5}	1×10^{-5}

5.2.4. Map production

Land cover maps were produced using the `iota`² processing chain [Inglada et al., 2016] for both *stratification* and *global* configurations for all the studied models (λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE).

The map production was performed on two adjacent tiles: *T31TCJ* and *T31TDJ*. In *global* configuration, predictions for the map production were done using the model trained with the data set DS-A on the 27 tiles with the best OA computed over the 11 runs. In *stratification* configuration, predictions were done for each region with the best corresponding model. The results are provided in Chapter 6.

5.2.5. Feature reduction

The estimation of the IP involves a high number of parameters and is time-consuming: reducing the number of features could be beneficial for the convergence of the algorithm. Focusing on the λt -GP model, 584 200 trainable parameters are defined for this model, with 553 150 corresponding to the optimization of the IP, as defined in Table 5.5. In the following, we propose to study the influence on reducing the number of spectro-temporal features for the λt -GP model.

Standalone feature extractor

In standalone feature extraction, the pixel \mathbf{x}_i of size $d = TD$, with T the number of observations and D the number of spectral bands, is transformed into $\hat{\mathbf{x}}_i$ of reduced size thanks to an extractor module h . This reduced pixel $\hat{\mathbf{x}}_i$ is then given to the classifier f , as represented in Figure 5.6. The feature extraction is performed as a pre-processing step independent of the downstream classification task (thus the *standalone* adjective). Three different extractors are studied:

1. **Spectral reduction:** the reduction is performed in the spectral dimension independently from the temporal dimension. Instead of using D bands, we used D' bands. We propose to keep only the three spectral indices (i.e. NDVI, NDWI, Brightness), as they are already a combination of spectral bands. The hypothesis is that this can reduce the redundancy in the spectral bands. Thus, it permits to reduce to $3 \times 37 = 111$ spectro-temporal features.
2. **Temporal reduction:** the reduction is performed in the temporal dimension independently from the spectral dimension. Instead of using T temporal acquisitions, we used

T' temporal acquisitions. Different statistical indicators (e.g. mean per month) can be chosen instead of the whole set of interpolated dates, as described in Table 5.7. Some combinations permit to reduce the number of spectro-temporal features but some of them do not reduce this number (e.g. $624 > 481$). In the following, results for only one configuration (i.e. mean value for each month) are presented.

3. **Linear Discriminant Analysis (LDA)**: it is a conventional method used for dimensionality reduction in machine learning. It projects the TD dimensional feature space into a $C - 1$ dimension space that best separates the classes. Indeed, it maximizes the ratio of the between-classes variances and the within-classes variances [Rao, 1948]. By using LDA as extractor, the number of spectro-temporal features is reduced at most to $C - 1 = 22$. Even if the class labels are used, the LDA is a pre-processing step totally independent of the downstream classification task.

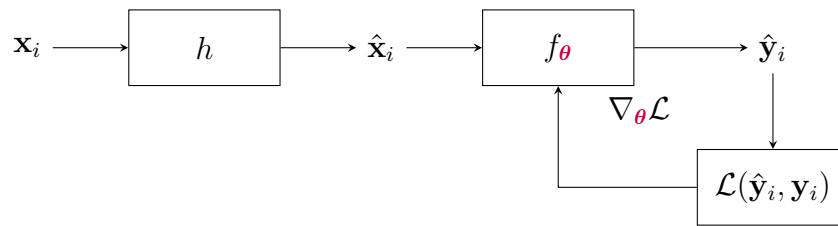


Figure 5.6: Feature extraction as a pre-processing. \mathbf{x}_i and $\hat{\mathbf{x}}_i$ are respectively the input pixel and the reduced pixel. h corresponds to the standalone feature extractor. f corresponds to the classifier (λt -GP model) and θ corresponds to its parameters. The loss \mathcal{L} is used to optimize θ and to minimize the error between the predicted class \hat{y}_i and the true class y_i .

Table 5.7.: Different combinations for the temporal reduction.

Frequency	Statistical indicators	Number of spectro-temporal features
Monthly	Mean	$13 \times 12 = 156$
Monthly	Mean, variance, min, max	$13 \times 12 \times 4 = 624$
Quarterly	Mean	$13 \times 4 = 52$
Quarterly	Mean, variance, min, max	$13 \times 4 \times 4 = 208$

End-to-end feature extractor

Like in standalone feature extraction, in end-to-end feature extraction, the pixel \mathbf{x}_i of size $d = TD$ is transformed into a pixel $\hat{\mathbf{x}}_i$ of reduced size. However, this reduction is learned for the classification task, as illustrated in Figure 5.7. Indeed, the parameters β of the extractor h are learned in order to minimize the classification loss. Two different end-to-end extractors are studied:

1. **Multilayer Perceptron (MLP)**: with two hidden layers. The first layer is composed of 300 neurons, the second of 200 neurons and the output layer is composed of 100 neurons. It can be written as:

$$h : \mathbb{R}^{TD} \rightarrow \mathbb{R}^{100}.$$

Different values were tested for the number of neurons in the hidden layers and in the output layer. This configuration was selected as it gave good performance results. The weights of the **MLP** corresponding to β are jointly optimized with the **SVGP**.

2. **Linear Projection:** as defined in [Constantin et al., 2021], we propose to represent the vector \mathbf{x}_i defined as

$$\begin{aligned}\mathbf{x}_i &= [\mathbf{x}_i(t_1), \dots, \mathbf{x}_i(t_T)] \\ &= [x_i^1(t_1), \dots, x_i^D(t_1), \dots, x_i^1(t_T), \dots, x_i^D(t_T)]^\top\end{aligned}$$

of size d by the matrix \mathbf{X}_i of size $D \times T$, such as:

$$\mathbf{X}_i = \begin{bmatrix} x_i^1(t_1) & \dots & x_i^1(t_T) \\ \dots & \dots & \dots \\ x_i^D(t_1) & \dots & x_i^D(t_T) \end{bmatrix}.$$

It allows us to take into account the spectro-temporal structure of the data. Two separate matrices are used for the linear projection. A first matrix $\mathbf{U} \in \mathbb{R}^{D \times D'}$ is used to reduce the spectral dimension and a second matrix $\mathbf{V} \in \mathbb{R}^{T \times T'}$ is used to reduce the temporal dimension:

$$\hat{\mathbf{X}}_i = \mathbf{U}^\top \mathbf{X}_i \mathbf{V}.$$

The extractor h can be written as:

$$h : \mathbb{R}^{D \times T} \rightarrow \mathbb{R}^{D' \times T'}$$

with $D' \ll D$ and $T' \ll T$ and with $T' = 10$ and $D' = 10$. Different values were tested for T' and D' and this configuration was chosen. Such as for the **MLP**, the number of spectro-temporal features is reduced to 100. The weights of the matrices corresponding to β are jointly optimized with the **SVGP**.

All the studied methods are summarized in Table 5.8. For all feature extraction methods, the number of trainable parameters used to optimize the **IP** is reduced compared to the baseline. For **LDA**, the reduction factor is around 15, whereas for **MLP**, it is close to 1.7. The number of features for **LDA** is significantly smaller than the other methods. Tests were carried out using a number of features similar to **LDA** but without good performances. The performance of each method in terms of classification accuracy is provided in Chapter 6.

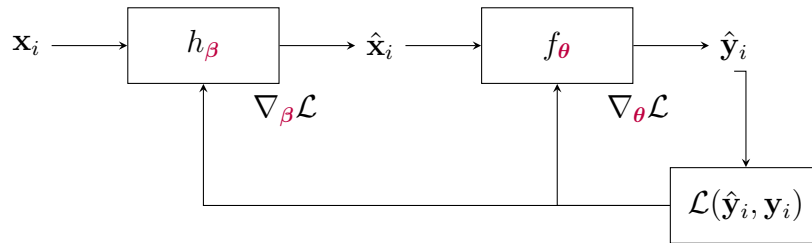


Figure 5.7: End-to-end feature extraction. In this case, β corresponds to the parameters of the end-to-end feature extractor. Moreover, the loss \mathcal{L} is used to optimize β and θ and to minimize the error between the predicted class \hat{y}_i and the true class y_i .

Table 5.8.: Comparison of the different methods used for the feature extraction. The baseline corresponds to the classifier (λt -GP model) without extractor. Spectral, Temporal and LDA correspond to the classifier (λt -GP model) combined with a feature extractor as a pre-processing. MLP and Linear Pro corresponds to the classifier (λt -GP model) combined with an end-to-end feature extractor. # of features corresponds to the number of features that are used by the classifier. # of parameters corresponds to the number of trainable parameters used to optimize the inducing points.

Name	Extractor	# of features	# of parameters
Baseline	None	481	553 150
Spectral	Spectral reduction	111	127 650
Temporal	Temporal reduction	156	179 400
LDA	LDA	22	35 904
MLP	MLP	100	339 300
Linear Pro	Linear projection	100	115 500

6.1. Comparison with competitive methods	162
6.1.1. Quantitative results	162
6.1.2. Qualitative results	170
6.2. Boundary study	174
6.2.1. Quantitative results	174
6.2.2. Qualitative results	175
6.3. Model evaluation	179
6.3.1. Hyper-parameters selection	179
6.3.2. Trainable parameters initialization	183
6.4. Analysis of the characteristics of the GP model	186
6.4.1. Posterior predictive distribution	186
6.4.2. Learned model parameters	189
6.5. Feature reduction	191

In the previous chapter, Chapter 5, the method based on GP as well as the experimental set-up were presented. In this chapter, the results associated are described. Firstly, the GP model is compared with competitive methods, both quantitative and qualitative results are provided. Then, a study of performance in areas between two eco-climatic regions is presented. Finally, the GP model is examined from several aspects.

6.1. Comparison with competitive methods

In this section, the GP model is compared to competitive methods both quantitatively and qualitatively in terms of classification accuracy and complexity. The studied models are: λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE. As a reminder, λt -model refers to a classification model using only the spectro-temporal features, whereas, $\phi\lambda t$ -model refers to a classification model using the spatial and the spectro-temporal features. The GP models are described in Section 5.2.3 and the competitive models (RF, MLP and LTAE) are described in Section 5.2.3.

6.1.1. Quantitative results

Classification metrics were computed using the *test* data set from the *classification* data set (c.f. Section 5.2.2) in both configurations: *stratification* and *global* (c.f. Section 5.2.1). A comprehensive description of the classification metrics is provided in Section A.2.2 in Appendix A. Classification metrics were averaged over the 11 runs of each model trained either with the DS-A or the DS-B *training* data set. Firstly, the global metrics are studied followed by the metrics per class. Then, confusion matrices are provided and finally, training and prediction times are considered.

Overall accuracy (OA)

The OA for each model trained with *training* data sets DS-A and DS-B is given in Figure 6.1. For all results, OA and mean F-score are very similar as all classes are well represented. Therefore, the mean F-score is presented in Figure B.1 in Appendix B.

With the data set DS-A, the LTAE achieves better performance, followed by GP, MLP and finally RF. The results are similar for the DS-B dataset, with a slight increase for each model. This result can be explained by the fact that more pixels are seen during training. By considering the *global* configuration with spatial information, in terms of OA, GP models are in average three points above RF models, one point above MLP models and one point below LTAE models.

For both data sets, all models benefit from the spatial information. Indeed, the OA is increased by less than one point for RF and MLP models and between one and three points for GP and LTAE models. GP models have the highest improvement, specifically in the *global* configuration.

For both data sets, only λt -GP and RF models have better results with the *stratification* configuration compared to the *global* one, as illustrated in Figure 6.1. For all the other models, better performances are achieved with the *global* configuration.

The Wilcoxon rank-sum test [Wilcoxon, 1945] was used to assess the statistical significance of the observed differences in terms of OA over the MC runs, for each pair of classification

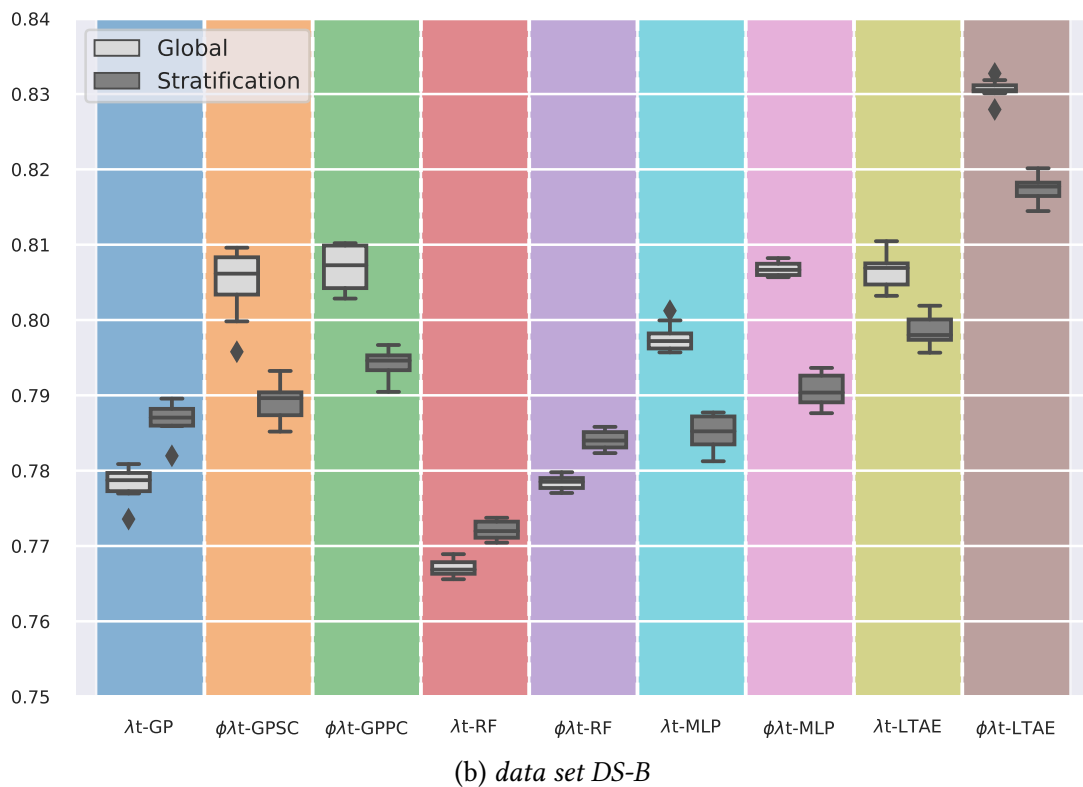
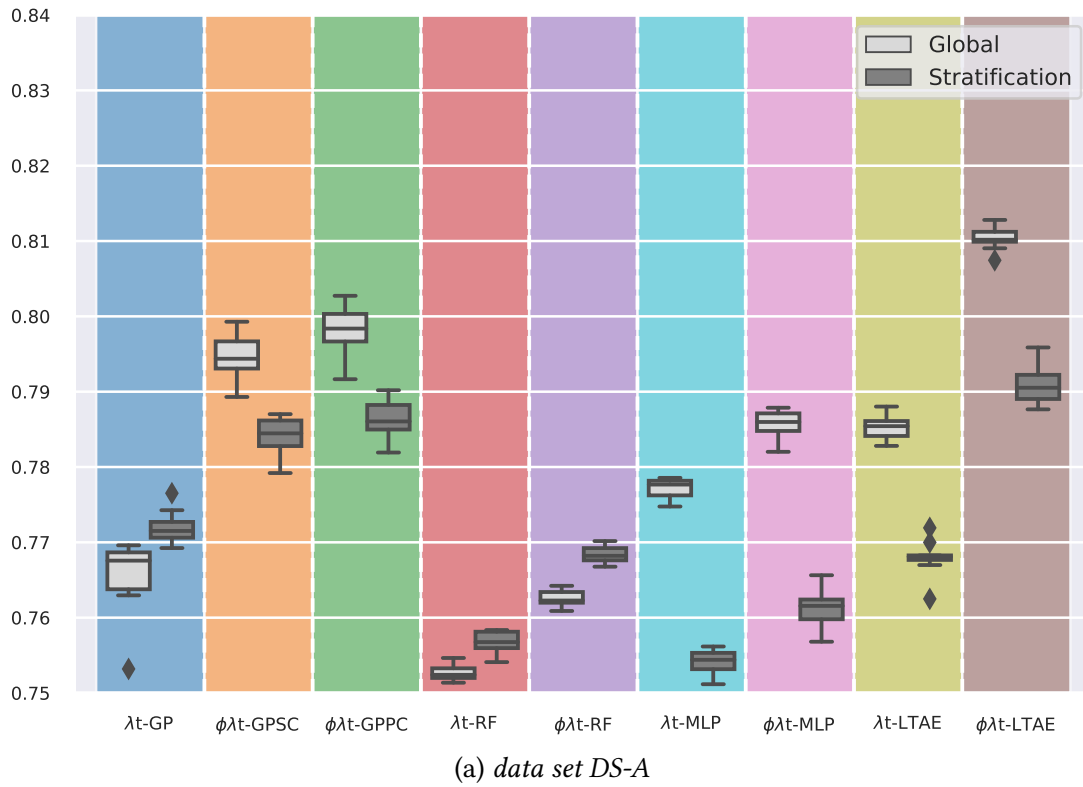


Figure 6.1: Boxplots of the OA for each studied model. Both data sets DS-A and DS-B are considered for each configuration: global and stratification.

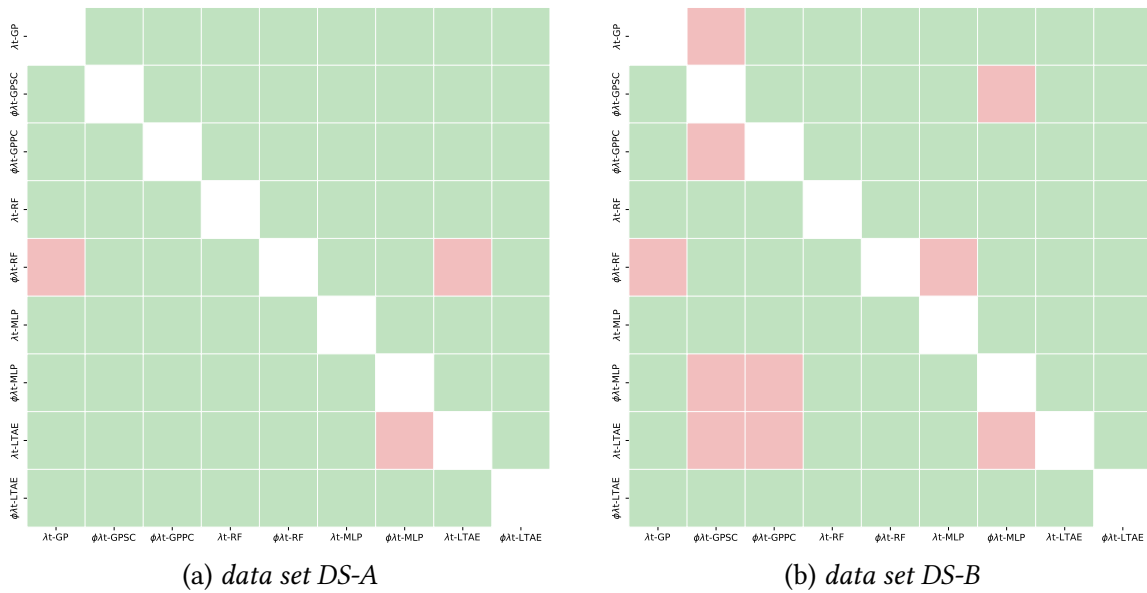


Figure 6.2: Wilcoxon rank-sum tests results for each competitive method. Both data sets DS-A and DS-B are considered for each configuration: global and stratification. Red cells indicate that the observed differences in terms of OA over the MC runs between the two classification methods are not significantly different. The null hypothesis was rejected at a significance level of $\alpha = 0.01$. Green cells correspond to significant observed differences. The cells above the diagonal of the table contain Wilcoxon test results for the stratification configuration, while cells below the diagonal contain the results for the global configuration.

methods. The null hypothesis was rejected at a significance level of $\alpha = 0.01$. Figures 6.2a and 6.2b show results obtained for the data set DS-A and DS-B, respectively.

With the data set DS-A, for the *stratification* configuration, all the results are significantly different, except between $\phi\lambda t$ -RF and λt -LTAE. Similar results are found for the *global* configuration, except between λt -GP and $\phi\lambda t$ -RF and between λt -LTAE and $\phi\lambda t$ -MLP.

For the larger training data set DS-B, for the *stratification* configuration, some results are not significantly different, such as λt -GP with $\phi\lambda t$ -GPSC. Moreover, $\phi\lambda t$ -GPSC is not significantly different from $\phi\lambda t$ -MLP and finally, $\phi\lambda t$ -RF with λt -MLP. In *global* configuration, more results are not significantly different in terms of classification accuracy.

The Wilcoxon tests confirm all the previous observations, particularly regarding the influence of spatial information and configurations.

F-score, precision and recall per class

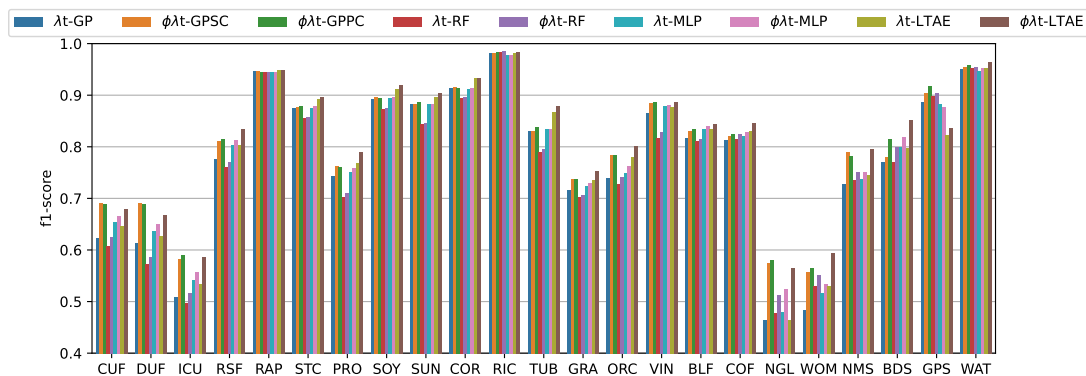
The F-score by class for each model trained with *training* data sets DS-A and DS-B on *global* and *stratification* configurations are presented in Figure 6.3. The precision and recall per class are presented in Figures B.2 and B.3 and in Tables B.1 and B.2 in Appendix B. The nomenclature of the classes is presented in Table 3.3.

With the data set DS-A, in *global* configuration, for each class the **LTAE** model is above all models except for the **GPS** class. Indeed, for the **GPS** class, both **LTAE** accuracies (spatial and non spatial) are below all models (**GP**, **RF**, **MLP**). This result can be explained by the fact that the class **GPS** has a very low number of pixels (less than 8 000 pixels for DS-A in *global*).

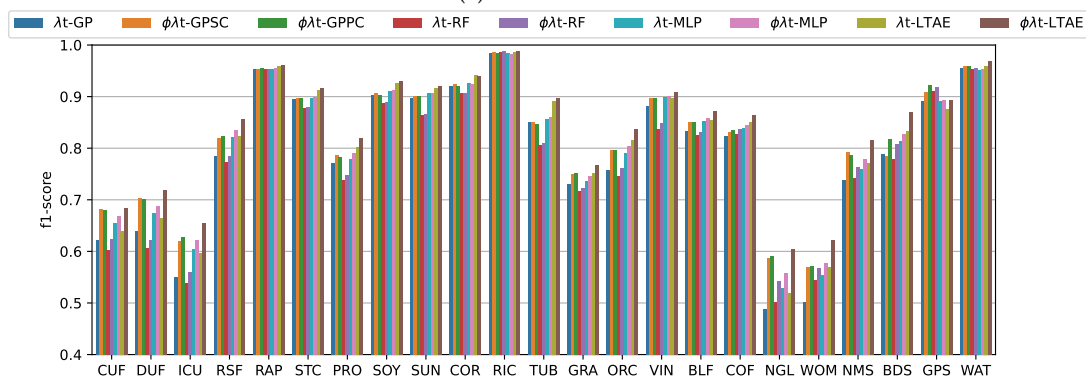
Moreover, for each class, the **MLP** model is above the **RF**, except for the classes **RIC**, **WOM**, **GPS** and **WAT**. Finally, for each class, the **GP** model is well above the **RF**, except for the **RIC** class for which the models have very similar values. Such as the **GPS** class, the **RIC** class has a very low number of pixels (8 000 pixels for DS-A in *global*).

For all classes, in both configurations, the spatial information allows for all methods to improve the F-score, there is no reduction in performance (except for the **WAT** class in *stratification* configuration with DS-A with **GP** models). Adding the spatial information enables to improve the results. This confirms the results found in the previous section with the **OA** or the mean F-score.

Considering all methods except **RF** models, the majority of classes perform better in *global* configuration than in *stratification* configuration. Indeed, for both $\phi\lambda t$ -GPSC and $\phi\lambda t$ -GPPC models, all the classes perform better in *global* configuration than in *stratification* configuration, except the following classes: **CUF**, **ICU**, **RSF**, **WOM**, **NMS** and **GPS**. For the $\phi\lambda t$ -LTAE model, only the **GPS** class performs better in *stratification* configuration than in *global* configuration. In contrast, for **RF** models, the majority of the classes has a better F-score in *stratification* configuration than in *global* configuration, except the following classes: **STC**, **PRO**, **SOY**, **SUN**, **COR**, **RIC** and **TUB**. Therefore, such as with the **OA** or the mean F-score, for the majority of the models, the *global* configuration enables better classification accuracy than the *stratification* configuration. In the following, we will focus on the confusions between classes.



(1) data set DS-A



(2) data set DS-B

(a) *global* configuration

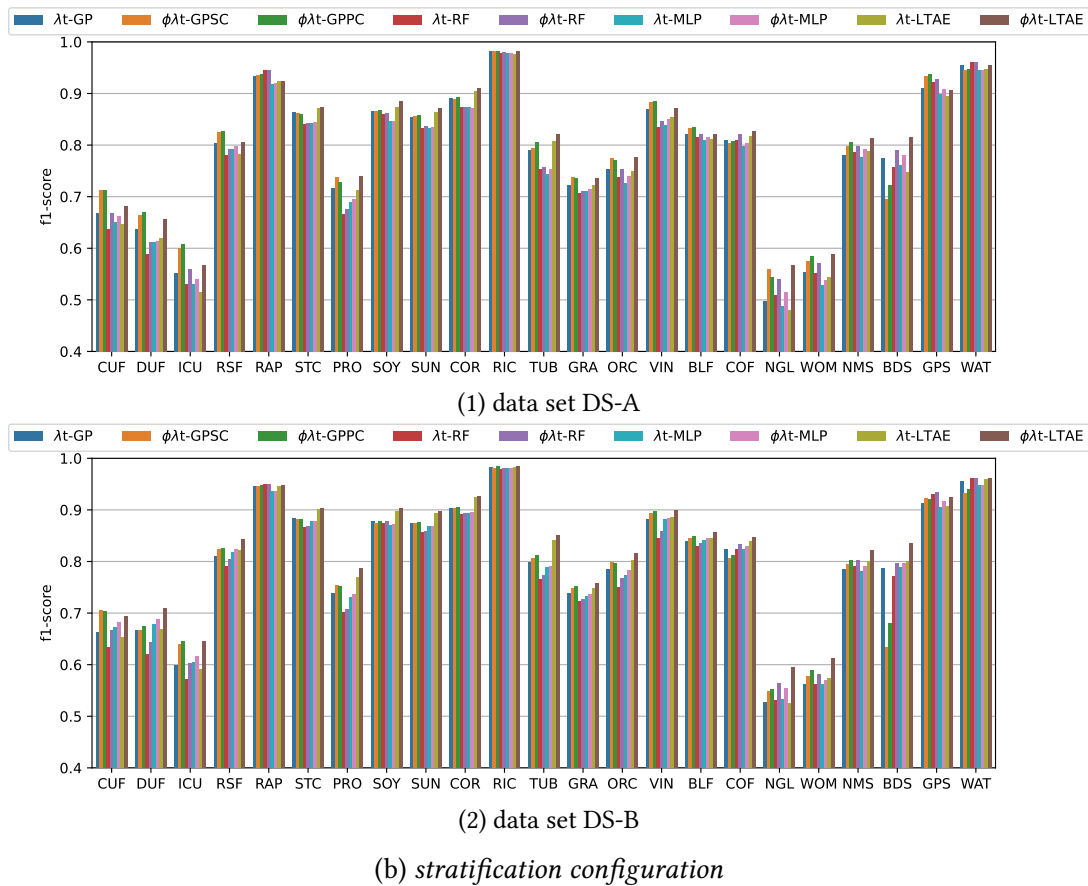


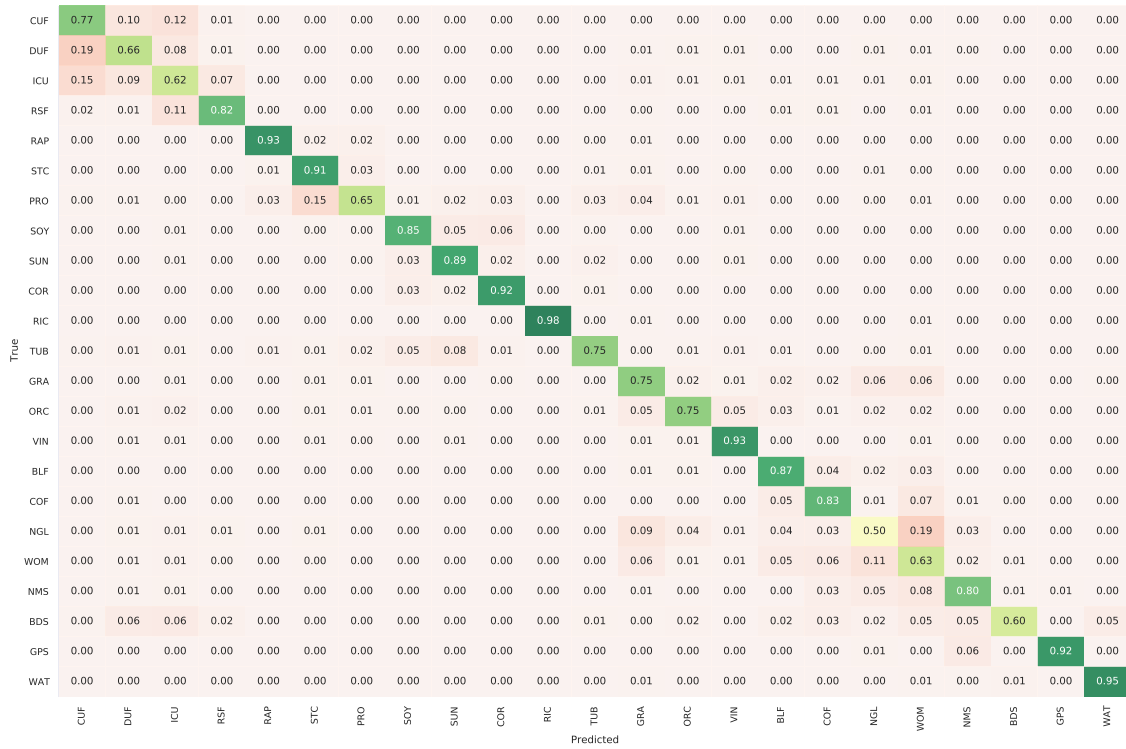
Figure 6.3: Barplots of the F-score per class for each studied model. Both data sets DS-A and DS-B are considered for each configuration: global and stratification.

Confusion matrices

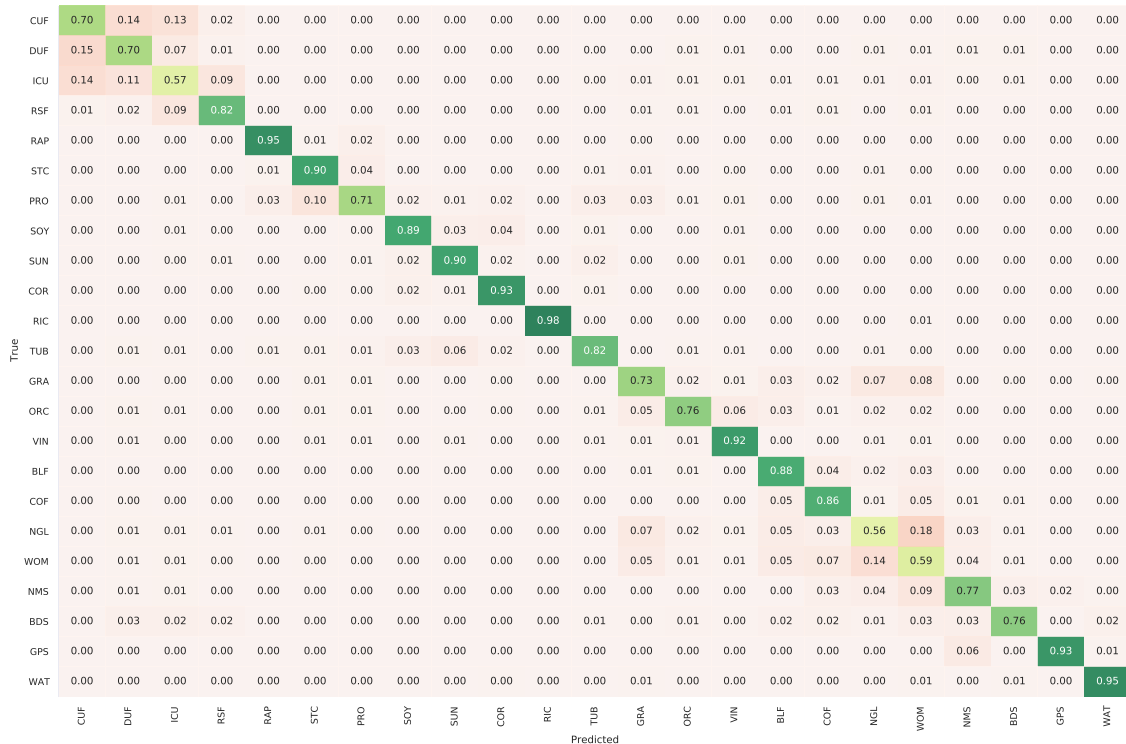
For both configurations, *stratification* and *global*, normalized confusion matrices are produced for each model with the data set DS-A. The normalization is applied over the true labels i.e. the sum of each row is equal to one. Figures 6.4a and 6.4b represent the confusion matrices for $\phi\lambda t$ -GPPC and $\phi\lambda t$ -LTAE, respectively. We have chosen to present only these two models, as they correspond to the models with the best performance. The confusion matrices for all models are presented in Figure B.4 in Appendix B.

For all models, there are confusions between CUF, DUF and ICU classes. These classes correspond to urban classes and are difficult to discriminate at Sentinel-2 pixel size units using only pixel-wise information. Moreover, for all models, some confusions are also found between the two classes: NGL and WOM. They are also very similar classes with a continuous gradient between woody and non-woody vegetation, difficult to discriminate.

As illustrated in Figure 6.4a, for $\phi\lambda t$ -GPPC, the confusion between the PRO and STC classes (PRO predicted as STC) is reduced from 0.15 in the *stratification* configuration to 0.1 in the *global* configuration. Same results are found for the $\phi\lambda t$ -LTAE, as illustrated in Figure 6.4b.

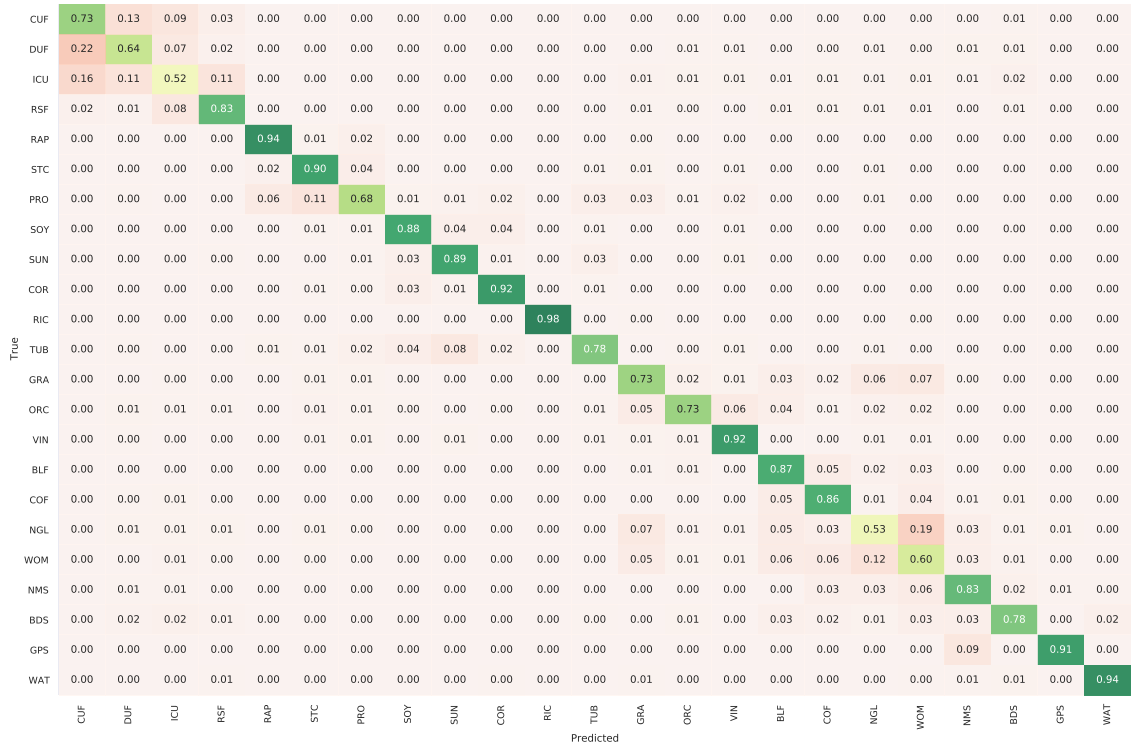


(1) stratification configuration



(2) global configuration

(a) $\phi\lambda t$ -GPPC model



(1) stratification configuration



(2) global configuration

(b) $\phi\lambda t$ -LTAE model

Figure 6.4: Normalized confusion matrices for $\phi\lambda t$ -GPPC and $\phi\lambda t$ -LTAE. Only the data set DS-A is considered for each configuration: global and stratification. The confusion matrices for the other models are presented in Figure B.4 in Appendix B.

Training and prediction times

The averaged training and prediction times computed for each region and each model over the 11 runs are presented in Table 6.1. To process the **RF** models, 20 CPU (100 GB of RAM) were available. For **GP** and **DL** models, 1 NVIDIA Tesla V100 GPU was used. In the *global* configuration, the **MLP** models have the shortest training time per epoch followed by **LTAE** and finally **GP**. Concerning prediction times, they all have the same order of duration except the **GP** models that are 10 times higher. **GP** are more demanding, because of the **MC** sampling for the variational posterior. Increasing the data set size, from DS-A to DS-B, leads to an increase of the training times per epoch of around 3.5 times for the **MLP**, **LTAE** and **GP** models. However, for **RF** models, it is increased by almost 5 times. For all models except the **GP** models, the spatial information has no effect on training or prediction times. The spatial **GP** models have higher training times than the non spatial **GP** model. Indeed, in *global* configuration, the training time of $\phi\lambda t$ -GPSC is 1.5 times higher than λt -GP. Parameters for the additional covariance function (i.e. spatial one) need to be computed. Moreover, $\phi\lambda t$ -GPSC have longer training times compared to $\phi\lambda t$ -GPPC, as it involves more computations. The sum of training times for all regions (row "1 + ... + 8" in Table 6.1) is lower than the training time in *global* configuration. Moreover, the advantage of *stratification* configuration is that the learning for each region can be performed in parallel but with a loss in terms of accuracy.

Table 6.1.: Averaged training (*T*) and prediction (*P*) times for each model and each region (mean in seconds averaged over 11 runs). The averaged training time is for one epoch except for the **RF** models for which it is the full time. The white line corresponds to the data set DS-A and the gray line corresponds to the data set DS-B.

Region	Time	Model									
		λt -GP	$\phi\lambda t$ -GPSC	$\phi\lambda t$ -GPPC	λt -RF	$\phi\lambda t$ -RF	λt -MLP	$\phi\lambda t$ -MLP	λt -LTAE	$\phi\lambda t$ -LTAE	
1	T	2.8	4.5	3.1	15.4	14.9	0.6	0.6	1.4	1.4	
		8.6	14.3	9.8	58.3	56.9	1.9	1.9	4.4	4.4	
	P	22.1	25.1	22.4	1.3	1.2	0.8	0.9	1.2	1.2	
		35.2	37.2	29.1	2.4	2.2	1.5	1.5	2.1	2.2	
2	T	3.0	5.0	3.4	16.5	16.5	0.7	0.7	1.6	1.6	
		10.5	17.4	12.0	72.3	71.1	2.4	2.4	5.5	5.5	
	P	25.4	30.7	25.2	1.4	1.4	1.0	1.0	1.5	1.5	
		42.9	40.7	47.7	3.2	2.7	1.9	1.9	2.7	2.7	
3	T	2.2	3.3	2.4	11.2	11.1	0.5	0.5	1.2	1.2	
		6.6	10.5	7.5	43.0	42.8	1.6	1.6	3.8	3.7	
	P	19.5	23.8	19.8	1.0	1.0	0.8	0.8	1.1	1.1	
		31.2	31.2	32.4	2.0	1.6	1.3	1.3	1.9	1.9	
4	T	3.1	5.0	3.4	17.3	16.8	0.7	0.7	1.6	1.6	
		12.0	19.9	13.7	84.4	82.8	2.7	2.7	6.3	6.3	
	P	27.0	32.7	27.3	1.6	1.5	1.1	1.1	1.6	1.6	
		50.3	54.4	50.3	3.5	3.4	2.2	2.2	3.1	3.1	
5	T	3.2	5.3	3.6	17.6	17.1	0.7	0.7	1.6	1.6	
		11.0	18.3	12.8	74.3	72.1	2.4	2.4	5.6	5.6	
	P	26.5	32.1	25.9	1.6	1.5	1.0	1.0	1.5	1.5	
		46.3	49.5	49.6	3.0	2.7	1.9	1.9	2.8	2.8	
6	T	3.2	5.3	3.6	16.8	16.6	0.7	0.7	1.6	1.6	
		12.0	19.7	13.6	75.4	74.8	2.6	2.6	6.1	6.0	
	P	27.7	33.5	27.5	1.7	1.6	1.1	1.1	1.6	1.6	
		52.0	57.5	49.6	3.3	3.0	2.1	2.1	3.0	3.0	

Region	Time	Model									
		λt -GP	$\phi\lambda t$ -GPSC	$\phi\lambda t$ -GPPC	λt -RF	$\phi\lambda t$ -RF	λt -MLP	$\phi\lambda t$ -MLP	λt -LTAE	$\phi\lambda t$ -LTAE	
7	T	2.9	4.0	3.2	15.0	14.9	0.6	0.6	1.5	1.5	
		10.9	15.4	12.3	70.2	69.2	2.5	2.5	5.8	5.8	
	P	24.4	27.9	23.9	1.3	1.2	1.0	1.0	1.4	1.4	
		47.7	49.0	47.1	2.7	2.7	1.9	1.9	2.8	2.8	
8	T	3.2	5.4	3.7	17.7	17.8	0.7	0.7	1.6	1.6	
		12.5	20.7	14.2	85.2	83.1	2.7	2.7	6.4	6.3	
	P	28.9	33.9	28.3	1.7	1.6	1.1	1.1	1.6	1.6	
		56.4	57.9	55.0	3.6	3.6	2.2	2.2	3.1	3.1	
1 + ... + 8	T	23.5	37.6	26.4	127.5	125.6	5.2	5.2	12.1	12.1	
		84.2	136.1	95.9	563.0	552.7	18.7	18.8	43.8	43.7	
	P	201.5	239.8	200.3	11.6	10.9	8.0	8.0	11.5	11.5	
		362.1	377.4	360.7	23.7	22.1	15.1	15.1	21.7	21.6	
Global	T	25.0	41.2	28.7	189.4	184.2	6.6	6.6	13.5	13.5	
		87.1	154.1	105.7	894.6	888.1	25.1	25.3	50.1	50.2	
	P	170.2	232.6	215.9	18.2	17.5	10.8	10.9	14.2	14.2	
		276.4	336.5	314.0	47.1	43.7	21.1	21.1	27.4	27.2	

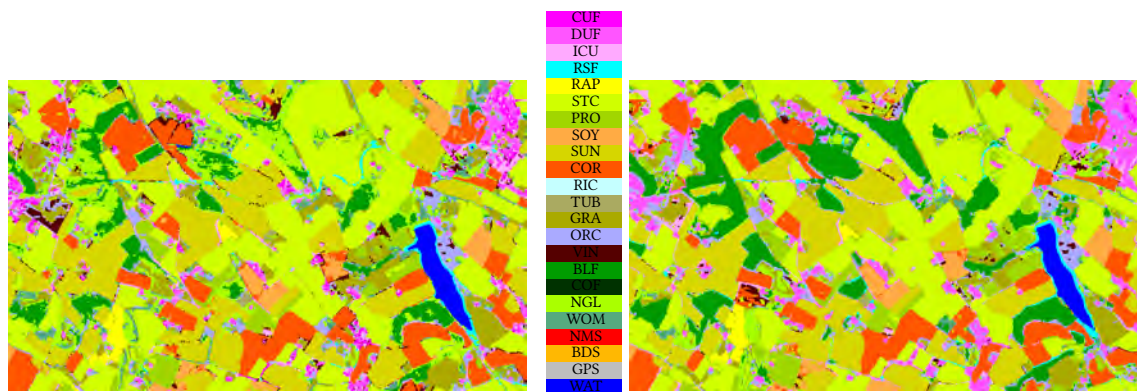
To conclude, for all models, the spatial information improves the classification performances. The *stratification* configuration is only beneficial for the **RF** models. For all other models, the *global* configuration performs better. **GP** models have very good performances, above **RF** and **MLP** models, and just below **LTAE** models. In **GP** models, the spatio and spectro-temporal structure is taken into account. Moreover, massive training data sets are handled without any problem, in particular thanks to the approximations used for the **GP** models.

6.1.2. Qualitative results

In the previous section, the quantitative assessment has been conducted, the qualitative study will now follow. For this purpose, land cover maps were produced using the *iota*² processing chain (a custom code has been developed for my PhD). Land cover maps were generated for all studied models on two different tiles: *T31TCJ* and *T31TDJ* in both configurations. The land cover maps are available for download: [10.5281/zenodo.7077887](https://doi.org/10.5281/zenodo.7077887).

Figure 6.5 represents a land cover map obtained on an agricultural area around Toulouse with all the studied models in both configurations (*stratification* and *global*). The study area is relatively flat (between 180 and 260 meters).

In *global* configuration, with **GP** models, land cover maps are more homogeneous (with less salt and pepper classification noise [Hirayama et al., 2019]) when the spatial information is added, as illustrated in Figures 6.5a and 6.5c. In contrast, for **MLP** and **LTAE** models, the spatial information does not reduce the salt and pepper classification noise. In **GP** models, with the addition of the spatial information, the main structures of the map are clearly represented (i.e. crop field border). Indeed, the classification map does not exhibit rounded borders as it is often the case with **CNN** models [Stoian et al., 2019b].



(1) stratification configuration

(2) global configuration

(a) λt -GP model

(1) stratification configuration

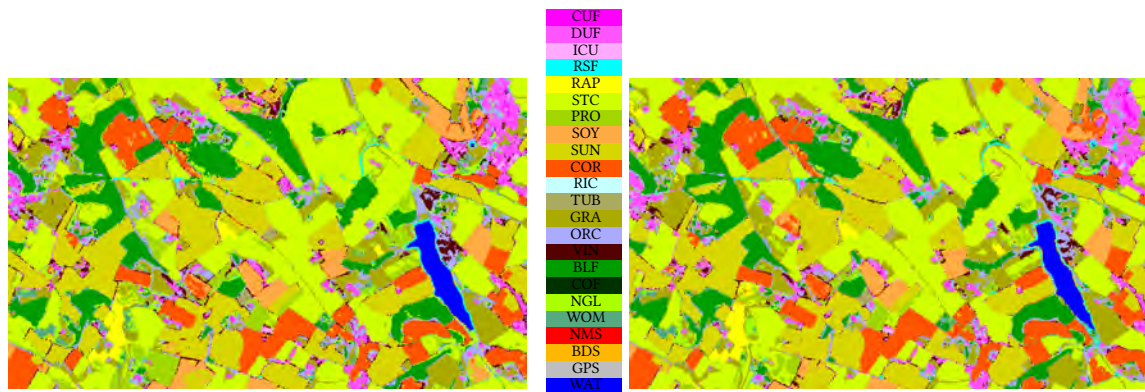
(2) global configuration

(b) $\phi\lambda t$ -GPSC model

(1) stratification configuration

(2) global configuration

(c) $\phi\lambda t$ -GPPC model

(1) *stratification configuration*(2) *global configuration*(d) λt -RF model(1) *stratification configuration*(2) *global configuration*(e) $\phi\lambda t$ -RF model(1) *stratification configuration*(2) *global configuration*(f) λt -MLP model(1) *stratification configuration*(2) *global configuration*(g) $\phi\lambda t$ -MLP model

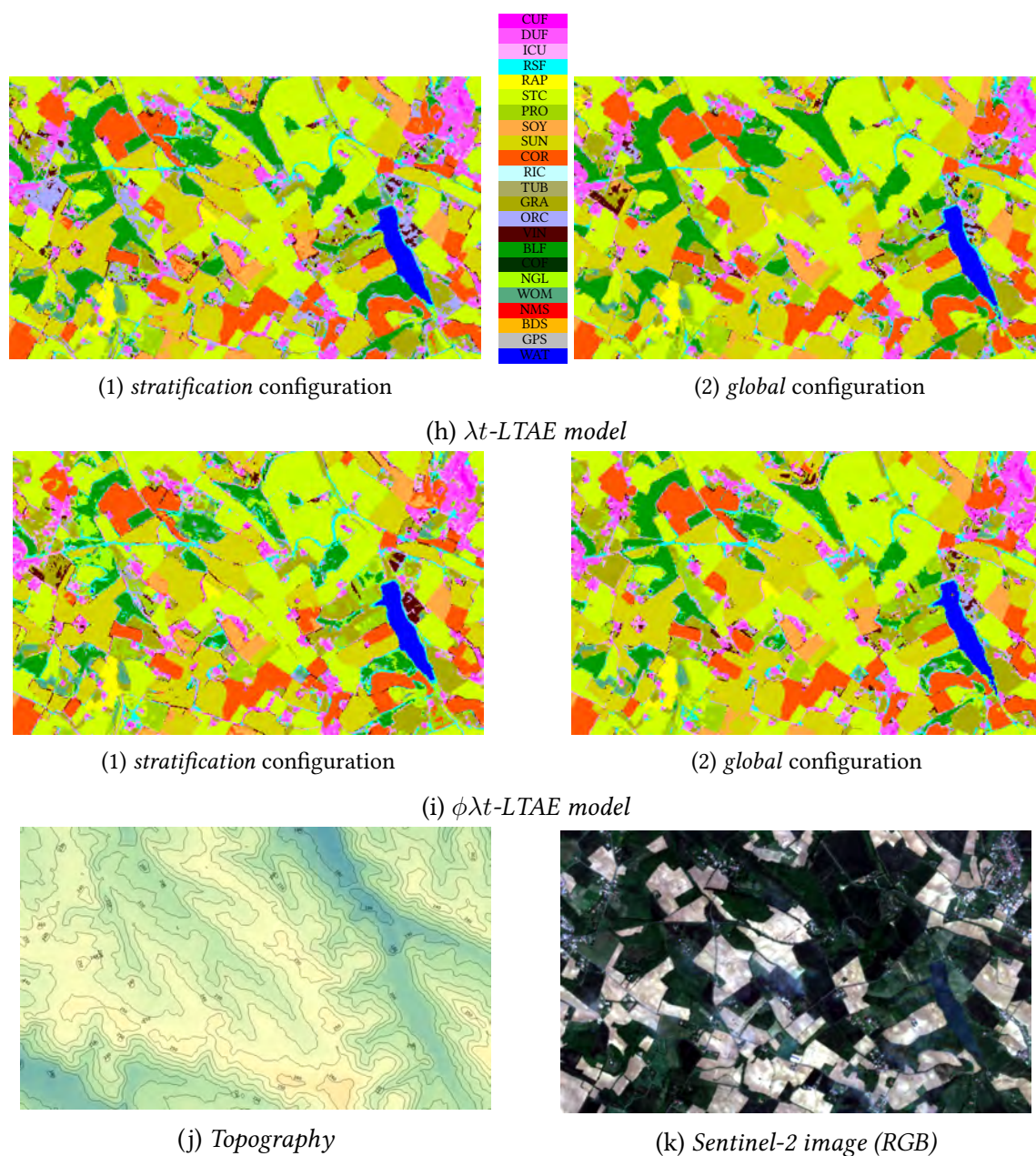


Figure 6.5: Comparison of the land cover maps obtained with each model in both configurations (stratification and global) on an agricultural area around Toulouse (tile T31TCJ). Topography information (30-meter STRM, contours are in meters) and Sentinel-2 image (RGB) (acquisition date: 15/05/18) of the specific zone are provided. Some clouds are visible in the Sentinel-2 image. The studied area is relatively flat (min: 180m, max: 260m). There are different types of landscape: towns, crop fields, a lake, forests, etc.

6.2. Boundary study

In the *stratification* configuration, models were trained independently: no constraints were imposed between model behavior for adjacent regions during processing. The goal of this section is to evaluate the continuity of the predictions inside the boundary zone. Like the previous section, the studied models are: λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE.

6.2.1. Quantitative results

In this section, we propose to study the continuity of the predictions inside the boundary zone thanks to the number of agreements i.e. the number of pixels assigned to the same class by both models in the boundary zone. Thus, in each boundary zone, all the pixels (i.e. *boundary* data set) were predicted by the two models surrounding this zone. The percentage of agreement corresponds to the number of agreements divided by the total number of pixels.

Table 6.2 represents the percentage of agreements for different boundary sizes: $B \in \{100, 200, 500, 1000\}$. The size of the boundary has no influence, results are similar. Besides, this percentage was calculated for unlabeled pixels and also for labeled pixels which are correctly predicted. **RF** models have higher agreement than other models for both unlabeled pixels and labeled pixels correctly predicted. For unlabeled pixels, **RF** models are followed by λt -LTAE with a difference of around four points. For labeled pixels correctly predicted, **RF** models have similar values with $\phi\lambda t$ -GPPC. For **RF** models, the two models tend to agree with each other i.e. there is a some continuity in the predictions. However, the continuity in predictions does not mean that the predictions are correct.

To evaluate the agreement between regions w.r.t. correctly classified pixels, we computed the **OA** on labeled pixels for different boundary sizes. Table 6.3 represents the **OA** for both *global* and *stratification* configurations. For all methods, the **OA** in the *global* configuration is above the *stratification* one. The difference between both configurations is only two points for **RF** models and more than four points for **DL** methods. In Section 6.1, we found that the *stratification* configuration performed better than the *global* configuration for **RF** models. In boundary zone, it is no longer the case. One hypothesis that could explain this result is that in each eco-climatic region, the model will tend to specialize. Therefore, between eco-climatic regions, models are quite different. Moreover, the pixels in the boundary zone are the furthest from the centroid of the region: they correspond less to the region and they may consequently be less accurately classified. For all models, the performances increase when the spatial information is added. For all models, better performance results are found in *global* configuration. Same results were found in the previous section for **GP**, **MLP** and **LTAE** models in a more general context (i.e. not in boundary areas). In the following, a visual assessment will be conducted.

Table 6.2.: Averaged percentage of agreement (between two adjacent models) for different sizes of boundary zones ($B \in \{100, 200, 500, 1000\}$) (mean % \pm standard deviation computed with 11 runs). Comparison between unlabeled pixels and labeled pixels correctly predicted. Models were trained in stratification configuration.

B	Pixels	λt -GP	$\phi \lambda t$ -GPSC	$\phi \lambda t$ -GPPC	λt -RF	$\phi \lambda t$ -RF	λt -MLP	$\phi \lambda t$ -MLP	λt -LTAE	$\phi \lambda t$ -LTAE
100	unlabeled	66.3 \pm 0.7	64.6 \pm 1.0	66.2 \pm 0.8	72.6 \pm 0.5	72.1 \pm 0.4	65.2 \pm 0.6	64.4 \pm 0.6	68.4 \pm 0.6	66.0 \pm 0.8
	labeled correctly predicted	66.6 \pm 0.6	68.5 \pm 0.6	69.8 \pm 0.6	69.2 \pm 0.4	70.5 \pm 0.8	64.9 \pm 0.4	65.6 \pm 0.4	66.4 \pm 0.4	68.2 \pm 0.5
200	unlabeled	66.2 \pm 0.7	64.7 \pm 0.9	66.2 \pm 0.8	72.6 \pm 0.5	72.1 \pm 0.3	65.1 \pm 0.6	64.4 \pm 0.6	68.3 \pm 0.6	66.0 \pm 0.9
	labeled correctly predicted	66.5 \pm 0.6	68.3 \pm 0.6	69.5 \pm 0.6	69.2 \pm 0.4	70.5 \pm 0.4	64.9 \pm 0.4	65.6 \pm 0.4	66.3 \pm 0.4	68.1 \pm 0.5
500	unlabeled	66.0 \pm 0.7	64.5 \pm 0.9	66.1 \pm 0.8	72.5 \pm 0.5	71.8 \pm 0.3	65.0 \pm 0.5	64.2 \pm 0.6	68.2 \pm 0.6	65.9 \pm 0.8
	labeled correctly predicted	66.6 \pm 0.5	68.2 \pm 0.5	69.4 \pm 0.5	69.3 \pm 0.4	70.5 \pm 0.3	65.1 \pm 0.4	65.8 \pm 0.4	66.4 \pm 0.4	68.2 \pm 0.5
1000	unlabeled	65.8 \pm 0.7	64.3 \pm 0.9	65.8 \pm 0.8	72.3 \pm 0.5	71.8 \pm 0.3	64.8 \pm 0.6	64.0 \pm 0.6	68.0 \pm 0.6	65.7 \pm 0.8
	labeled correctly predicted	66.9 \pm 0.5	68.5 \pm 0.5	69.7 \pm 0.5	69.4 \pm 0.4	70.8 \pm 0.4	65.4 \pm 0.3	66.2 \pm 0.3	66.8 \pm 0.4	68.6 \pm 0.5

Table 6.3.: Averaged OA computed on labeled pixels for different sizes of boundary zones ($B \in \{100, 200, 500, 1000\}$) (mean % \pm standard deviation computed with 11 runs). Comparison between global configuration and stratification configuration.

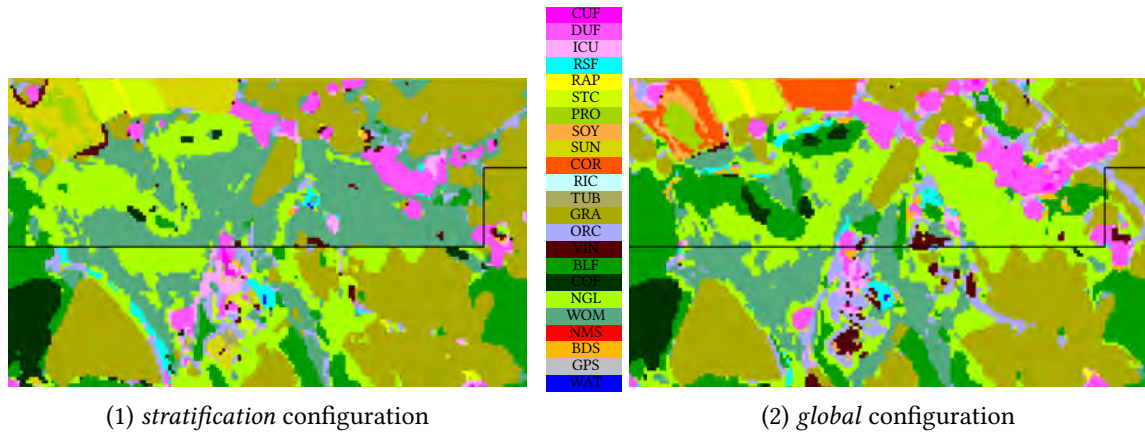
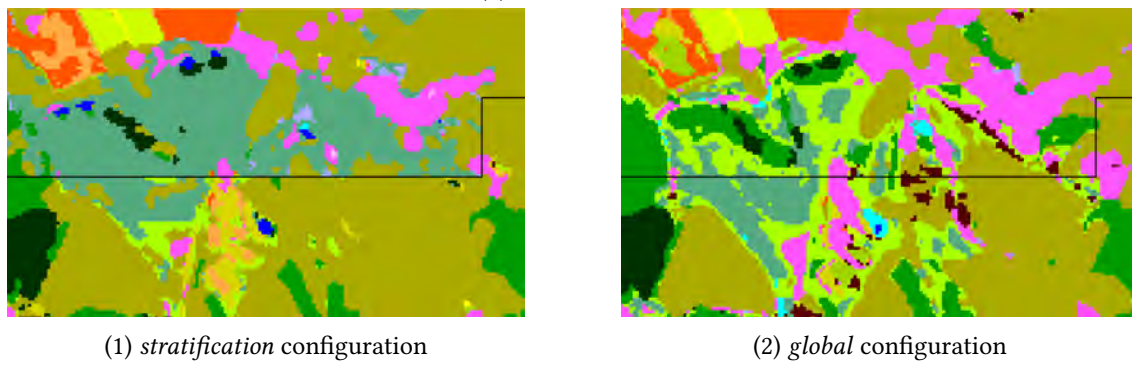
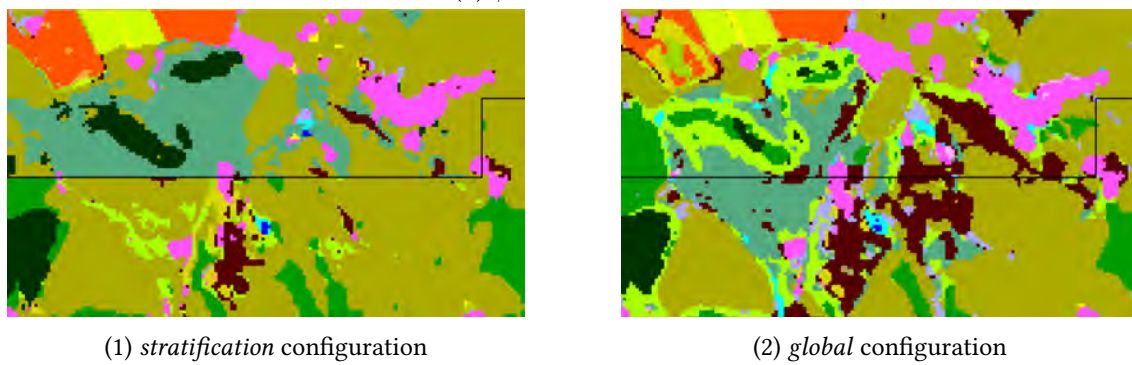
B	Pixels	λt -GP	$\phi \lambda t$ -GPSC	$\phi \lambda t$ -GPPC	λt -RF	$\phi \lambda t$ -RF	λt -MLP	$\phi \lambda t$ -MLP	λt -LTAE	$\phi \lambda t$ -LTAE
100	global	77.1 \pm 0.6	79.3 \pm 0.7	79.9 \pm 0.6	77.7 \pm 0.1	78.7 \pm 0.4	77.8 \pm 0.2	78.8 \pm 0.1	78.0 \pm 0.4	80.6 \pm 0.2
	stratification	74.6 \pm 0.4	76.5 \pm 0.4	77.3 \pm 0.4	75.6 \pm 0.2	76.8 \pm 0.7	73.1 \pm 0.3	74.0 \pm 0.2	74.2 \pm 0.3	76.2 \pm 0.3
200	global	77.0 \pm 0.6	79.2 \pm 0.6	79.8 \pm 0.6	77.6 \pm 0.1	78.7 \pm 0.1	77.8 \pm 0.3	78.7 \pm 0.1	78.0 \pm 0.4	80.6 \pm 0.2
	stratification	74.6 \pm 0.4	76.5 \pm 0.3	77.2 \pm 0.3	75.6 \pm 0.2	76.9 \pm 0.2	73.2 \pm 0.3	74.0 \pm 0.2	74.1 \pm 0.3	76.2 \pm 0.3
500	global	77.3 \pm 0.6	79.3 \pm 0.7	79.9 \pm 0.6	77.7 \pm 0.1	78.7 \pm 0.1	77.9 \pm 0.2	78.9 \pm 0.1	78.1 \pm 0.3	80.6 \pm 0.2
	stratification	74.8 \pm 0.3	76.4 \pm 0.4	77.2 \pm 0.3	75.9 \pm 0.2	77.0 \pm 0.2	73.6 \pm 0.2	74.4 \pm 0.2	74.4 \pm 0.3	76.4 \pm 0.3
1000	global	77.5 \pm 0.6	79.6 \pm 0.7	80.1 \pm 0.6	77.8 \pm 0.1	79.0 \pm 0.1	78.1 \pm 0.2	79.1 \pm 0.1	78.3 \pm 0.3	80.9 \pm 0.2
	stratification	75.4 \pm 0.3	77.0 \pm 0.4	77.7 \pm 0.2	76.2 \pm 0.3	77.5 \pm 0.2	74.1 \pm 0.3	75.0 \pm 0.2	74.8 \pm 0.3	76.8 \pm 0.3

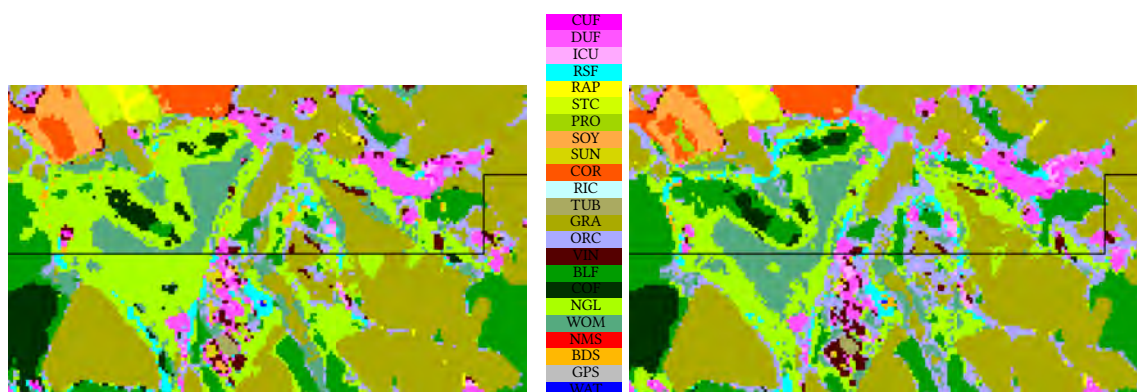
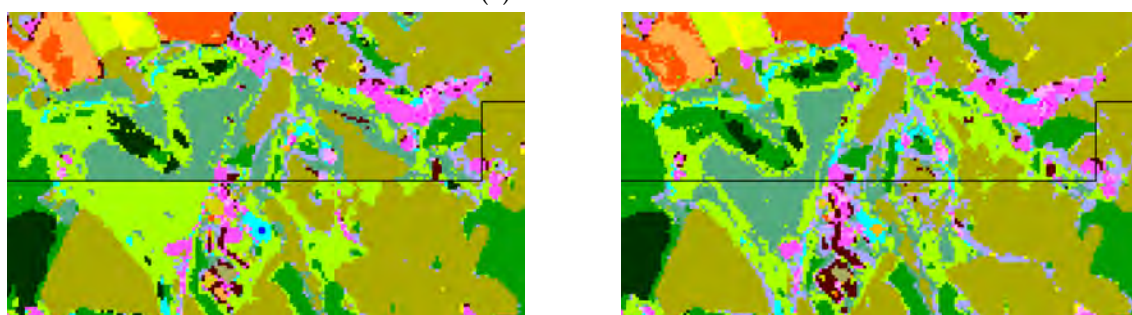
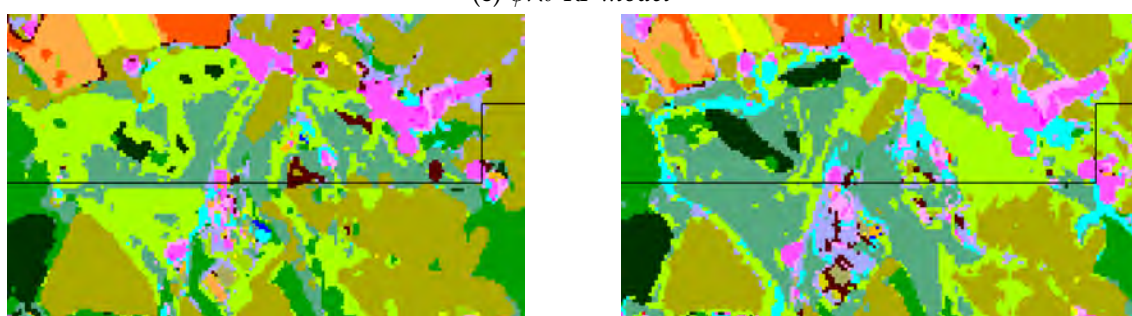
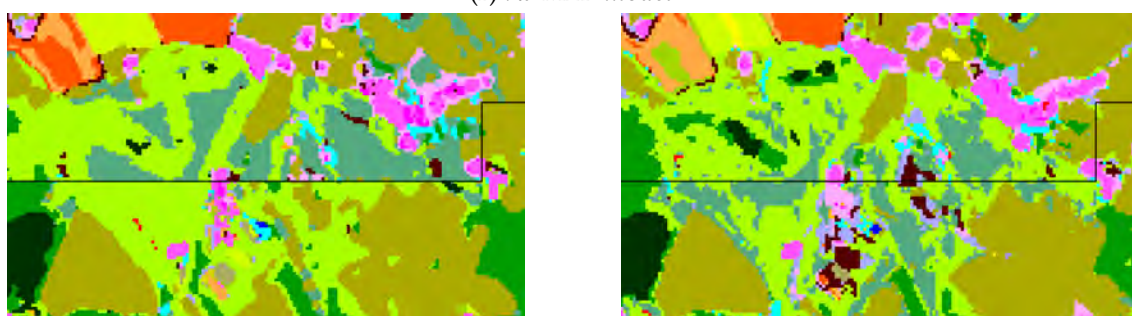
6.2.2. Qualitative results

In the following, we are going to study land cover maps produced with the *iota*² processing chain and more precisely in areas between two eco-climatic regions. As a reminder, all the land cover maps produced on both tiles are available here: [10.5281/zenodo.7077887](https://zenodo.org/record/7077887).

Figure 6.5 represents a land cover map between two eco-climatic regions computed with each studied model. For all models, in *stratification* configuration, some discontinuities in predictions between two eco-climatic regions are found, even for RF models as illustrated in Figures 6.6d and 6.6e. In contrast, in *global* configuration, there is no discontinuity for all methods. This confirms the quantitative results found previously.

In *stratification* configuration, adding the spatial information appears to improve the continuity of prediction for $\phi \lambda t$ -GPSC, as illustrated in Figure 6.6c. In this example, adding the spatial information does not improve the continuity of prediction for $\phi \lambda t$ -GPPC, as illustrated in Figure 6.6b. Note that this is a special case, since Table 6.3 states the opposite. In *stratification* configuration, for MLP and LTAE models, adding the spatial information did not improve the prediction continuity. In *global* configuration, for RF models, adding the spatial information did not change the land cover map, as illustrated in Figures 6.6d and 6.6e. In *global* configuration, for other models, adding the spatial information has an impact on land cover maps.

(a) λt -GP model(b) $\phi \lambda t$ -GPSC model(c) $\phi \lambda t$ -GPPC model

(d) λt -RF model(e) $\phi\lambda t$ -RF model(f) λt -MLP model(g) $\phi\lambda t$ -MLP model

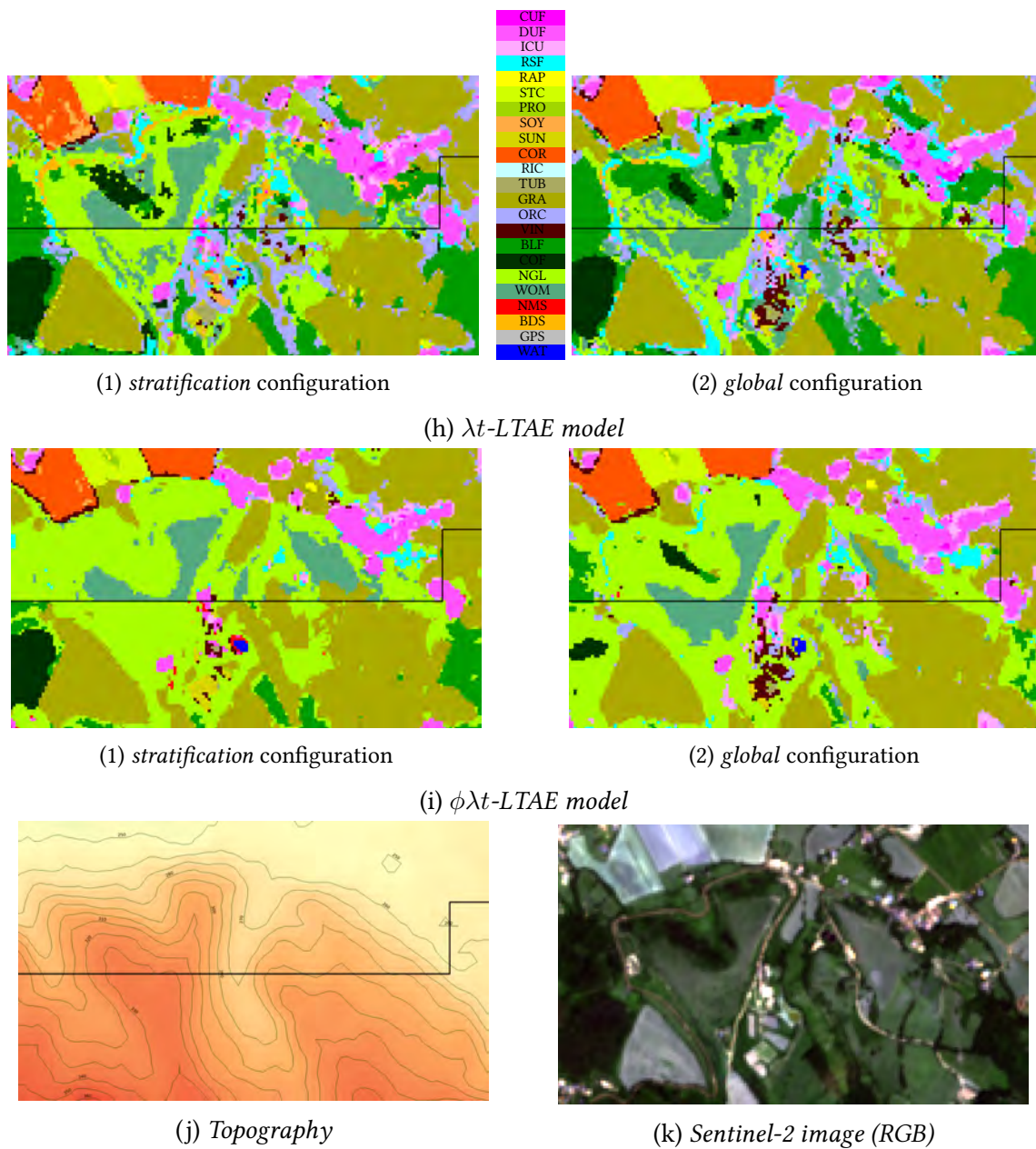


Figure 6.6: Comparison of the land cover maps obtained with each model in both configurations (stratification and global) on an boundary zone between two eco-climatic regions (tile $T31TDJ$). Topography information (30-meter STRM, contours are in meters) and Sentinel-2 image (RGB) (acquisition date: 15/06/18) of the specific zone are provided.

6.3. Model evaluation

6.3.1. Hyper-parameters selection

In the following, we study the influence of hyper-parameters selection on model performance. Results were obtained with the λt -GP model trained on *global* configuration with a smaller data set. Indeed, to simplify the computations, the training and test data sets are composed of 92 000 and 230 000 pixels, respectively. The data sets are balanced: 4 000 pixels per class for the training data set and 10 000 pixels per class for the test data set.

Number of inducing points (IP)

The number M of IP is selected at the initialization of the model. If the number M of IP is large and close to the size of the training set N , fewer approximations are made. Thus, a more accurate approximate representation of the posterior is provided [Leibfried et al., 2020]. However, the larger the number, the higher are the computational complexity and the memory requirements. Indeed, M has a significant impact on the total number of trainable parameters, as described in Table 5.2. Hence, a compromise between complexity and accuracy needs to be made.

In the literature, few studies have been conducted on the choice of the number of points relative to the number of training inputs N and the number of features d [Seeger et al., 2003], [Titsias, 2009], [Azzimonti et al., 2016]. In general, the choice is purely made to reduce computational complexity, and a small number of inducing points are taken. However, the optimal number of IP depends on many factors: the form of the covariance function, the size of the training inputs, the number of features, the structure of the data, etc. Galy-Fajou and Opper [Galy-Fajou and Opper, 2021] proposed a bound on the expected number of IP by making some assumptions about the data (i.e. distribution). This bound is computed for a regression case with a RBF kernel. We did not apply this method because our data did not verify those assumptions.

Different values were tested: $M \in \{30, 50, 100, 250\}$, as illustrated in Figure 6.7. Increasing the number of inducing points slightly increases performance, but more importantly, significantly increases learning time. A compromise has been made between accuracy and training times. Finally, the value selected was $M = 50$. This number was used for each region but also in the *global* configuration. A major advantage is that our approach gives better results in *global* configuration and does not require more points than in *stratification* configuration.

Number of latent processes

The number of g_l latent functions L is also selected a priori. As stated in [Liu et al., 2022], taking $L > C$, increases significantly the model complexity and the number of trainable parameters (c.f. Table 5.5) and can lead to over-fitting. In contrast, taking $L < C$, leads to a very little flexibility and can lead to under-fitting.

In this work, different values were tested: $L \approx C/2 = 11$, $L = C = 23$ and $L = 2 \times C = 46$, as illustrated in Figure 6.8. With $L = 11$, the performances are severely degraded. Nevertheless, between $L = 23$ and $L = 46$, the OA is very similar. However, the training times is almost 1.5 times greater for $L = 46$ than for $L = 23$. Therefore, we decided to select the number of g_l latent functions with $L = C = 23$.

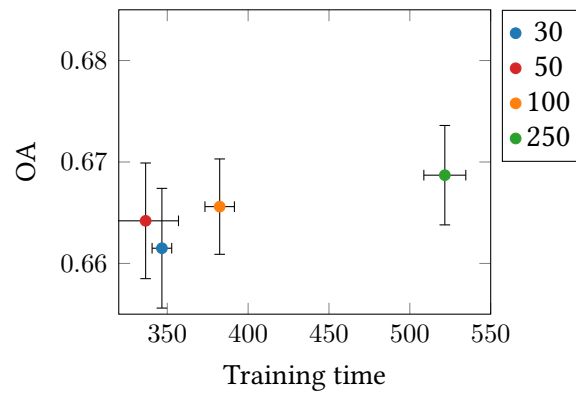


Figure 6.7: Comparison between OA and training times in seconds for different number of inducing points: $M \in \{30, 50, 100, 250\}$. The averaged values and the standard deviations are computed over 9 runs. The OA is computed over 230 000 pixels on the 27 tiles with the λt -GP model. The training time corresponds to the training of the λt -GP model over 92 000 pixels on the 27 tiles.

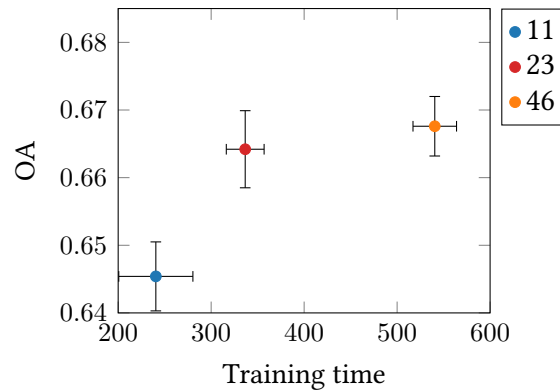


Figure 6.8: Comparison between OA and training times in seconds for different number of latent process: $L \in \{11, 23, 46\}$. The averaged values and the standard deviations are computed over 9 runs. The OA is computed over 230 000 pixels on the 27 tiles with the λt -GP model. The training time corresponds to the training of the λt -GP model over 92 000 pixels on the 27 tiles.

Number of MC samples for the prediction

The number of draws for the MC sampler is usually different between the training step and the prediction step. As explained in the previous chapter, one draw was selected for the training step. For the prediction step, ten draws were selected. These are the default values in Gpytorch.

We studied the influence of the number of draws for the MC sampler for the prediction step. In this work, different values were tested: {10, 50, 100}. In addition to the averaged OA and the averaged prediction time, the averaged percentage of agreement was computed. The percentage of agreement corresponds to the number of pixels with the same class membership over two different runs divided by the total number of pixels. The averaged percentage of agreement was computed for all the various combinations of runs.

The OA does not change significantly as a function of the number of draws, while the prediction time is slightly increased, as illustrated in Figure 6.9. For 10 draws, the averaged percentage of agreement is equal to 96.46%, for 50 draws it is equal to 98.38% and for 100 draws it is equal to 98.84%. Basically, it means that with more draws, the different runs agree more with each other. However, it does not indicate that they correctly predicted the class membership. Indeed, we just showed that a larger number of draws does not significantly improve the OA.

Figure 6.10 represents a land cover map with different number of draws ({10, 50, 100}) for three different runs. As the averaged percentage agreement showed previously, a larger number of draws allows to have more similar predictions. Indeed, as illustrated in Figures 6.10g, 6.10h and 6.10i, with 100 draws, the predictions of the land cover maps are very similar. However, with 10 draws, there are more differences in predictions across runs, as illustrated in Figures 6.10a, 6.10b and 6.10c. The agreement between runs increases with the number of draws, which is very important for the reproducibility of the results and more precisely in an operational context. However, in our case, we are focusing on quality metrics and more precisely on accuracy. Therefore, in the following, we decided to keep the value 10, as there is almost no impact on the OA.

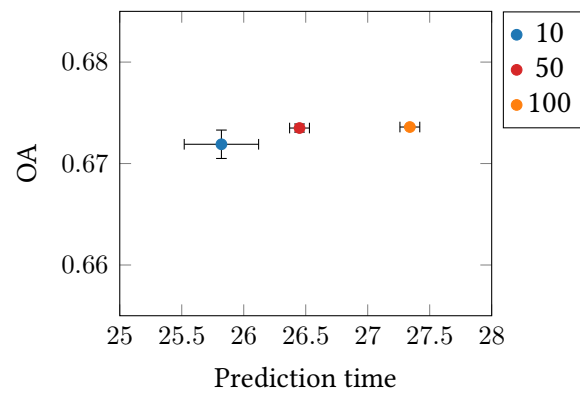


Figure 6.9: Comparison between OA and prediction times in seconds for different number of draws: $\{10, 50, 100\}$. The averaged values and the standard deviations are computed over 9 runs. The OA and prediction time are computed over 230 000 pixels on the 27 tiles with the λ -GP model.

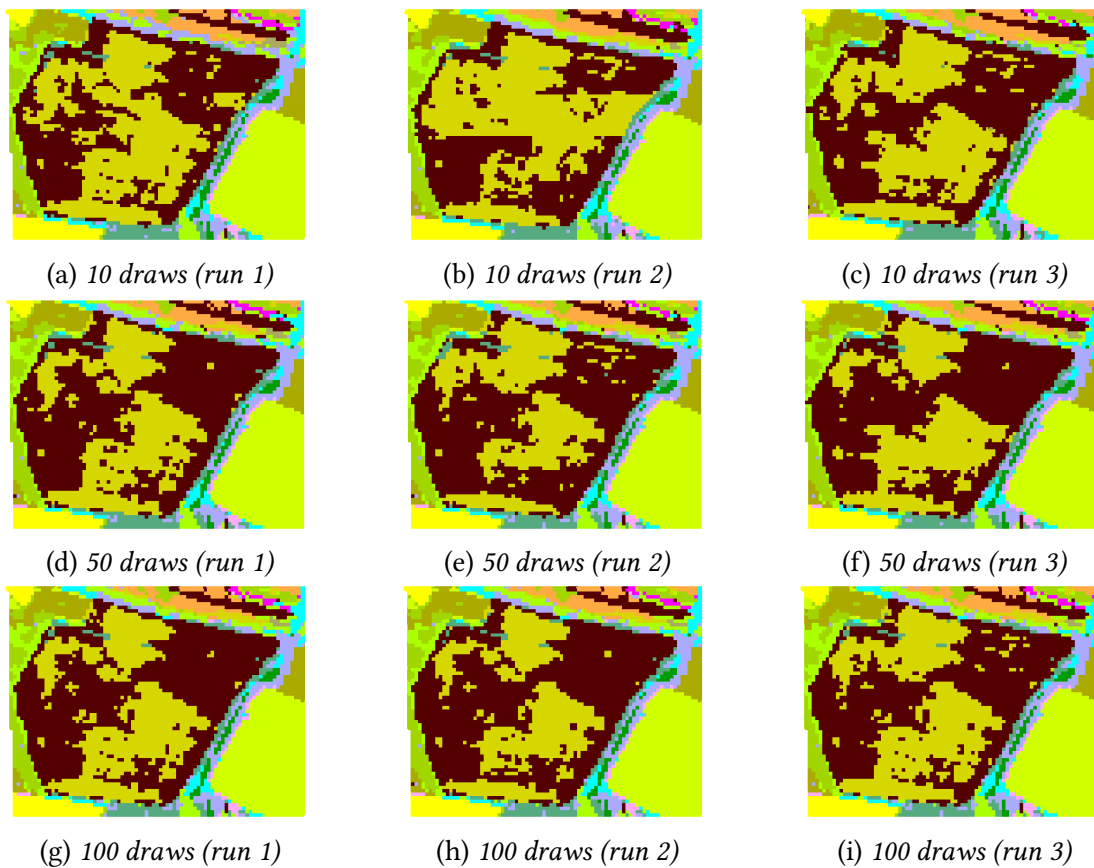


Figure 6.10: Land cover map with different number of draws ($\{10, 50, 100\}$) for three different runs.

6.3.2. Trainable parameters initialization

An appropriate initialization of the parameters can facilitate the optimization and help the model to converge faster. In this section, we investigate different initializations for the trainable parameters: the values of the inducing points \mathbf{Z}_l , the parameters of the covariance function and the values of the mixing matrix \mathbf{A} .

Inducing points (IP) values

In the literature, IP are usually initialized from a subset of the training data set or using k-means clustering [Hensman et al., 2015]. I have co-supervised a Master's internship where the objective was to assess the effect of different initializations for the IP. Several methods for the initialization of IP from the training set were investigated:

- Random selection without constraint;
- Random selection with the same number of pixels for each class;
- K-means clustering with no constraint;
- K-means clustering with the same number of pixels for each class.

For each method, two cases were considered: same initialization for each g_l latent process or different initialization for each g_l latent process.

Results from this study are provided in Figure 6.11. They were produced using the λt -GP model on the data set DS-A on *global* configuration. The results are only provided for the *global* configuration as it performs better for GP model than the *stratification* configuration (c.f. Section 6.1). The OA is very similar between all methods. The configuration "random by class" with different initialization appears to have the best results: good OA with small dispersion and with outliers relatively close. However, this result does not particularly stand out and the results between the methods remain relatively similar. Therefore, we decided to keep the simplest initialization to implement: random selection without constraint and same set of inducing points for each g_l .

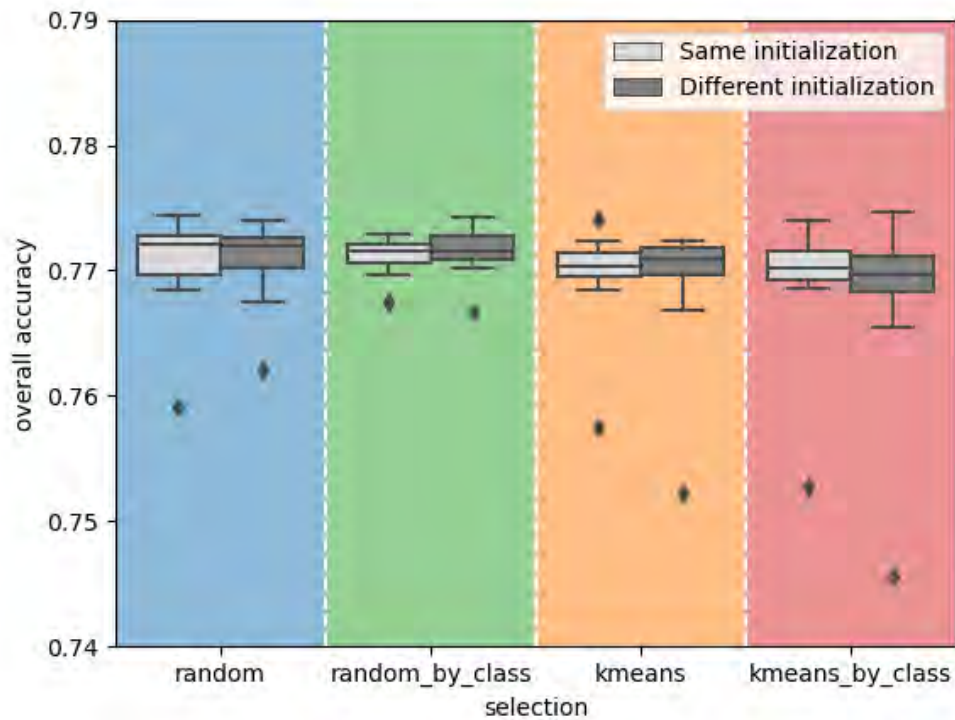


Figure 6.11: Boxplot of the OA (computed over 11 runs) for different methods of the selection of inducing points. *random*: random selection with no constraint ; *random_by_class*: random selection with the same number of pixels for each class ; *kmeans*: *k*-means clustering with no constraint and *kmeans_by_class*: *k*-means clustering with the same number of pixels for each class. For each method, two cases are considered: same initialization for each latent process or different initialization for each latent process. Results were produced with the data set DS-A in global configuration.

Covariance function parameters

The covariance function for each latent GP g_l is based on the RBF covariance function, defined by the following equation:

$$\sigma^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right). \quad (6.1)$$

Two parameters can be initialized for this RBF covariance function: ℓ , the length-scale and σ , the output-scale. By default, Gpytorch initializes this covariance function with the following values:

- $\ell = \ln(1 + \exp(\tilde{\ell}))$ with $\tilde{\ell} = 0$ and
- $\sigma = \ln(1 + \exp(\tilde{\sigma}))$ with $\tilde{\sigma} = 0$.

$\tilde{\ell}$ and $\tilde{\sigma}$ are optimized instead of directly ℓ and σ , respectively. With this formula, ℓ and σ are always positive and it is easier to optimize $\tilde{\ell}$ and $\tilde{\sigma}$ that are centered around zero. In our case, we study different combinations of kernels (see Equations (5.9) and (5.10)). Values for the initialization of the length-scale ℓ were studied. The length-scale was initialized with either the mean value of the Euclidean distance between IP or the square root of the features dimension [Fauvel, 2007, Chapter 1]. Several initialization combinations between the spatial and the spectro-temporal covariance functions were studied. Results showed that the best combination is the length-scale initialized with the square root of its dimensions for both covariance functions, such as:

- $\ell_{\lambda t} = \sqrt{d}$ for the spectro-temporal covariance function with $d = 481$, and
- $\ell_{\phi} = \sqrt{d'}$ for the spatio covariance function with $d' = 2$.

The initialization of the output-scale was not modified from the default value.

Mixing matrix values

By default, Gpytorch initializes the mixing matrix \mathbf{A} with random values drawn from a standard Gaussian distribution, $\mathbf{A} \sim \mathcal{N}(0, 1)$. Some tests were made by initializing the mixing matrix with an identity matrix. With this initialization, each output f_c only depends on one latent GP function g_l . However, there was no improvement in the performances with this initialization. Thus, the random initialization was kept.

To conclude, with all the selected values for the initialization, we did not have any convergence problems. Around selected values, the optimization always converged.

6.4. Analysis of the characteristics of the GP model

In the following, the posterior predictive distribution of the GP model is studied as well as the spatial location of the inducing point values and the values of the mixing matrix. Results were obtained with $\phi\lambda t$ -GPPC model trained on *global* configuration.

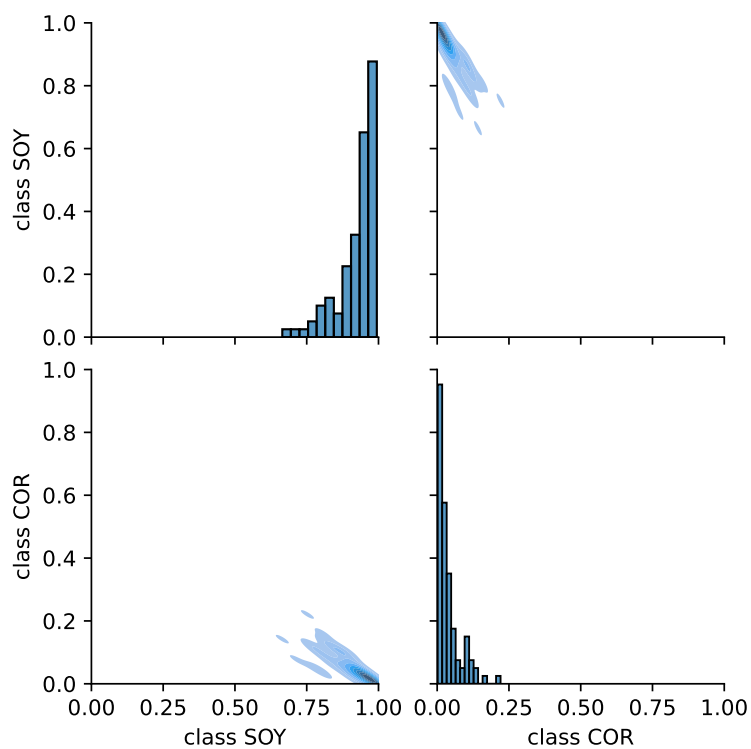
6.4.1. Posterior predictive distribution

The posterior predictive distribution is not Gaussian and has to be estimated with MC sampling. For each sample, the class membership probabilities are computed by averaging the random draws. The class with the highest average value is selected as the predicted class.

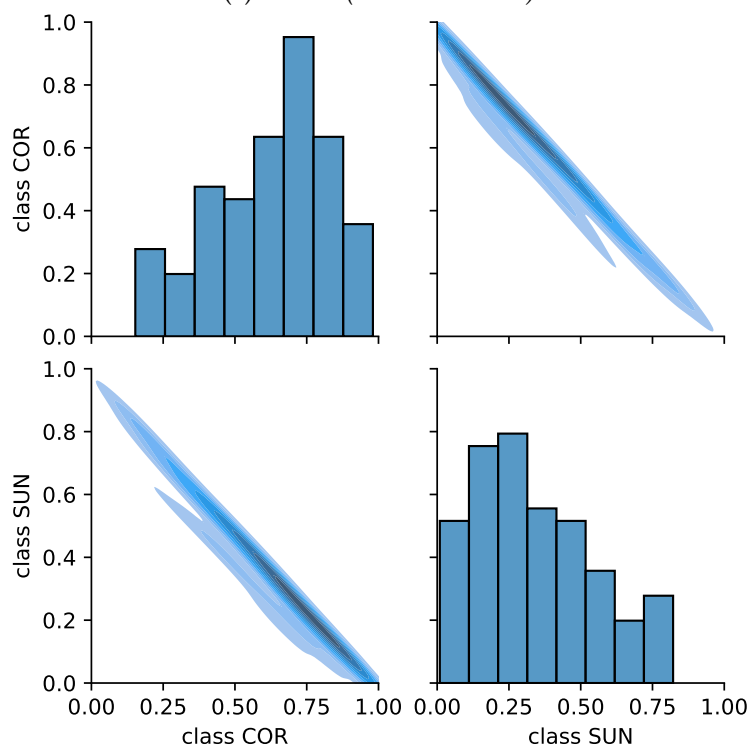
Figures 6.12a and 6.12b represent the approximate posterior predictive distributions obtained with 100 draws. The two largest class membership probabilities are represented for respectively a correct predicted class membership and an incorrect predicted class membership. In the case of a correct predicted class membership (Figure 6.12a), regardless of the draw, the model is very confident: the marginal distributions are tight, thus the variance is low. However, in the case of an incorrect predicted class membership (Figure 6.12b), we observe wide marginal distributions with higher variance. Thus, standard deviation can also be used as a metric in order to compute the classifier uncertainty.

It is also possible to observe this trend by looking at the marginal distributions of the selected class membership for correctly or incorrectly predicted pixels. Figure 6.13 shows that, on average, the posterior predictive distribution of the chosen class membership of correctly predicted pixels has a higher mean but also a lower standard deviation than the posterior predictive distribution of incorrectly predicted pixels.

To conclude, GP model allows to obtain the posterior predictive distribution. However, in practice, in large scale, this information represents a large amount of data. Moreover, producing this information requires more time. Therefore, in the following, only the classification prediction is considered.



(a) correct (true label: SOY)



(b) incorrect (true label: SUN)

Figure 6.12: Posterior predictive distributions, estimated with 100 draws, for the two largest class membership probabilities for a correct (a) and incorrect (b) predicted class membership. For (a), the first class membership is SOY followed by COR. For (b), the first class membership is COR followed by SUN. Marginal distribution of each class is shown on the diagonal and joint distribution between two classes is shown on the off-diagonal.

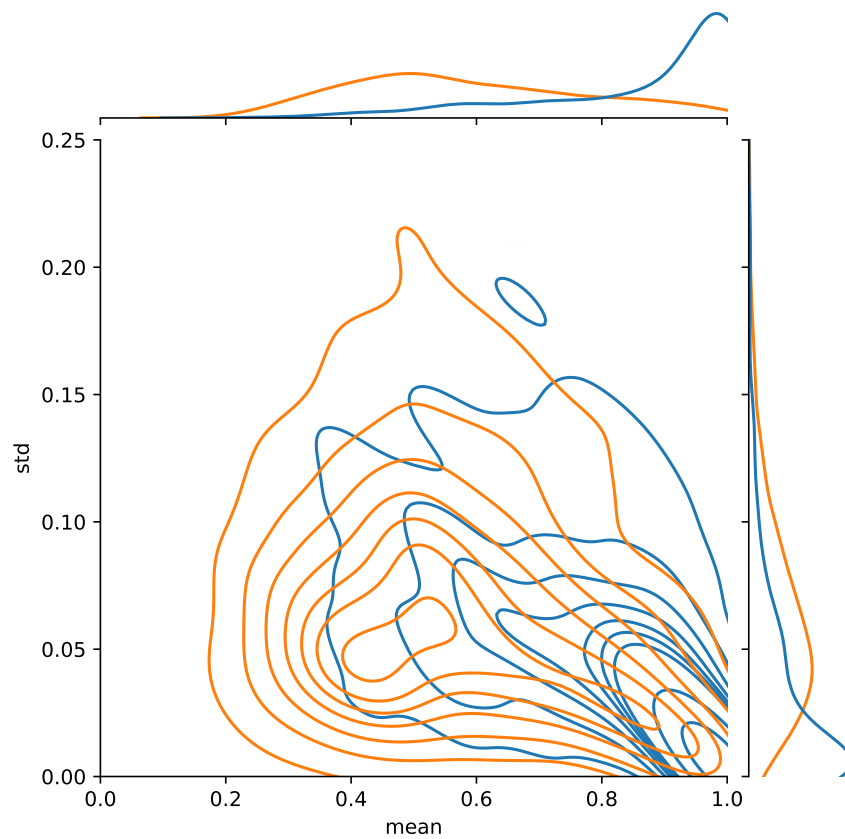


Figure 6.13: Joint density of the standard deviation and the mean of the posterior predictive distribution for the selected class membership (obtained with 10 draws) and their respective marginal densities. — corresponds to 1000 correctly predicted pixels and — corresponds to 1000 incorrectly predicted pixels.

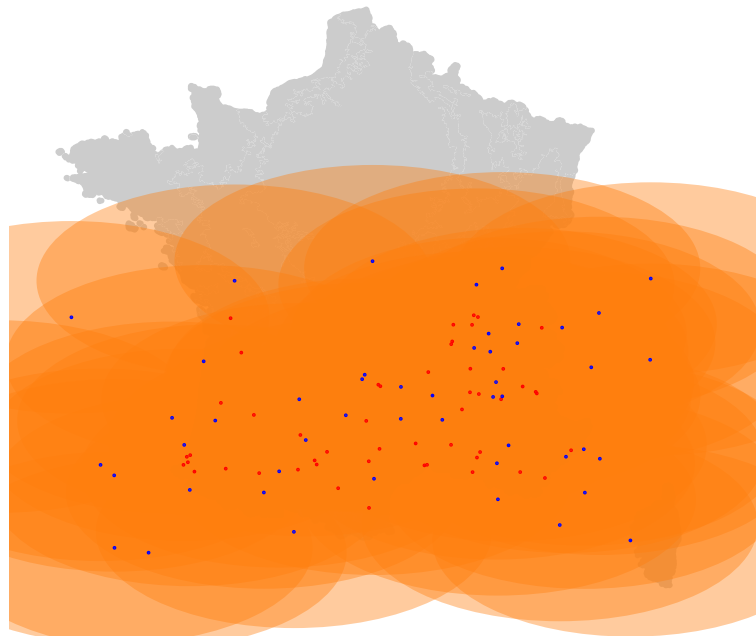
6.4.2. Learned model parameters

Spatial location of inducing points (IP)

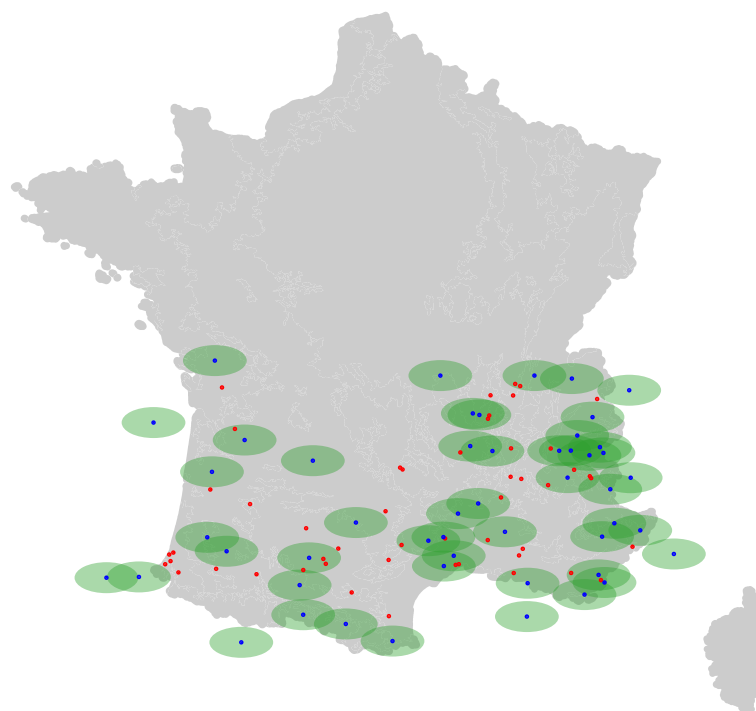
IP are used to approximate the posterior and their values are optimized to find a posterior as similar as possible to the true posterior on the training samples [van der Wilk et al., 2020]. Therefore, relevant information can be obtained by looking at the IP after optimization.

Visualizing the 481 spectro-temporal features is not possible and we restrict here to the two spatial features only, as illustrated in Figure 6.14. The plotted ellipses represent the spatial area inside which the spatial correlation is greater than 0.9. The spatial distribution of the optimized IP can be qualified as regular: the points are more regularly spaced than in the initial random distribution. Also, the obtained spatio-length-scale ℓ_ϕ varies w.r.t. the latent GP, Figure 6.15 represents their distribution.

One possible interpretation is that the model achieves a multi-scale analysis in the spatial domain. Indeed, a latent GP with small spatial length-scale perform a local analysis i.e. its spatial kernel rapidly tends to zero even for spatially close pixels and thus limits its influence locally in the spatial domain. Thus, the correlation is strongly influenced by the spatial distance between two pixels, whatever the spectro-temporal profile (latent GP number 15). The latent GP with large spatial length-scale performs a spatially wider analysis: the spatial kernel is always close to one, even for spatially faraway pixels. In this case, the correlation is very weakly affected by the spatial distance, only the spectro-temporal information is taken into account (latent GP number 12).



(a) latent GP number 12



(b) latent GP number 15

Figure 6.14: Spatial location of inducing points (IP) for 2 different latent GP: \bullet and \bullet represent spatio IP respectively before and after optimization. Orange and green ellipses correspond to the spatial area inside which the spatial correlation is greater than 0.9 respectively for the latent GP number 12 and 15.

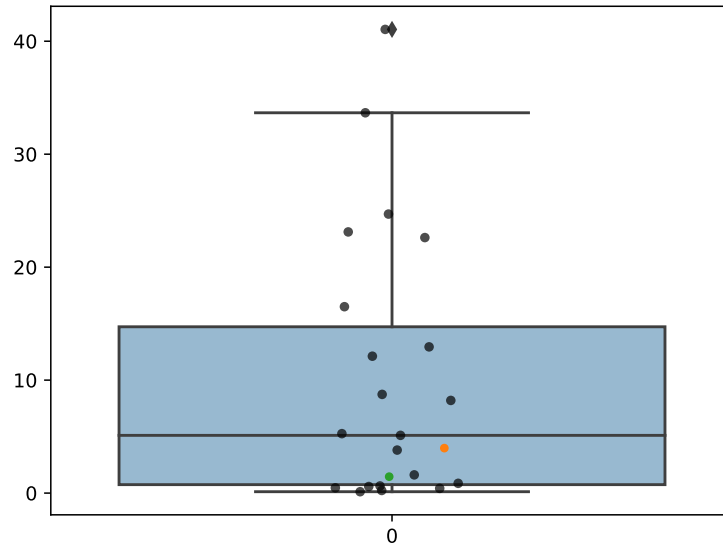


Figure 6.15: Boxplot of the distribution of the length-scale values for each latent GP. The ● and ● represent the length-scale values respectively for the latent GP number 12 and 15 of Figures 6.14a and 6.14b.

Mixing matrix

The coefficients a_{cl} of the mixing matrix \mathbf{A} are used to combine the L independent univariate latent GP g_l to estimate a final GP f_c such as $f_c = \sum_{l=1}^L a_{cl}g_l$. The a_{cl} can be interpreted as the contribution of a latent GP to the class-conditional posterior predictive distribution. Yet, we have found no specific pattern in \mathbf{A} among the different results and we were not able to derive any specific interpretations: all GP contribute significantly.

By accepting an increase in the number of the trainable procedure, some constraints, such as orthogonality ($\mathbf{A}^\top \mathbf{A} = \mathbf{I}_C$), were applied on \mathbf{A} in order to improve the interpretability. However, no improvement in performance was found.

6.5. Feature reduction

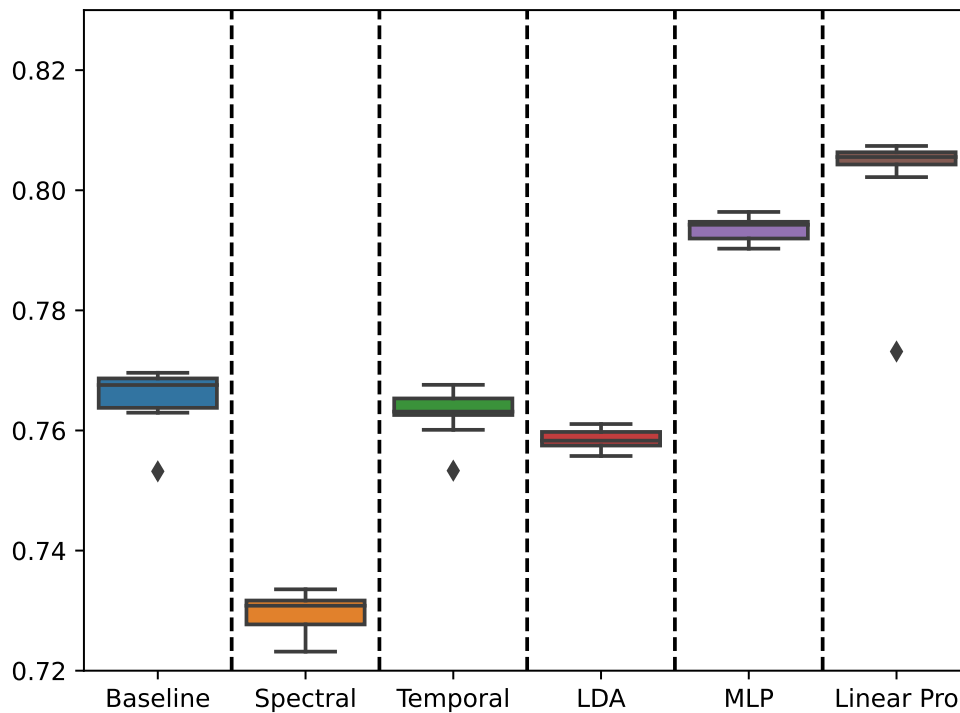
This section constitutes the second part of the Master’s internship that I co-supervised. The objective was to study the influence of feature reduction on model performance, as defined in Section 5.2.5. The λt -GP model is compared to five different methods: Spectral, Temporal, LDA, MLP and Linear Pro. The first three correspond to feature extraction as pre-process. The last two correspond to end-to-end feature extraction.

Figure 6.16 represents the OA computed for the different methods with the data set DS-A in both configurations (*global* and *stratification*). The nomenclature of the methods are presented in Table 5.8. The F-score is presented in Figure B.5 in Appendix B. The OA and the F-score provide similar results.

In Chapter 5, we discussed several approaches for temporal reduction. All the temporal reduction approaches showed results inferior to the baseline method, i.e. without feature extractor. The method with the closest results was with monthly statistics (mean, variance, min, max). However, a total of 624 features were extracted, which does not correspond to a feature reduction. Thus, we choose to show the results for the second method with the closest

results: monthly frequency with only the mean, for a total of 156 features.

In *global* configuration, all the standalone feature extraction methods (i.e. spectral reduction, temporal reduction and LDA) have lower performance results than the baseline method. The temporal reduction is followed by the LDA which is followed by the spectral reduction. In *stratification* configuration, the LDA outperformed temporal and spectral reductions and have similar results to the baseline method. In both configurations, both end-to-end feature extraction methods outperformed the baseline method. Moreover, in both configurations, the Linear Pro end-to-end feature extraction method outperformed the MLP one. Linear Pro is a particular case of the MLP. Thanks to its two distinct linear layers, the spectro-temporal structure of the data can be taken into account. Moreover, Linear Pro has fewer parameters than MLP: almost three times less.



(a) Global configuration

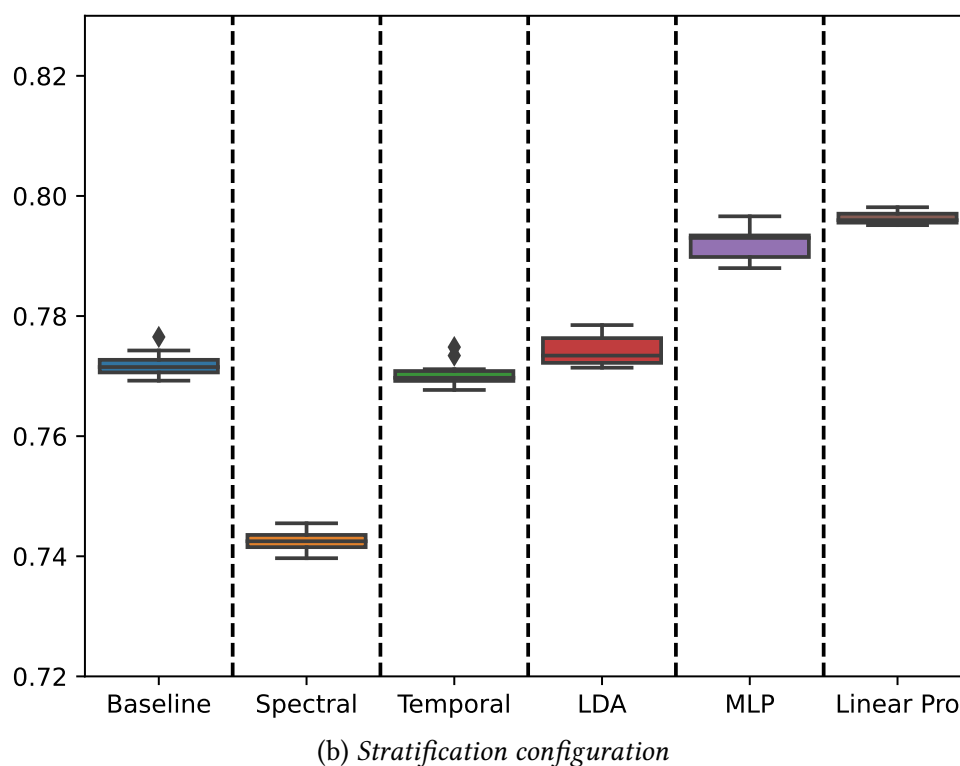


Figure 6.16: Comparison of the OA for the different feature extraction methods computed with the data set DS-A in both configurations: global and stratification. The nomenclature of the methods is described in Table 5.8.

The **SVGP** model presented in the previous part has interesting properties. It was able to handle massive data sets and it outperformed the current **CES OSO** based approach (i.e. **RF** with spatial stratification). Moreover, thanks to a spatial covariance function combined with a spectro-temporal covariance function, the spatial variability was taken into account. Therefore, with the **SVGP** model, the spatial stratification was not needed anymore. Furthermore, by using an end-to-end spectro-temporal feature reduction, the classification performance was improved, as described in Section 6.5. However, it might be suboptimal to perform temporal reduction on linearly interpolated time series. Indeed, it might be more interesting to make the reduction during interpolation. Therefore, in the following part, we propose to use directly the irregular and unaligned time series by using a time and space informed kernel interpolator. The spectro-temporal reduced latent representation produced by the interpolator will be optimized for the classification task during the training process.

Part III.

**Attention-based interpolation with
Gaussian Processes for land cover
classification using irregular and
unaligned SITS at large scale**

7.1. Standard temporal resampling methods	201
7.1.1. Imputation methods	201
7.1.2. Filtering methods	203
7.1.3. Kernel-based methods	207
7.2. Transformer methods for temporal resampling	210
7.2.1. Main concepts of the attention mechanisms	210
7.2.2. Attention-based interpolation	211
7.3. Multi Time Attention Networks (mTAN)	212

In the following, the main idea is to take into account the missing values from irregular and unaligned time series. In Chapter 2, notations were introduced in the case of regular time series of fixed size (i.e. all the pixels are acquired on the same dates). In this chapter and the following chapters (Chapters 8 and 9), we propose to introduce the notations for irregular and unaligned time series (i.e. pixels are acquired on different dates).

The i th pixel time series at time t_k is still defined as: $\mathbf{x}^i(t_k)$ with $i \in \{1, \dots, N\}$ and N the number of pixels and its spectral measurements still correspond to: $\{x_1^i(t_k), \dots, x_j^i(t_k), \dots, x_D^i(t_k)\}$ with D the number of spectral features. However, not all spectral measurements may be observed i.e. a spectral feature j is observed at T_j^i timestamps: $\mathbf{T}_j^i = \{t_{j1}^i, \dots, t_{jk}^i, \dots, t_{jT_j^i}^i\}$, where T_j^i is the number of valid observations (e.g., no clouds or no cloud shadows). Because of satellite swaths and weather, time series are unaligned, i.e., $\mathbf{T}_j^i \neq \mathbf{T}_j^{i'}$. For simplicity, in this work, we assume that all spectral features are available for each timestamp, i.e., $\mathbf{T}_j^i = \mathbf{T}_j^{i'} = \mathbf{T}^i$. This is commonly the case when working with only one sensor, but the proposed method can be extended to multi-source data straightforwardly. We define the set of all timestamps \mathbf{T} such as:

$$\begin{aligned} \mathbf{T} &= \bigcup_{i=1}^N \mathbf{T}^i \\ &= \{t_1, \dots, t_k, \dots, t_T\} \end{aligned}$$

with T the total number of observations. For each pixel, we define a mask time series $\mathbf{m}^i \in \{0, 1\}^T$ such as

$$m^i(t_k) = \begin{cases} 1 & \text{if } t_k \in \mathbf{T}^i \\ 0 & \text{otherwise} \end{cases} \quad \forall t_k \in \mathbf{T}, \quad (7.1)$$

which indicates whether the pixel i at time t_k is observed or not. We further define an *augmented* pixel time series \mathbf{x}_j^{i*} as the pixel

$$x_j^{i*}(t_k) = \begin{cases} x_j^i(t_k) & \text{if } m^i(t_k) = 1 \\ 0 & \text{otherwise} \end{cases} \quad \forall t_k \in \mathbf{T}, \quad (7.2)$$

For clarity, in the following, we consider only one pixel time series and we drop the index i in the remaining of the chapter.

As discussed in Chapter 2, most conventional classifiers work only with regular time series. As a consequence, a required preprocessing step is to resample irregular and unaligned time series onto a regular temporal grid of R dates: $\mathbf{R} = \{r_1, \dots, r_l, \dots, r_R\}$. In the following, a review on the main techniques used for temporal resampling is proposed.

7.1. Standard temporal resampling methods

Resampling methods can be sorted in three different categories [Poggio et al., 2012]:

1. fusion methods: informations from different sensors are used,
2. spatial methods: informations from neighboring pixels in a spatial window are used,
3. temporal methods: informations from the same pixel in a temporal window are used.

In the following, we will focus only on the last technique corresponding to temporal resampling methods and we will present a very brief review. Shen *et al.* [Shen et al., 2015] provided a more extended review on these temporal approaches including temporal replacement methods, temporal filter methods and temporal learning model methods.

Figure 7.1 illustrates an irregularly sampled NDVI time series of a pixel labeled as COR (see Table 3.3 for the description of this class). Some observations are identified as valid observations and some of them correspond to observations flagged as clouds or cloud shadows in the level 2A (L2A) masks. In the following, we will show the results of different temporal resampling methods applied to map this irregular time series onto a regular temporal grid of $R = 365$ dates with an interval of one day.

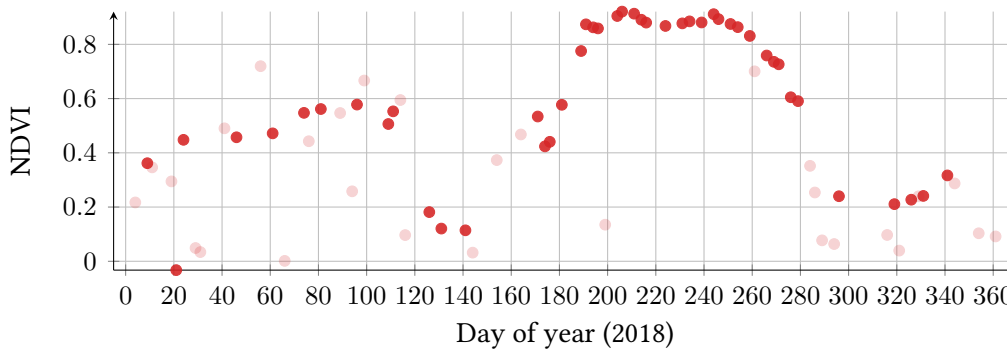


Figure 7.1: NDVI time series for a pixel labeled as COR. Filled red dots correspond to valid observations, transparent red dots correspond to observations flagged as clouds or cloud shadows in the level 2A masks.

7.1.1. Imputation methods

In imputation techniques, the missing value can be estimated, for instance, by:

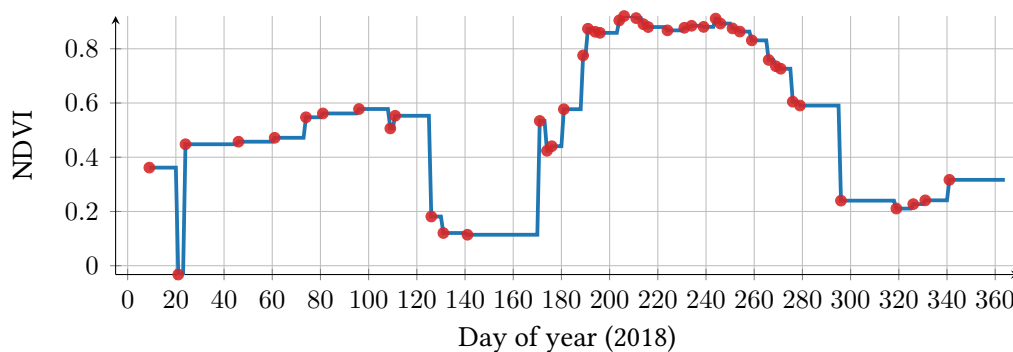
- a value from another date (direct replacement) or
- the mean value of the time series (mean imputation).

In direct replacement, the missing value can be replaced by the previous, the next or the nearest cloud-free value. Figures 7.2a, 7.2b and 7.2c represent the direct replacement with the previous, the next and the nearest cloud-free value, respectively. With previous direct replacement or next direct replacement, we suppose that no changes can occur over the temporal period considered [Rulloni et al., 2012]. However, even if the time interval between two dates is very

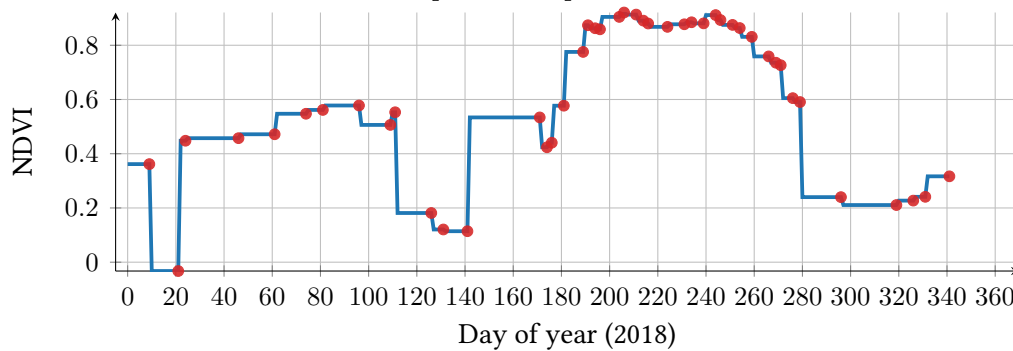
short, there is still temporal difference and bias can be introduced. Therefore, it is not correct to ignore the temporal difference between two dates, in particular with Sentinel-2 time series. However, with the nearest direct replacement, this temporal difference is taken into account.

In mean imputation, the missing value can be replaced by the mean value of all the cloud-free values. In the case of the NDVI time series of the COR class, the reconstructed profile has no physical meaning, as illustrated in Figure 7.2d. Another strategy is to replace the missing value of a specific date by the mean value of some selected pixels for this date. It implies that the selected pixels should all have the same class. Mouret *et al.* [Mouret *et al.*, 2022] computed the mean imputation with the latter strategy on Sentinel-2 time series for rapeseed parcels. They showed that if the values to be imputed are very unusual, the mean imputation gives very poor results. In general, mean imputation leads to an underestimation of the variance.

Both techniques presented, direct replacement and mean imputation, are quite easy to implement at large scale. However, they introduce huge bias. In practice, these techniques are not suitable for temporal time series such as Sentinel-2.



(a) Direct replacement: previous value



(b) Direct replacement: next value

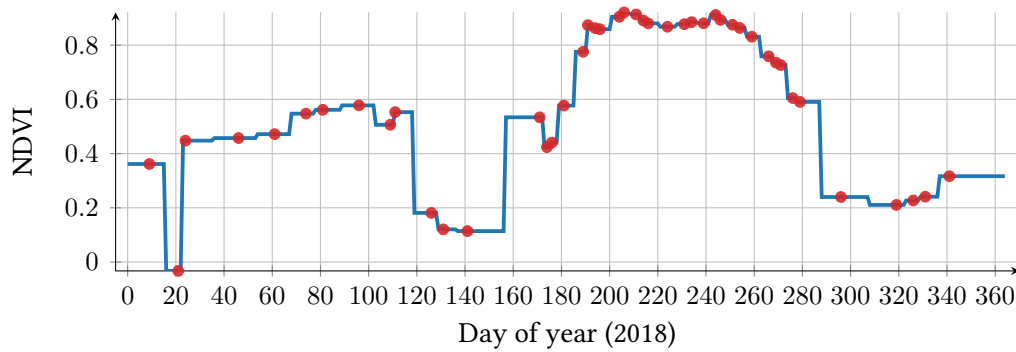
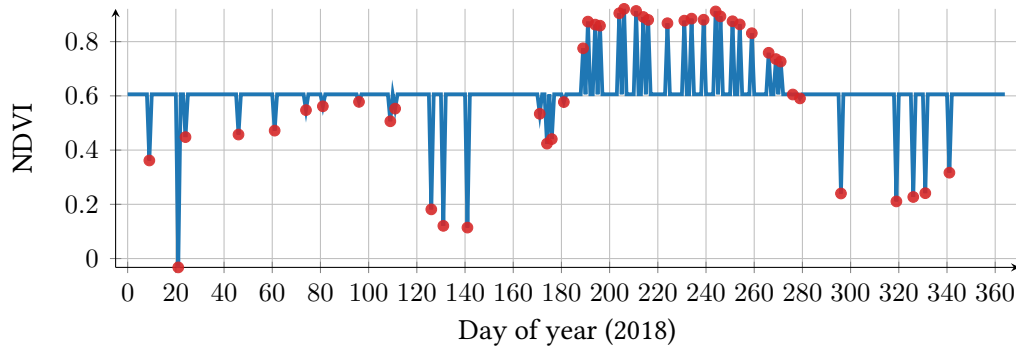
(c) *Direct replacement: nearest value*(d) *Mean imputation*

Figure 7.2: Imputation methods for the COR NDVI time series represented in Figure 7.1. Four different methods are studied: previous direct replacement, next direct replacement, nearest direct replacement, mean imputation. The blue curve corresponds to the imputed values every one day. The red dots represent the valid observations.

7.1.2. Filtering methods

Filtering methods were originally used to reduce the noise in time series but they can also be used to interpolate missing values. They are mainly divided into global and local methods. In global methods, the characteristics of the full time series are considered. Whereas, in local methods, the characteristics of only a portion of the time series, through a local temporal window, are considered.

Global methods

In global methods, a polynomial function can be used to parametrize the time series. No prior knowledge of the temporal behavior is required. Depending on the number of valid dates, a polynomial with more or less degrees is chosen. If the degree is too high, the interpolation between two distant dates can lead to extreme fluctuations. Furthermore, if the degree is too low, the interpolation is too smooth and does not intersect the valid observations. Therefore, a compromise should be made between the degree of the polynomial function and the number of valid dates.

If there is an a priori on the data and more precisely if the time series considered are NDVI time series of vegetation classes, different techniques can be used such as asymmet-

ric Gaussian function [Jonsson and Eklundh, 2002] or double logistic functions [Beck et al., 2006]. These methods work quite well but they can show some difficulties to detect short-term changes in vegetation classes [Zhu et al., 2012]. Besides, these two methods require knowing the class, which you do not know before ... doing the classification!

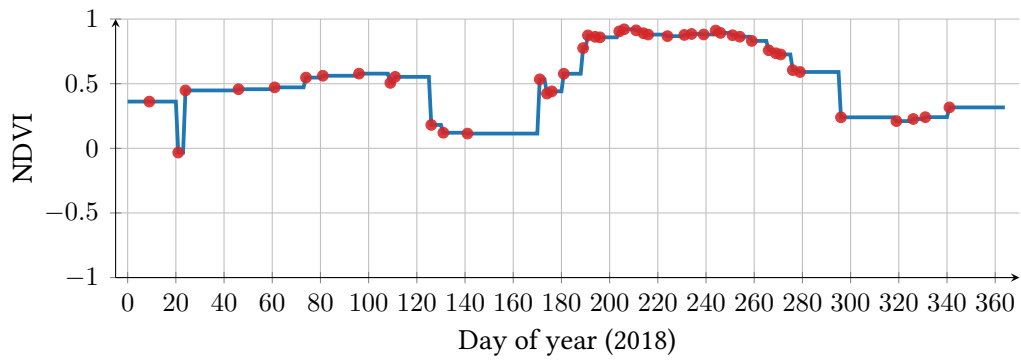
Besides, signal processing methods can also be used as temporal interpolation methods such as the **Harmonic ANalysis of Time Series (HANTS)** method [Yang et al., 2015a], [Zhou et al., 2015], the Fourier analysis [G. J. Roerink and Verhoef, 2000] or wavelet methods [Lu et al., 2007]. Fourier analysis and wavelet methods require regular time series. They are usually used to filter multi-annual regular time series [Scharlemann et al., 2008], [De Oliveira et al., 2009].

Local methods

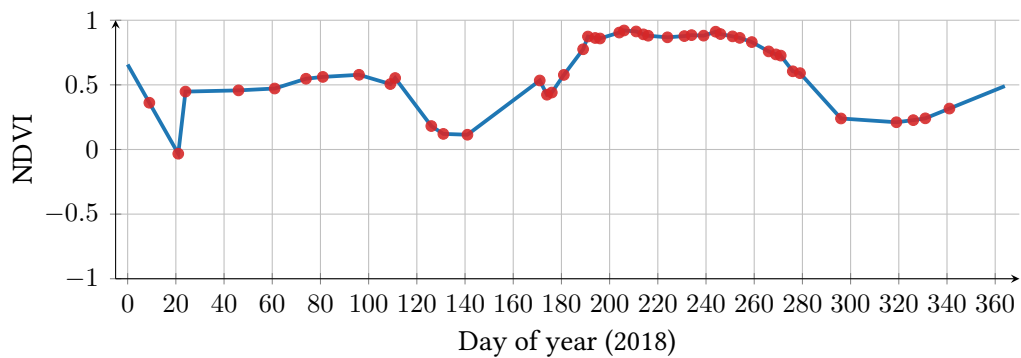
Instead of applying methods on the complete time series, local methods work on small parts of the time series corresponding to local temporal windows. In local basis function expansion methods, a different function is assigned to each window. In sliding window filtering methods, the time series are processed with a local moving window.

Concerning local basis function expansion methods, spline interpolation allows to fit low-degree polynomial functions in multiple parts instead of a high degree polynomial function in the complete time series (i.e. global method). Figure 7.3 represents the spline interpolation of zeroth, first (i.e. linear), second (i.e. quadratic) and third (i.e. cubic) degree for the COR **NDVI** time series of Figure 7.1. The smoother results are obtained with the linear spline interpolation, as illustrated in Figure 7.3b. With the zeroth degree spline interpolation, Figure 7.3a, the results obtained are very similar to the ones obtained with the direct replacement. Besides, with higher degrees, i.e. quadratic and cubic, the interpolation is continuous everywhere but less smooth for temporal domain with no seen samples, as illustrated in Figures 7.3c and 7.3d. Hence, the interpolation between two distant dates is very unstable.

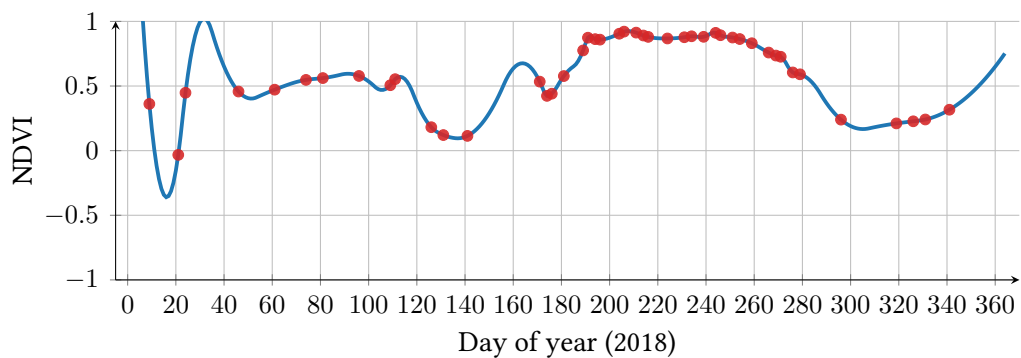
For sliding window filter methods, different techniques are defined in the literature for **NDVI** time series such as the Savitzky-Golay filter [Chen et al., 2004], the **Best Index Slope Extraction (BISE)** [Viovy et al., 1992], the **Iterative interpolation for Data Reconstruction (IDR)** [Julien and Sobrino, 2010] or the adapted local regression filter [Moreno et al., 2014]. Focusing on the Savitzky-Golay filter, two parameters need to be defined: the order of the polynomial function and the size of the window. Figure 7.4 represents the Savitzky-Golay filter method applied to the COR **NDVI** time series of Figure 7.1 for different sizes of window (3, 7 and 13 days). The function used `scipy.signal.savgol_filter` is implemented for filtering, not for interpolation. Therefore, with this function, an interpolation with one day interval can not be performed. The wider the window, the smoother the curve. More generally, the performance of the sliding window filter methods is highly affected by the selection of the filtering parameters.



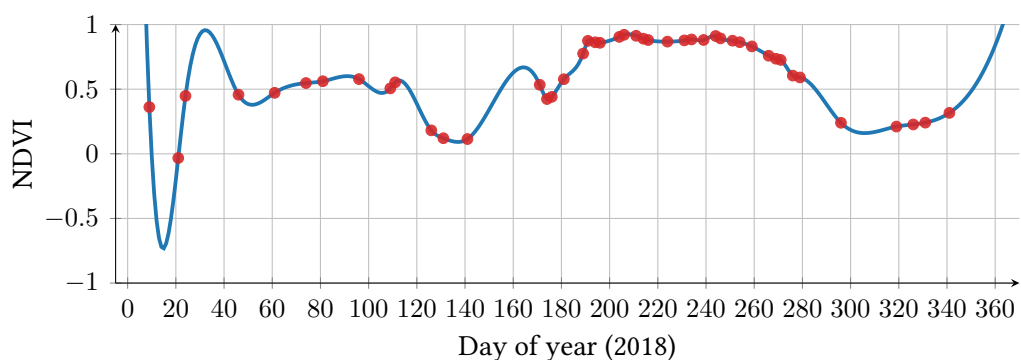
(a) Zeroth degree



(b) First degree / Linear

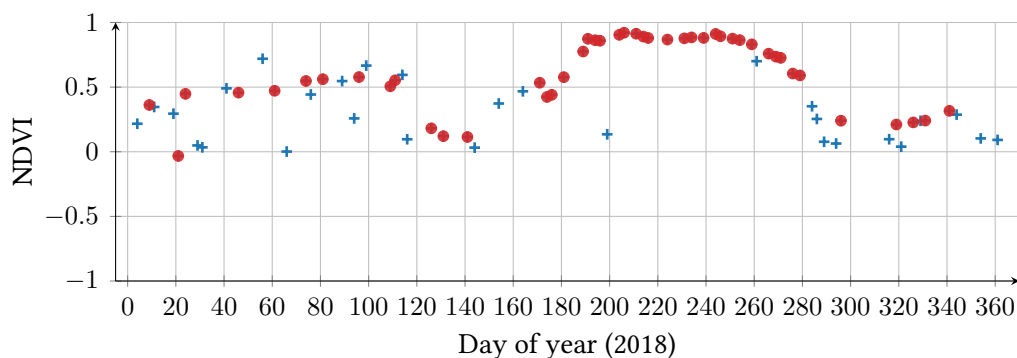


(c) Second degree / Quadratic

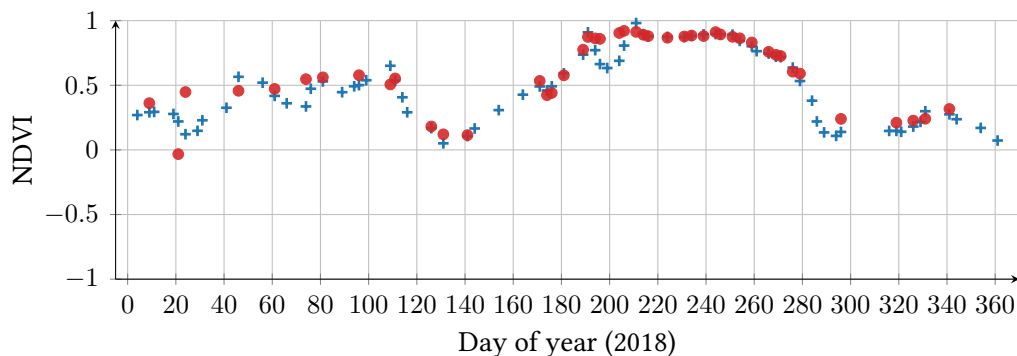


(d) Third degree / Cubic

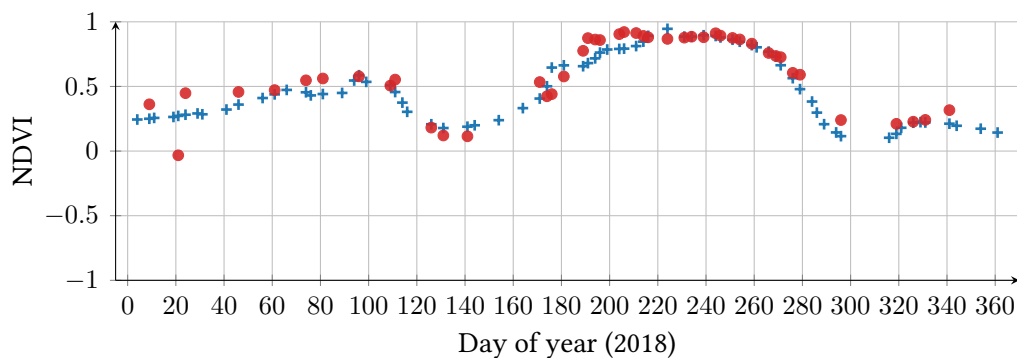
Figure 7.3: Spline interpolation methods for the COR NDVI time series represented in Figure 7.1. The blue curve corresponds to the interpolated values every one day. The red dots represent the valid observations. The Python function used is: `scipy.interpolate.splrep`.



(a) Window size: 3, polynomial degree: 2



(b) Window size: 7, polynomial degree: 2



(c) Window size: 13, polynomial degree: 2

Figure 7.4: Savitzky-Golay filter methods for the COR NDVI time series represented in Figure 7.1. The blue crosses correspond to the filtered values by using only the cloud and cloud-free observations. The red dots represent the valid observations. The Python function used is: `scipy.signal.savgol_filter`.

7.1.3. Kernel-based methods

Kernel-based methods are non parametric local filtering methods. They compute a weighted sum of neighboring points based on a kernel function in order to estimate missing values.

In this work, we focus on the well-established Nadaraya-Watson kernel smoother [Hastie et al., 2001, Chapter 6]. In the case where all the observations t_k are available, the interpolated \hat{x}_j at latent timestamp¹ r_l , for a given pixel time series \mathbf{x}_j (with $j \in \{1, \dots, D\}$), using the Nadaraya-Watson kernel smoother, is given by:

$$\hat{x}_j(r_l) = \frac{\sum_{t_k=t_1}^{t_T} K_\lambda(r_l, t_k) x_j(t_k)}{\sum_{t'_k=t_1}^{t_T} K_\lambda(r_l, t'_k)} \quad (7.3)$$

with K_λ some similarity kernel [Hastie et al., 2001, Chapter 6]. For the situation where not all observations t_k are available, Equation (7.3) can be rewritten using the augmented pixel time series \mathbf{x}_j^* defined in Equation (7.2):

$$\hat{x}_j(r_l) = \frac{\sum_{t_k=t_1}^{t_T} K_\lambda(r_l, t_k) m(t_k) x_j^*(t_k)}{\sum_{t'_k=t_1}^{t_T} K_\lambda(r_l, t'_k) m(t'_k)} \quad (7.4)$$

with $m(t_k)$ the masked value at time t_k . The isotropic kernel can be written as:

$$K_\lambda(r_l, t_k) = D\left(\frac{|t_k - r_l|}{\lambda}\right) = D(\Delta t) \quad (7.5)$$

with $D(\Delta t)$ a positive real valued function. The value of $D(\Delta t)$ is decreasing for increasing distance between r_l and t_k . From Equation (7.3), $\hat{x}_j(r_l)$ is a convex combination of original pixel values, whose weights are computed using the kernel applied on the temporal domain: a weight is assigned to $x_j^*(t_k)$ based on its distance from r_l , scaled by the bandwidth λ . This parameter needs to be determined from the training data set and can be found by cross-validation [Li and Racine, 2023].

Different functions are commonly found in literature for $D(\Delta t)$:

- Uniform (or rectangular):

$$D(\Delta t) = \begin{cases} \frac{1}{2}, & \text{if } |\Delta t| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (7.6)$$

- Triangular:

$$D(\Delta t) = \begin{cases} (1 - |\Delta t|), & \text{if } |\Delta t| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (7.7)$$

- Epanechnikov (or parabolic):

$$D(\Delta t) = \begin{cases} \frac{3}{4}(1 - \Delta t^2), & \text{if } |\Delta t| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (7.8)$$

¹Unobserved date on which the value is estimated.

- Tricube:

$$D(\Delta t) = \begin{cases} \frac{70}{81}(1 - |\Delta t|^3)^3, & \text{if } |\Delta t| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (7.9)$$

- Gaussian (or RBF):

$$D(\Delta t) = \exp(-\Delta t^2). \quad (7.10)$$

Figure 7.5 illustrates the previous standard functions $D(\Delta t)$ defined in Equations (7.6), (7.7), (7.8), (7.9) and (7.10). In Equations (7.8) and (7.9), the factor $\frac{3}{4}$ and $\frac{70}{81}$, respectively, is a normalization constant ensuring that the total area under the Epanechnikov and tricube kernel curve is equal to 1 between $[-1, 1]$, as illustrated in Figure 7.5.

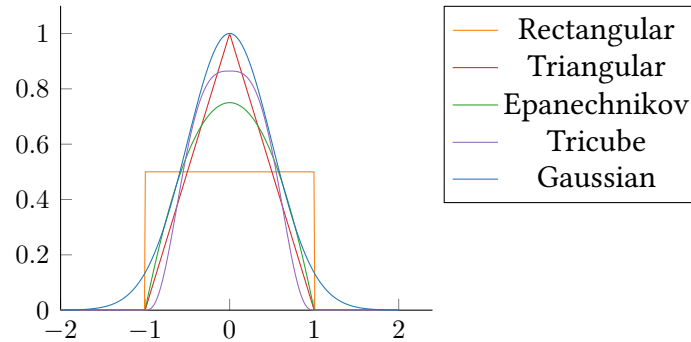
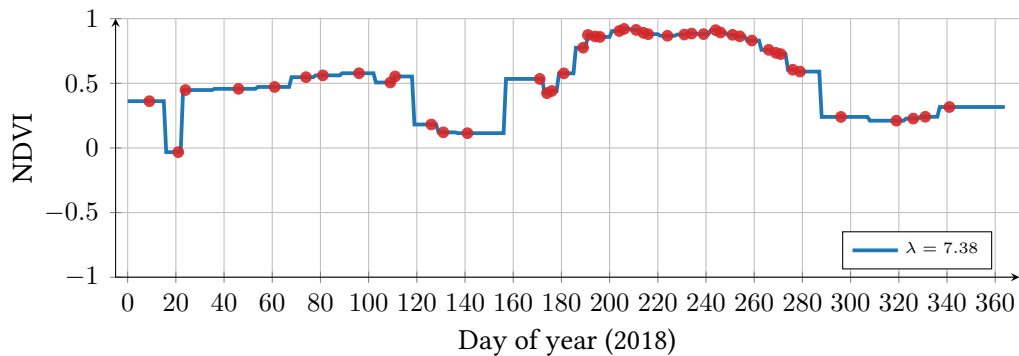


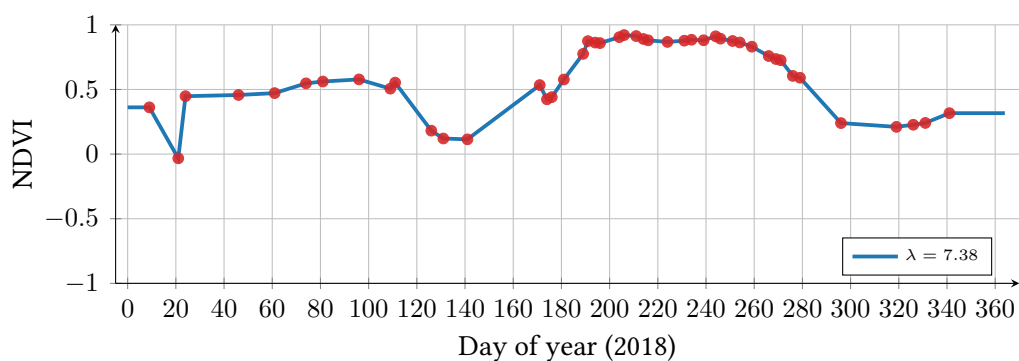
Figure 7.5: Representation of the previous standard functions $D(\Delta t)$ defined in Equations (7.6), (7.7), (7.8), (7.9) and (7.10). For all functions, we have $r_l = 0$, $t_k \in [-2, 2]$ and $\lambda = 1$.

Figure 7.6 represents the kernel-based interpolation of the COR NDVI time series obtained with three different functions: uniform, triangular and Gaussian. The uniform kernel-based interpolation gives results very similar to the zeroth degree spline interpolation, as illustrated in Figure 7.6a. All points in the neighborhood have equal weight. In triangular kernel-based interpolation, the weights linearly die off with the distance. This kernel-based interpolation is very similar to the linear interpolation (with by extension was also called temporal gap-filling [Inglada et al., 2015, Inglada et al., 2017]), as illustrated in Figure 7.6b. Figure 7.6c represent the Gaussian kernel-based interpolation with different values for the bandwidth $\lambda \in \{5, 7.38, 10\}$. $\lambda = 7.38$ correspond to the averaged distance between valid dates in the COR NDVI time series. With a small λ , the interpolation between two distant observations are very smooth. Indeed, the interpolation is smoother with $\lambda = 5$ between the day 140 and the day 170 than with $\lambda = 10$. Moreover, with a large λ , the interpolation between two close observations are very smooth. Indeed, the interpolation is smoother with $\lambda = 10$ between the day 80 to the day 100 than with $\lambda = 5$. Therefore, a compromise for λ needs to be made in order to minimize the reconstruction error.

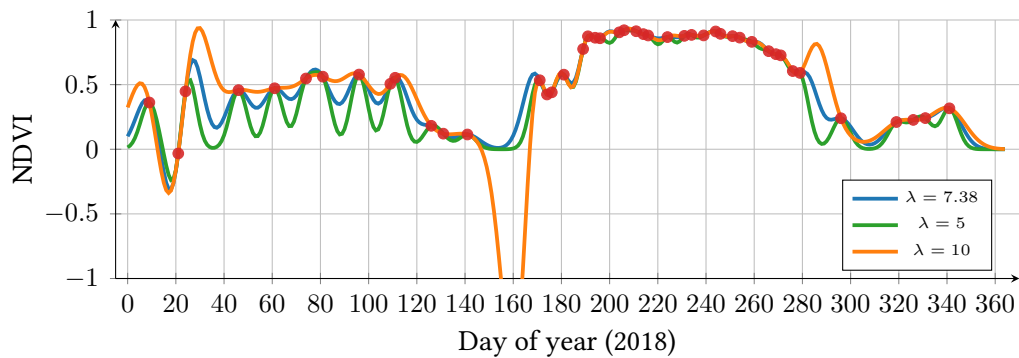
With the kernel-based methods, the bandwidth λ is constant over the temporal domain. It corresponds to an homoscedastic noise assumption: the noise level does not change over time. However, in SITS the noise may vary along the seasons or because of cloud cover. Therefore, the noise can be characterized as heteroscedastic. By proposing an heteroscedastic modeling instead of an homoscedastic one, results can be improved [Ferraty et al., 2019]. In the following, we propose a strategy which can take into account this heteroscedastic noise through the use of the attention mechanism.



(a) Uniform



(b) Triangular



(c) Gaussian

Figure 7.6: Kernel-based interpolation methods for the COR NDVI time series represented in Figure 7.1. The curves correspond to the interpolated values every one day. The red dots represent the valid observations. $\lambda = 7.38$ correspond to the averaged distance between dates in the COR NDVI time series. Different bandwidth values $\lambda \in \{5, 7.38, 10\}$ were used for the Gaussian kernel. The Python function used for (a) is: `scipy.interpolate.interp1d` with `kind='nearest'`. The Python function used for (b) and (c) is: `scipy.interpolate.Rbf` with `function='linear'` and `function='gaussian'`, respectively.

7.2. Transformer methods for temporal resampling

The transformer architecture is widely used in machine learning. As described in Section 2.2.2, transformers can handle irregular time series. The transformer architecture generalized the use of the attention mechanism. In the following, the main concepts of the attention mechanisms are introduced. Then, a link between the attention mechanism and the kernel-based methods is proposed.

7.2.1. Main concepts of the attention mechanisms

In this section, we start by introducing the attention mechanisms in a standard machine translation framework. We defined the following components:

- a key \mathbf{k}_i of size d_k which represents a word or a sequence of words in a specific language (e.g. "Envie d'aller courir ?"),
- a value \mathbf{v}_i of size d_v which corresponds to the translation of \mathbf{k}_i in another language (e.g. "Fancy a run ?"),
- $\mathcal{D} = \{(\mathbf{k}_1, \mathbf{v}_1), \dots, (\mathbf{k}_N, \mathbf{v}_N)\}$ which represents the collection of the N keys and values, and,
- a query \mathbf{q} of size d_q which represents a new word or a new sequence of words to translate.

The attention mechanism firstly introduced by Bahdanau *et al.* [Bahdanau et al., 2014] can be written such as:

$$\text{Attention}(\mathbf{q}, \mathcal{D}) = \sum_{i=1}^N \alpha(\mathbf{q}, \mathbf{k}_i) \mathbf{v}_i, \quad (7.11)$$

with $\alpha(\mathbf{q}, \mathbf{k}_i)$ the scalar attention weights. The attention mechanism can be considered as a mapping between the query \mathbf{q} and the collection of N keys and values to an output.

A common strategy used in attention mechanisms to ensure that the scalar attention weights sum up to 1 and are also nonnegative is to use the softmax function such as:

$$\alpha(\mathbf{q}, \mathbf{k}_i) = \text{softmax}(a(\mathbf{q}, \mathbf{k}_i)) = \frac{\exp(a(\mathbf{q}, \mathbf{k}_i))}{\sum_{j=1}^N \exp(a(\mathbf{q}, \mathbf{k}_j))}, \quad (7.12)$$

with $a(\mathbf{q}, \mathbf{k}_i)$ an attention function.

An attention function that is often used in Transformers [Vaswani et al., 2017] is the scaled-dot product attention scoring function:

$$a(\mathbf{q}, \mathbf{k}_i) = \frac{\mathbf{q}^\top \mathbf{k}_i}{\sqrt{d_k}} \quad (7.13)$$

with $d_k = d_q$. If \mathbf{k}_i and \mathbf{q} have different sizes (i.e. $d_k \neq d_q$), a matrix \mathbf{M} can be used to project the spaces: $\mathbf{q}^\top \mathbf{M} \mathbf{k}_i$. The weights of this matrix can be learned during the training. In general, the attention function a can take any form and can be learned from the data. For example, this function can be parametrized by a neural network.

7.2.2. Attention-based interpolation

In the previous section, the attention mechanism was introduced for machine translation. However, in our case, we are interested in temporal interpolation. In the case where all the observations t_k are available, the interpolated \hat{x}_j at latent timestamp r_l , for a given pixel time series \mathbf{x}_j (with $j \in \{1, \dots, D\}$), by using the attention mechanism, defined in Equation (7.11), can be written as:

$$\hat{x}_j(r_l) = \text{Attention}(r_l, \mathcal{D}') \quad (7.14)$$

$$= \sum_{t_k=t_1}^{t_T} \alpha(r_l, t_k) x_j(t_k) \quad (7.15)$$

$$= \sum_{t_k=t_1}^{t_T} \text{softmax}(a(r_l, t_k)) x_j(t_k) \quad (7.16)$$

$$= \frac{\sum_{t_k=t_1}^{t_T} \exp(a(r_l, t_k)) x_j(t_k)}{\sum_{t'_k=t_1}^{t_T} \exp(a(r_l, t'_k))} \quad (7.17)$$

with $\mathcal{D}' = \{(t_1, x_j(t_1)), \dots, (t_T, x_j(t_T))\}$.

By taking $\exp(a(r_l, t_k)) = K_\lambda(r_l, t_k) = \exp\left(-\frac{|t_k - r_l|^2}{\lambda^2}\right)$ in Equation (7.17), we recover the Gaussian kernel smoother defined in Equations (7.3) and (7.10). Therefore, with the attention-based interpolator, the distance and the bandwidth are now trained by attention mechanisms. The bandwidth λ is not fixed and can now take into account the heteroscedastic noise. Indeed, the bandwidth can adapt itself to distances in dates.

Equation (7.17) can be rewritten in the case where all the observations t_k are not available using the augmented pixel time series \mathbf{x}_j^* :

$$\hat{x}_j(r_l) = \sum_{t_k=t_1}^{t_T} \alpha(r_l, t_k) x_j^*(t_k) \quad (7.18)$$

$$= \frac{\sum_{t_k=t_1}^{t_T} \exp(a(r_l, t_k)) m(t_k) x_j^*(t_k)}{\sum_{t'_k=t_1}^{t_T} \exp(a(r_l, t'_k)) m(t'_k)} \quad (7.19)$$

$$= \sum_{t_k=t_1}^{t_T} \text{mSoftmax}(a(r_l, t_k), m(t_k)) x_j^*(t_k) \quad (7.20)$$

with

$$\text{mSoftmax}(a(r_l, t_k), m(t_k)) = \frac{\exp(a(r_l, t_k)) m(t_k)}{\sum_{t'_k=t_1}^{t_T} \exp(a(r_l, t'_k)) m(t'_k)} \quad (\text{mSoftmax for masked softmax}). \quad (7.21)$$

In the following, the implementation of an attention-based interpolator, with a specific choice for $a(r_l, t_k)$, called **multi Time Attention Networks (mTAN)**, is presented.

7.3. Multi Time Attention Networks (mTAN)

Shukla *et al.* [Shukla and Marlin, 2021] proposed to extend the attention-based interpolator in an end-to-end framework: the **multi Time Attention Networks (mTAN)**. Instead of using directly r_l and t_k in the scaled-dot product attention (c.f. Equation (7.13)), the main idea is to use a representation of them. Therefore, they proposed to use a learnable time embedding function ϕ (named *temporal positional encoding*). It maps a given timestamp t onto a higher dimensional space of size E such as:

$$\phi : \mathbb{R} \rightarrow \mathbb{R}^E$$

$$t \mapsto \phi(t) = \begin{bmatrix} \omega_1 t + \alpha_1 \\ \sin(\omega_2 t + \alpha_2) \\ \vdots \\ \sin(\omega_E t + \alpha_E) \end{bmatrix} \quad (7.22)$$

with ω_p and α_p (with $p \in \{1, \dots, E\}$), the learnable parameters. From the attention point of view, $a(r_l, t_k)$ can be written as:

$$a(r_l, t_k) = \frac{\phi(r_l)^\top \mathbf{W}_q^\top \mathbf{W}_k \phi(t_k)}{\sqrt{E}}$$

with \mathbf{W}_q and \mathbf{W}_k two learnable matrices of size $E \times E$. The indices q and k refer to *query* and *key* terms in the attention mechanism framework defined in Section 7.2.1. From the kernel point of view, the similarity kernel is written as:

$$K_\lambda(r_l, t_k) = \exp\left(\frac{\phi(r_l)^\top \mathbf{W}_q^\top \mathbf{W}_k \phi(t_k)}{\sqrt{E}}\right).$$

The attention-based kernel smoother can be written as:

$$\begin{aligned} \hat{x}_j(r_l) &= \sum_{t_k=t_1}^{t_T} \text{mSoftmax}\left(\frac{(\phi(t_k)^\top \mathbf{W}_k^\top \mathbf{W}_q \phi(r_l))}{\sqrt{E}}, m(t_k)\right) x_j^*(t_k) \\ &= \text{mSoftmax}\left(\frac{(\Phi(\mathbf{T})^\top \mathbf{W}_k^\top \mathbf{W}_q \phi(r_l))}{\sqrt{E}}, \mathbf{m}\right)^\top \mathbf{x}_j^* \\ &= \gamma_{r_l}^\top \mathbf{x}_j^*. \end{aligned} \quad (7.23)$$

with $\Phi(\mathbf{T}) = [\phi(t_1), \dots, \phi(t_T)]$, the matrix of embeddings of \mathbf{T} of size $E \times T$ and $\mathbf{m} = [m(t_1), \dots, m(t_T)]^\top$, the vector of the mask values of size T .

The authors of [Shukla and Marlin, 2021] further propose to use multi-head attention, i.e., H matrices of embeddings with $\Phi_H(\mathbf{T}) = \{\Phi_h(\mathbf{T})\}_{h=1}^H$, and also H time embedding functions with $\phi_H(r_l) = [\phi_1(r_l), \dots, \phi_H(r_l)]$. A learnable linear layer β_H of size H is used to produce the interpolated value

$$\hat{x}_j(r_l) = \beta_H^\top \Gamma_{r_l}^H \mathbf{x}_j^*. \quad (7.24)$$

with $\Gamma_{r_l}^H = [\gamma_{r_l}^1, \dots, \gamma_{r_l}^H]$ of size $T \times H$. This equation can be computed for every spectral feature j and every latent date r_l .

The **mTAN**, as defined in Equation (7.24), has extended interpolation flexibility w.r.t. the conventional kernel smoother. The **mTAN** is a kernel-based interpolator, whose kernels are adaptive and can be optimized from the data with a loss function for a specific task. Also, it is worth noting that Equation (7.24) benefits from the computational efficiency of attention mechanism (parallel computation) and all parameters are learnable during the training step.

In [Shukla and Marlin, 2021], the **mTAN** was used as input and output layers in a encoder-decoder architecture and a classifier was jointly learned using features from the latent-space. In the next chapter, we propose an extension of the **mTAN** (called **EmTAN**) in order to use the spatial information and to reduce the spectral dimension of the **SITS**.

CHAPTER 8

EmTAN-SVGP CLASSIFICATION: METHOD AND EXPERIMENTAL SET-UP

8.1. Spatially informed interpolator for GP classification	216
8.1.1. Spectro-temporal feature reduction	218
8.1.2. Spatial positional encoding	218
8.1.3. Trainable parameters	220
8.2. Experimental set-up	221
8.2.1. Data set generation	221
8.2.2. Methods set-up	221
8.2.3. Map production	224

In this chapter, we propose to combine a temporal interpolator with the **SVGP** classifier described in Chapter 5. The interpolation method is adapted from the **mTAN**, described in the previous chapter, in order to take into account the structure of the SITS and to reduce their dimension. Firstly, a description of the method is proposed. Then, the experimental set-up is described.

8.1. Spatially informed interpolator for GP classification

We propose to use end-to-end learning by combining a spatially informed interpolator, denoted as h_{θ_1} with a **SVGP** classifier, denoted as f_{θ_2} . Figure 8.1 represents the workflow for the classification of one irregular and unaligned pixel time series $\mathbf{X}^* = [\mathbf{x}_1^*, \dots, \mathbf{x}_j^*, \dots, \mathbf{x}_D^*]^\top \in \mathbb{R}^{D \times T}$ through its learned latent representation $\mathbf{Z} \in \mathbb{R}^{D' \times R}$, with D the number of spectral features, T the total number of observations, D' the number of latent spectral¹ features and R the number of latent dates. The loss \mathcal{L} is defined in Equation (5.2) in Chapter 5 for the classification with **GP**. Hence, the temporal interpolator is optimized by maximizing the classification accuracy, not the reconstruction error as it is conventionally done with standard interpolation methods. The parameters of h_{θ_1} and f_{θ_2} : θ_1 and θ_2 , respectively, are optimized using this loss.

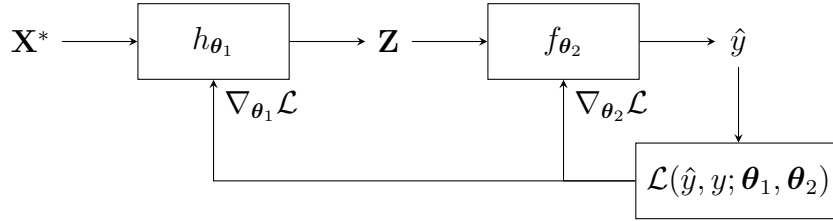


Figure 8.1: End-to-end learning for the classification of one irregular and unaligned pixel time series \mathbf{X}^* and its associated representation \mathbf{Z} .

In the following, a description of the spatially informed interpolator h_{θ_1} is proposed. The description of the **SVGP** classifier was presented in Chapter 5. For clarity, in the following, only the interpolation of one pixel \mathbf{X}^* is considered. Indeed, a matrix or tensorial notation applied to all the pixels (e.g. a batch) would be too cumbersome. The nomenclature used in this chapter and the next chapter is defined in Table 8.1.

¹We take the liberty of using the term "spectral" as a misnomer, as it does not concern the temporal dimension.

Table 8.1.: Nomenclature used in Chapters 8 and 9

Symbol	Meaning
\mathbf{B}	Spectral reduction matrix of size $D' \times D$
β_H	Weight vector of size H
D	Number of spectral features
D'	Number of latent spectral features
E	Output dimension of the temporal positional encoding function ϕ , i.e. temporal embedding dimension
f_{θ_2}	SVGP classifier
F	Output dimension of the spatial positional encoding function φ
γ_{r_l}	Attention weights vector of size T
$\mathbf{\Gamma}$	Attention weights matrix of size $T \times R$, $\mathbf{\Gamma} = [\gamma_{r_1}, \dots, \gamma_{r_R}]$
h_{θ_1}	Spatially informed interpolator
H	Number of heads in h_{θ_1}
L_1, L_2	Number of the neurons for the two layers in the MLP which is used to produce the matrix \mathbf{P}
\mathbf{m}	Vector of masked values of size T , $\mathbf{m} = [m(t_1), \dots, m(t_k), \dots, m(t_T)]^\top$
\mathbf{P}	Spatial positional encoding matrix of size $D \times T$
ϕ	Temporal positional encoding function
$\phi(t_k)$	Embedding vector for the timestamp t_k of size E
$\Phi(\mathbf{T})$	Matrix of embeddings of one head of size $E \times T$, $\Phi(\mathbf{T}) = [\phi(t_1), \dots, \phi(t_T)]$
$\Phi_H(\mathbf{T})$	Matrix of embeddings of all heads of size $H \times E \times T$, $\Phi_H(\mathbf{T}) = \{\Phi_h(\mathbf{T})\}_{h=1}^H$
φ	Spatial positional encoding function
(ψ_1, ψ_2)	Spatial coordinates
R	Number of latent dates
\mathbf{R}	Vector of latent dates of size R , $\mathbf{R} = [r_1, \dots, r_l, \dots, r_R]^\top$
T	Total number of observations
\mathbf{T}	Vector of observations of size T , $\mathbf{T} = [t_1, \dots, t_k, \dots, t_T]^\top$
$\{\omega_p, \alpha_p\}_{p=1}^E$	Trainable parameters used in the temporal positional encoding function ϕ
$\mathbf{W}_k, \mathbf{W}_q$	Trainable embedding matrices of size $E \times E$
$x_j^*(t_k)$	Pixel value for the spectral feature j at timestamp t_k
\mathbf{X}^*	Augmented** matrix of one pixel of size $D \times T$, $\mathbf{X}^* = [\mathbf{x}_1^*, \dots, \mathbf{x}_j^*, \dots, \mathbf{x}_D^*]^\top = [x^*(t_1), \dots, x^*(t_k), \dots, x^*(t_T)]$
$\hat{x}_j(r_l)$	Interpolated value for the spectral feature j at timestamp r_l
$\hat{\mathbf{x}}(r_l)$	Vector of all interpolated spectral features at timestamp r_l
y, \hat{y}	True and predicted class
$\mathbf{z}(r_l)$	Vector of the reduced latent representation of size D' at timestamp r_l
\mathbf{Z}	Matrix of the reduced latent representation of size $D' \times R$, $\mathbf{Z} = [\mathbf{z}(r_1), \dots, \mathbf{z}(r_l), \dots, \mathbf{z}(r_R)]$

** An augmented pixel is defined in Equation (7.2).

8.1.1. Spectro-temporal feature reduction

As a reminder, from Equation (7.24), the **mTAN** described in the previous chapter produces the following interpolated value from the pixel time series \mathbf{X}^* for the latent date r_l :

$$\hat{\mathbf{x}}(r_l) = \mathbf{X}^* \mathbf{\Gamma}_{r_l}^H \boldsymbol{\beta}_H \quad (8.1)$$

with $\mathbf{\Gamma}_{r_l}^H = [\gamma_{r_l}^1, \dots, \gamma_{r_l}^H]$ of size $T \times H$ and $\boldsymbol{\beta}_H$ the weight vector of size H . Therefore, $\hat{\mathbf{x}}(r_l) \in \mathbb{R}^D$ corresponds to the vector of all interpolated spectral features at timestamp r_l . In the following, for simplicity, we will consider only one head i.e. $H = 1$ and we can simplify the Equation (8.1) as:

$$\hat{\mathbf{x}}(r_l) = \mathbf{X}^* \gamma_{r_l}. \quad (8.2)$$

By taking $R < T$, the **mTAN** interpolation allows to perform feature reduction in the temporal domain. From the results in Section 6.5, feature reduction in the temporal and spectral domains is beneficial for the classification task. In the following, we propose to also perform feature reduction in the spectral domain. Therefore, we propose to add a linear layer after the interpolation. Noting \mathbf{B} , the spectral reduction matrix of size $D' \times D$ with $D' \leq D$, the reduced interpolated pixel $\mathbf{z}(r_l)$ can be written as

$$\mathbf{z}(r_l) = \mathbf{B} \hat{\mathbf{x}}(r_l).$$

Therefore, the overall spectro-temporal feature reduction can be written as:

$$\mathbf{Z} = \mathbf{B} \mathbf{X}^* \mathbf{\Gamma} \quad (8.3)$$

with $\mathbf{Z} = [\mathbf{z}(r_1), \dots, \mathbf{z}(r_R)] \in \mathbb{R}^{D' \times R}$ and $\mathbf{\Gamma} = [\gamma_{r_1}, \dots, \gamma_{r_R}] \in \mathbb{R}^{T \times R}$. As defined in Equation (8.3), the matrix $\mathbf{\Gamma}$ does not depend on the spectral features and the matrix \mathbf{B} does not depend on time. Thus, as in Constantin *et al.* [Constantin et al., 2021], the temporal reconstruction does not depend on the spectral features and the spectral feature reduction does not depend on the time. This constrained spectro-temporal structure reduces the complexity of the model: $(DD') + (RT)$ parameters are learned instead of $DTD'R$.

Yet, the spatial information is not taken into account. In the following section, we discuss how the spatial coordinates are integrated in the processing by means of spatial positional encoding.

8.1.2. Spatial positional encoding

We propose to add the spatial information in the estimation of \mathbf{Z} by using a *spatial positional encoding*. As in [Baudoux et al., 2021], the spatial coordinates (ψ_1, ψ_2) are embedded onto a

higher dimensional space of dimension F using φ :

$$\begin{aligned} \varphi : \mathbb{R}^2 &\rightarrow \mathbb{R}^F \\ (\psi_1, \psi_2) &\mapsto \varphi(\psi_1, \psi_2) \\ &= \begin{bmatrix} \sin(\psi_1 \nu_1) \\ \cos(\psi_1 \nu_1) \\ \dots \\ \sin(\psi_1 \nu_q) \\ \cos(\psi_1 \nu_q) \\ \dots \\ \sin(\psi_2 \nu_q) \\ \cos(\psi_2 \nu_q) \\ \dots \\ \sin(\psi_2 \nu_{F/4}) \\ \cos(\psi_2 \nu_{F/4}) \end{bmatrix} \end{aligned}$$

with $\nu_q = 10000^{-(2q)/F}$ and $q \in \{1, \dots, F/4\}$. $\varphi(\psi_1, \psi_2)$ is then given to a two-layer perceptron (first layer with L_1 neurons and second layer with L_2 neurons) with **ReLU** activation functions to obtain a vector of size D which is repeated for each timestamp to get a spatial positional encoding matrix \mathbf{P} of the same shape as \mathbf{X}^* (i.e. $D \times T$). This matrix is added to the augmented matrix for one pixel \mathbf{X}^* before the spectro-temporal interpolation:

$$\tilde{\mathbf{X}}^* = \mathbf{X}^* + \mathbf{P}. \quad (8.4)$$

The parameters of the **MLP** (weights of the layers) are jointly optimized with the time and space informed kernel interpolator and the **SVGP** classifier during the learning step. Finally, using Equation (8.4) in Equation (8.3), the latent representation \mathbf{Z} is:

$$\mathbf{Z} = \mathbf{B}\tilde{\mathbf{X}}^*\mathbf{\Gamma}$$

The **RBF** is used as covariance function for the **SVGP** classifier. Therefore, this covariance function over the latent spectro-temporal representations of two pixels respectively noted \mathbf{Z}^i and $\mathbf{Z}^{i'}$ can be defined as:

$$k(\mathbf{Z}^i, \mathbf{Z}^{i'}) = \exp\left(-\frac{\|\mathbf{Z}^i - \mathbf{Z}^{i'}\|_F^2}{2\ell^2}\right), \quad (8.5)$$

with $\|\cdot\|_F$ and $\langle \cdot, \cdot \rangle_F$ the Frobenius norm and inner product over matrices and ℓ the length-scale parameter of the kernel. The square Frobenius norm can be written as

$$\begin{aligned} \|\mathbf{Z}^i - \mathbf{Z}^{i'}\|_F^2 &= \underbrace{\|\mathbf{B}\mathbf{X}^{i*}\mathbf{\Gamma}^i - \mathbf{B}\mathbf{X}^{i'*}\mathbf{\Gamma}^{i'}\|_F^2}_A \\ &+ \underbrace{\|\mathbf{B}\mathbf{P}^i\mathbf{\Gamma}^i - \mathbf{B}\mathbf{P}^{i'}\mathbf{\Gamma}^{i'}\|_F^2}_B \\ &+ 2 \underbrace{\langle \mathbf{B}(\mathbf{X}^{i*}\mathbf{\Gamma}^i - \mathbf{X}^{i'*}\mathbf{\Gamma}^{i'}), \mathbf{B}(\mathbf{P}^i\mathbf{\Gamma}^i - \mathbf{P}^{i'}\mathbf{\Gamma}^{i'}) \rangle_F}_C. \end{aligned}$$

Terms **A** and **B** correspond to the distance between two pixels for spectro-temporal latent variables and for spatial latent variables, respectively. Term **C** corresponds to an interaction term between spectro-temporal and spatial latent variables. Therefore, we can rewrite Equation (8.5) as:

$$k(\mathbf{Z}^i, \mathbf{Z}^{i'}) = \exp\left(-\frac{A}{2\ell^2}\right) \times \exp\left(-\frac{B}{2\ell^2}\right) \times \exp\left(-\frac{C}{\ell^2}\right), \quad (8.6)$$

Equation (8.6) is very similar to Equation (5.10) i.e. product of covariance functions. However, three differences arise. Firstly, in addition to the spatial covariance function and the spectro-temporal covariance function, there is an additional element: a spatio-spectro-temporal covariance function. We have a supplementary source of information that links spectro-temporal and spatial terms. Secondly, the length-scale ℓ is the same for the three terms in Equation (8.6), whereas in Equation (5.10), a different length-scale was specified for each covariance function. Thirdly, the spatial distance is learned in the term **B**. Indeed, the spatial positional encoding matrix **P** found in the term **B** is produced using a MLP with two layers. To conclude, as the spatial information is already included, combinations of covariance functions will not be retested, as described in Section 5.1.3. Therefore, a simple **RBF** covariance function is used for the **SVGP** classifier.

8.1.3. Trainable parameters

Different parameters, denoted θ_1 , need to be optimized during the training for the time and space informed kernel interpolator h_{θ_1} described in the previous sections. Firstly, regarding the interpolation, we have the parameters of the temporal positional encoding function ϕ , denoted as $\{\omega_p, \alpha_p\}_{p=1}^E$, representing $2E$ parameters. Therefore, for the H heads, they correspond to $H2E$ parameters. Moreover, for each head, the embeddings matrices \mathbf{W}_q and \mathbf{W}_k represent $2E^2$ parameters. Finally, we have H parameters for the linear layer β_H of size H . Then, for the spectral reduction, the weights of the matrix **B** need to be optimized, representing $D'D$ parameters. Finally, for the spatial positional encoding, the weights $L_1L_2 + L_2D$ and also the biases $L_2 + D$ of the two-layer perceptron (i.e. **MLP**), used to obtain a vector of size D , need to be learned. These parameters θ_1 and their corresponding sizes are summarized in Table 8.2. The total number of trainable parameters is given by the following equation:

$$\text{Card}(\theta_1) = 2HE(1 + HE) + DD' + H + L_2(L_1 + D) + D + L_2.$$

In the following, the experimental set-up for the implementation of the spatially informed interpolator for **GP** classification is described.

Table 8.2.: Description of the trainable parameters θ_1 and their corresponding sizes.

Parameters	Size
$\{\omega_p, \alpha_p\}_{p=1}^E$	$2(HE)$
$\mathbf{W}_q, \mathbf{W}_k$	$2(HE)^2$
β_H	H
B	$D'D$
MLP	$L_2(L_1 + D) + D + L_2$

8.2. Experimental set-up

This section describes the experimental set-up implemented, the results associated are presented in the next chapter. As a reminder, the methods used to prepare the training/validation/test sets and to measure the classification accuracy are presented in Appendix A. Chapter 6 showed that the results were better without the *stratification* configuration. Therefore, in the following, the model is learned using pixels over the full area, illustrated in Figure 3.3.

8.2.1. Data set generation

The data is described in Chapter 3. In the following, only specific pre-processing for Chapters 8 and 9 is described. A total of $D = 13$ spectral features were extracted for each pixel \mathbf{x}_i at time t_k . Moreover, two spatial features describe each pixel. However, in contrast with Part II, no temporal resampling is used. The irregular and unaligned augmented time series \mathbf{x}_i^* are directly used with their associated masks \mathbf{m}_i . The union of the acquisition dates of the 27 tiles results in $T = 303$ dates.

Different pixels were extracted randomly from the polygons described in Chapter 3 in order to form three *spatially disjoint* data subsets: *training*, *validation* and *test*. However, unlike Part II, the three data sets are produced over the full study area (i.e. 27 tiles) and not for each eco-climatic region. These three data sets are class-balanced: 4 000 pixels per class in the *training* data set, 1 000 pixels per class in the *validation* data set and 10 000 pixels per class in the *test* data set. The total number of pixels for each data set is provided in Table 8.3. To correctly estimate the classification metrics, 9 runs with different random pixel samplings were done.

Standardization was performed only for the valid acquisition dates and not on raw data. Mean and standard deviation were estimated for each spectral band and for each spectral index on the *training* data set and then used to standardize the other data sets (*validation*, *test*).

Table 8.3.: Number of pixels for each of the three spatially disjoint data subsets: training, validation and test.

	Training	Validation	Test
# of pixels	92 000	23 000	230 000

8.2.2. Methods set-up

Model implementation

The spatially informed interpolator h_{θ_1} described in Section 8.1 is called **Extended multi Time Attention Networks (EmTAN)**. Our model made of the **EmTAN** combined with the **SVGP** classifier is called *EmTAN-SVGP*.

As described in Section 8.1.2, with the use of the spatial positional encoding matrix \mathbf{P} , we decided to use a simple **RBF** as covariance function for the **SVGP** classifier. Therefore, the implementation of the **SVGP** classifier corresponds to the model called λt -GP in Chapters 5 and 6. Same initializations than described in Section 5.2.3 are used.

Besides, the **EmTAN** was implemented using the Pytorch library by Julien Michel².

Competitive methods

Four different classification methods were defined as competitive methods:

1. *Gapfilled-SVGP*: the λt -GP model described in Section 5.2.3 feed with linearly interpolated data. The irregular and unaligned time series from the data sets described in Section 8.2.1 are linearly interpolated every 10 days. The **SVGP** classifier has the same configurations than in Section 5.2.3.
2. *EmTAN-MLP*: a **MLP** classifier combined with **EmTAN**. The **MLP** classifier has the same configurations than in Section 5.2.3. The **MLP** was used as a competitive method because it is a standard approach enabling a similar end-to-end learning.
3. *EmTAN-LTAE*: a **LTAE** classifier combined with **EmTAN**. The **LTAE** classifier has the same configurations than Section 5.2.3. The **LTAE** is used as a competitive method because the best performance results were obtained with this model in Chapter 6.
4. *raw-LTAE*: a **LTAE** classifier without **EmTAN**. Unlike **SVGP** or **MLP** classifiers, the **LTAE** classifier uses attention mechanisms. It may be redundant to use attention mechanisms both in the **EmTAN** and in the **LTAE** classifier. However, the **LTAE** classifier was not defined to deal with the irregular and unaligned time series pixels. Thus, we choose to use the augmented pixel formulation and to provide the mask \mathbf{m} as an additional feature.

For all the competitive methods using the **EmTAN**, the same values were selected for the hyper-parameters at the initialization of the model. Section 9.2 provides a study on the influence of these values on model performance. From this study, a compromise was made between the number of parameters and the performance. Thus, a summary of the selected values is given:

- The number of latent dates is selected as $R = 13$.
- The number of latent spectral features is selected as $D' = 9$.
- The number of heads is selected as $H = 1$.
- The temporal embedding dimension is selected as $E = 64$.
- Finally, the spatial positional encoding matrix \mathbf{P} is used. The spatial coordinates (northing ψ_1 and easting ψ_2) are in meters in the Lambert 93 projection. The dimension F of the spatial positional encoding function φ is selected as $F = 16$. Moreover, the size of the layers in the **MLP** are $L_1 = 16$ and $L_2 = 14$. As a reminder, the **MLP** is used to project the positional encoding vector of size F into a vector of size D . This vector is then repeated in order to obtain a matrix of size $D \times T$ which corresponds to the size of the input \mathbf{X}^* .

²<https://src.koda.cnrs.fr/mmdc/torchmuntan>

As described in Section 5.1.4, the number of IP M in the SVGP classifier has a significant influence on the number of parameters θ_2 . In Chapters 5 and 6, the number of IP was selected as $M = 50$. Therefore, the *Gapfilled-SVGP* model is implemented with $M = 50$. In this chapter and in the next chapter, we propose to increase the number of IP for the *EmTAN-SVGP* model as the number of spectro-temporal features is reduced to $D' \times R$. Figure 8.2 represents the number of trainable parameters θ_2 based on the number of latent spectro-temporal features $R \times D'$ and the number of inducing points M . The reduction of the number of latent spectro-temporal from 481 ($R = 37, D' = 13$) to 117 ($R = 13, D' = 9$) results in a significant reduction of the number of trainable parameters θ_2 as shown in Figure 8.2. From this figure, we can see that it is possible to double the number of inducing points from 50 to 100, while keeping the number of parameters θ_2 with $R = 13, D' = 9$ lower than with 50 inducing points and $R = 37, D' = 13$. In the following, we propose to use $M = 200$ for the SVGP classifier in the *EmTAN-SVGP* model. The influence of the number of IP on the performance of the model is studied in Section 9.2.3.

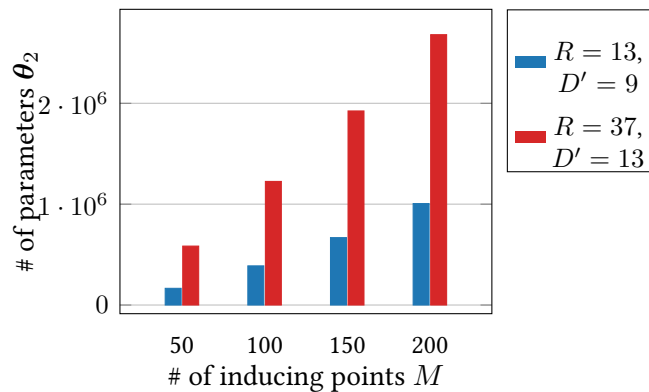


Figure 8.2: Number of trainable parameters θ_2 based on the number of inducing points M and the number of spectro-temporal features $R \times D'$. $R = 37$ and $D' = 13$ corresponds to the configuration of the linear interpolation proposed in Chapters 5 and 6.

The number of trainable parameters for each method is summarized in Table 8.4. The model with the smallest number of trainable parameters is *EmTAN-MLP*. In contrast, the model with the largest number of trainable parameters is *EmTAN-SVGP* with approximately 30 times more parameters than *EmTAN-MLP*. Besides, the *EmTAN-LTAE* model has 4 times fewer trainable parameters than the *raw-LTAE* model. In the *EmTAN-LTAE* model, the classifier has a much smaller number of features than in the *raw-LTAE* model.

For all the models, the Adam optimizer was used. The optimizer parameters (i.e. number of epochs, learning rate and batch size) are given in Table 8.5. They were found by trial and error. The performance of each model in terms of classification accuracy was computed using the OA and F-score, described in Appendix A.2.2. The results are provided in Chapter 9.

Table 8.4.: *Number of trainable parameters for each model.*

Model	Total # of parameters
Gapfilled-SVGP ($M = 50$)	584 200
EmTAN-SVGP ($M = 200$)	1 014 546
EmTAN-MLP	33 113
EmTAN-LTAE	184 376
raw-LTAE	761 380

Table 8.5.: *Parameter values for the Adam optimizer for the models: Gapfilled-SVGP, EmTAN-SVGP, EmTAN-MLP, EmTAN-LTAE and raw-LTAE.*

	Gapfilled-SVGP	EmTAN-SVGP	EmTAN-MLP	EmTAN-LTAE	raw-LTAE
Number of epochs	100	100	300	100	100
Batch size	1024	1024	1000	1000	1000
Learning rate	1×10^{-3}	1×10^{-3}	1×10^{-4}	5×10^{-5}	1×10^{-4}

8.2.3. Map production

Land cover maps were produced using the `iota`² processing chain [Inglada et al., 2016] for all the following models: *EmTAN-SVGP*, *EmTAN-MLP*, *EmTAN-LTAE* and *raw-LTAE*.

Even if the quantitative evaluation was carried out on the 27 tiles, the production of land cover maps for the qualitative evaluation was performed on two adjacent tiles: *T31TCJ* and *T31TDJ*. Inference was performed using the model trained on the 27 tiles with the best overall accuracy over the nine runs. The results are provided in Chapter 9.

9.1. Comparison with competitive methods	226
9.1.1. Quantitative results	226
9.1.2. Qualitative results	231
9.1.3. Robustness to the temporal sampling	234
9.2. Model evaluation	237
9.2.1. Spectral and temporal feature reduction	237
9.2.2. Spatial positional encoding	240
9.2.3. Influence of the number of inducing points	242
9.3. Analysis of the spatially informed interpolator	243
9.3.1. Latent representation	243
9.3.2. Versatility of the similarity kernel	244

In this chapter, the *EmTAN-GP* model is first compared with competitive methods, both quantitatively and qualitatively. Then, an evaluation of the *EmTAN-GP* model is provided. Finally, an analysis of the spatially informed interpolator (i.e. **EmTAN**) is proposed.

9.1. Comparison with competitive methods

Quantitative and qualitative evaluations are proposed, in this section, for the *EmTAN-SVGP* model and its competitive methods in terms of classification accuracy and processing times. The studied models are: *Gapfilled-SVGP*, *EmTAN-SVGP*, *EmTAN-MLP*, *EmTAN-LTAE* and *raw-LTAE*. The *EmTAN-SVGP* model and its competitive models are described in Section 8.2.2. In the last part of this section, an additional comparative study is made between the *EmTAN-SVGP* and the *raw-LTAE* to evaluate the robustness to temporal sampling.

9.1.1. Quantitative results

Classification metrics were computed using the *test* data set composed of 230 000 pixels over the 27 tiles (c.f. Section 8.2.1). Classification metrics were averaged over the nine runs of each model trained with the *training* data set. The global metrics are first studied followed by the metrics per class. Then, confusion matrices are provided and finally, training and prediction times are considered.

Overall accuracy (OA)

The **OA** for each model is given in Figure 9.1. The *EmTAN-SVGP* model is more than 12 points above the *Gapfilled-SVGP* model. Note that results for the *Gapfilled-SVGP* model differ from Section 6.1.1 because the data sets (training, validation and test) are different. From these results, we can state that the learned latent representation \mathbf{Z} obtained by the **EmTAN** contains more meaningful information for the classification task for the **SVGP** classifier compared to the linearly interpolated data. Besides, from the results, the **SVGP** model took greater advantage of the interpolator than the **MLP** or the **LTAE** models. Indeed, the **OA** of the *EmTAN-SVGP* model is seven points above the *EmTAN-MLP* model and around four points above the *EmTAN-LTAE* model. On the other hand, the *EmTAN-SVGP* model is in average two points below the *raw-LTAE* model. Furthermore, the *EmTAN-SVGP* model is the model with the smallest dispersion.

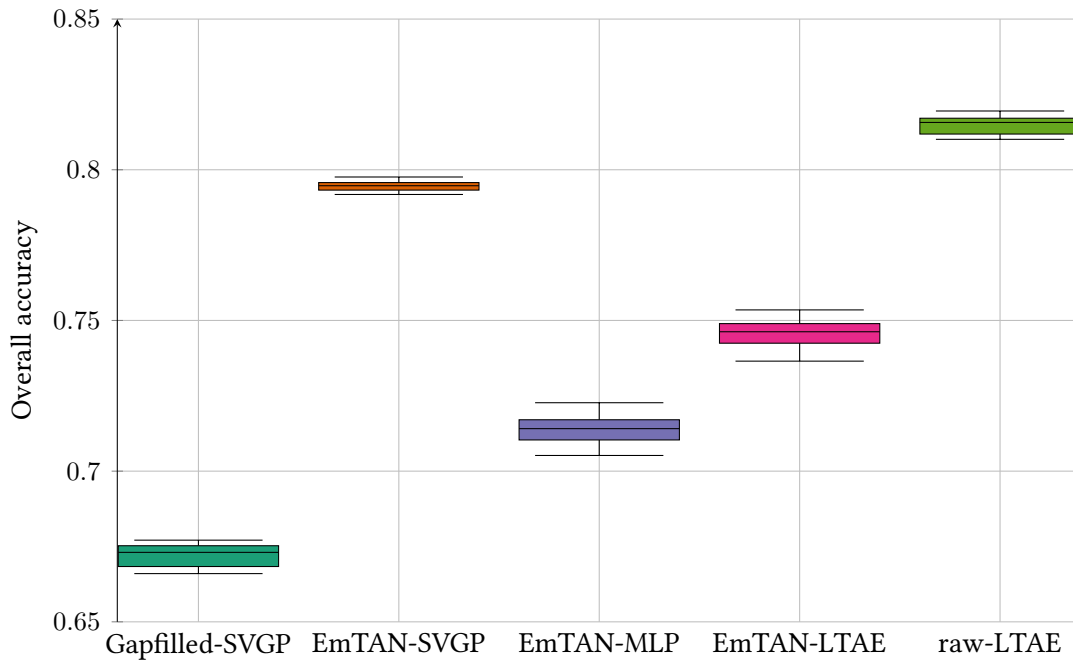


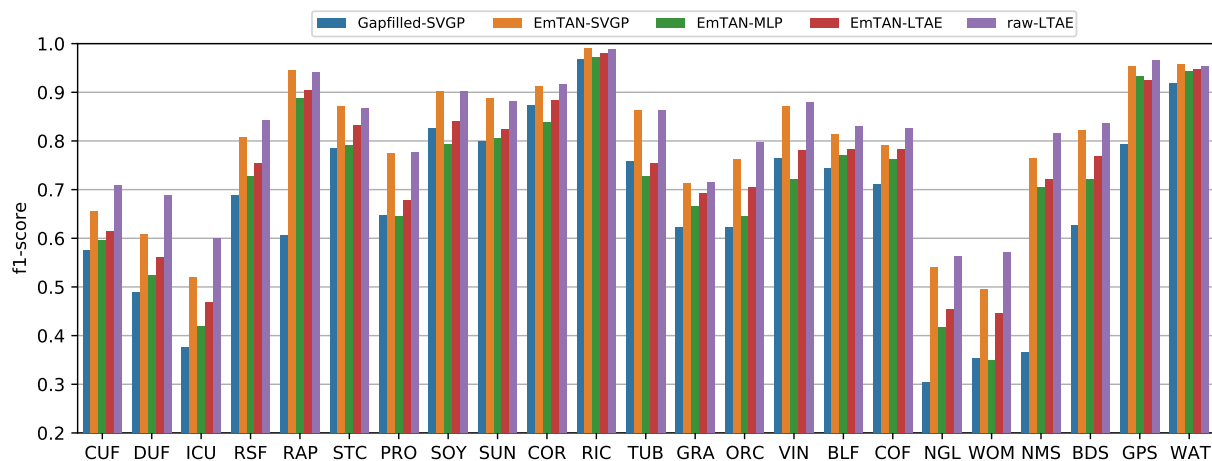
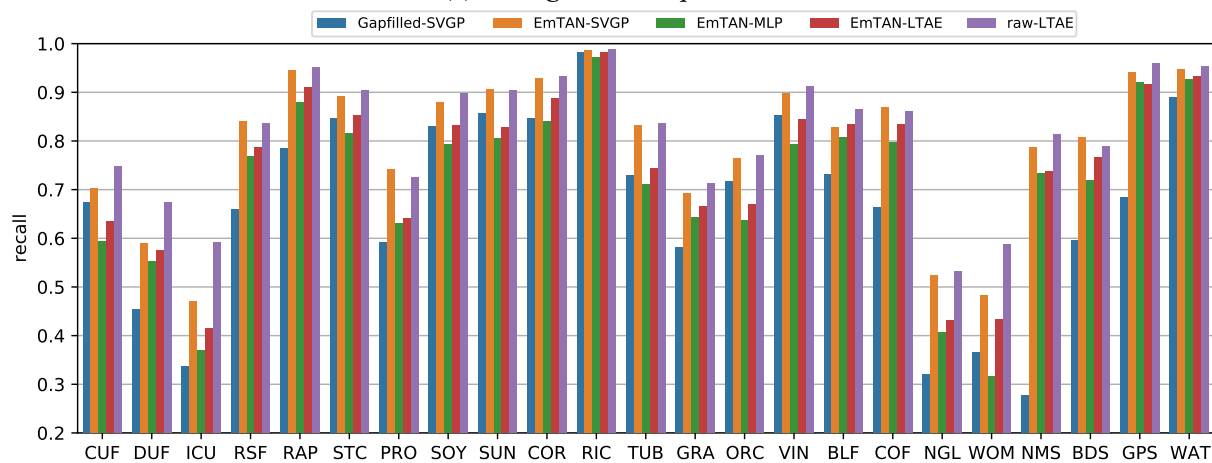
Figure 9.1: Boxplots of the OA for each studied model computed over nine runs. The full configuration of the hyper-parameters is given in Section 8.2.2.

F-score, precision and recall per class

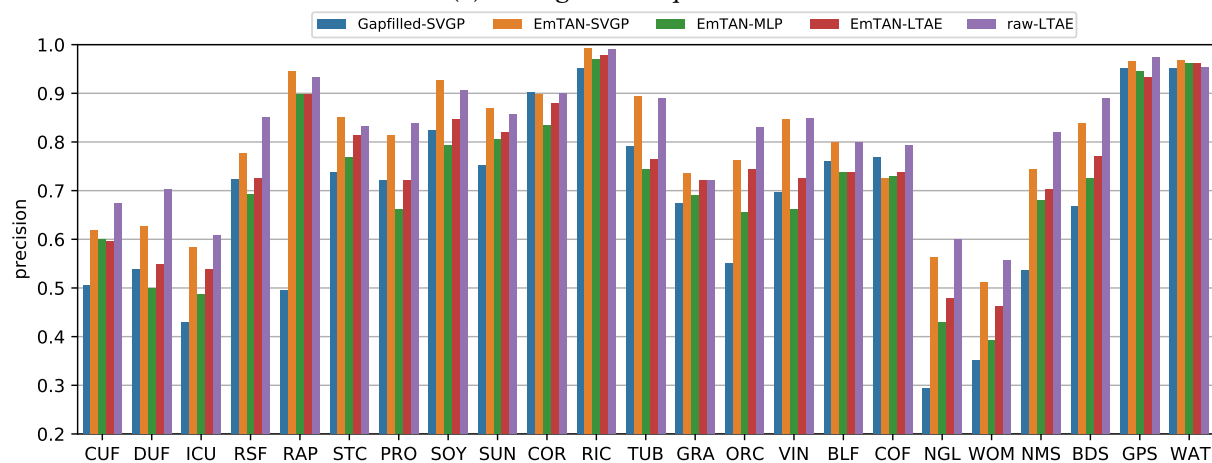
The F-score, recall and precision per class are represented in Figure 9.2. As a reminder, the nomenclature of the classes is presented in Table 3.3.

For all classes, the F-score per class of the *EmTAN-SVGP* model is above the one of the *Gapfilled-SVGP* model, as illustrated in Figure 9.2a. The class with the largest difference is the **RAP** class and there is very few difference for the **RIC** and **WAT** classes. For all classes, *EmTAN-MLP* and *EmTAN-LTAE* have lower F-score than the *EmTAN-SVGP* model. For the classes **CUF**, **DUF**, **ICU**, **RSF**, **BLF**, **COF**, **ORC**, **WOM** and **NMS**, the *raw-LTAE* has a higher F-score than the other methods. **CUF**, **DUF**, **ICU** and **RSF** are urban classes, difficult to discriminate at Sentinel-2 pixel size units using only pixel-wise information. **BLF** and **COF** are very similar classes (i.e. forest classes) as well as **WOM** and **NMS**, therefore, they are difficult to discriminate. Regarding agricultural classes, the *raw-LTAE* has similar F-score values or even lower than the *EmTAN-SVGP*.

For all classes, the recall per class of the *Gapfilled-SVGP* model is below all models except for agricultural classes such as **STC**, **SOY**, **SUN**, **COR** or **TUB**, as illustrated in Figure 9.2b. Same results are found with the precision per class. In conclusion, the results by class are similar to the global results.

(a) Averaged F -score per class.

(b) Averaged recall per class.



(c) Averaged precision per class.

Figure 9.2: Barplots of the averaged metrics per class for each studied model computed over nine runs.

Confusion matrices

Figures 9.3a and 9.3b represent the normalized confusion matrices for the *EmTAN-SVGP* and the *raw-LTAE*, respectively. The normalization is applied over the true labels i.e. the sum of each row is equal to one. We have chosen to present only these two models, as they correspond to the models with the highest overall accuracy. The confusion matrices for *Gapfilled-SVGP*, *EmTAN-MLP* and *EmTAN-LTAE* models are presented in Figure C.1 in Appendix C.

The same confusions that in Section 6.1.1 are found i.e. between CUF, DUF and ICU classes and between NGL and WOM classes. Regarding urban classes (i.e. CUF, DUF and ICU classes), confusions arise only between the urban classes themselves. For the NGL and WOM classes, the confusions are with the other vegetation classes for instance COF classe. The *EmTAN-SVGP* model has more confusions for these two groups of classes than the *raw-LTAE* model. For all the other classes, the confusion values are very similar between *EmTAN-SVGP* and *raw-LTAE*. Concerning the other models (*Gapfilled-SVGP*, *EmTAN-MLP* and *EmTAN-LTAE*), the urban classes have more confusion. Besides, the confusions are more important for the vegetation classes for instance between RAP and NMS for the *Gapfilled-SVGP* model.

True \ Predicted	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT	
CUF	0.70	0.14	0.11	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUF	0.22	0.59	0.08	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.00	0.01	0.01	0.00	0.01	0.02	0.00	0.00	0.00	0.00
ICU	0.17	0.12	0.47	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.00	0.00
RSF	0.02	0.01	0.06	0.84	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.95	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.01	0.89	0.05	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.01	0.00	0.00	0.02	0.08	0.74	0.01	0.01	0.02	0.00	0.03	0.03	0.01	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.88	0.04	0.04	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.91	0.01	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.93	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
TUB	0.01	0.01	0.00	0.00	0.01	0.01	0.02	0.01	0.05	0.01	0.00	0.83	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GRA	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.69	0.03	0.01	0.03	0.02	0.08	0.08	0.01	0.00	0.00	0.00	0.00
ORC	0.00	0.01	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.04	0.76	0.06	0.02	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00
VIN	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.01	0.02	0.90	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.83	0.06	0.02	0.03	0.00	0.00	0.00	0.00	0.00	0.00
COF	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.03	0.87	0.01	0.04	0.02	0.01	0.00	0.00	0.00	0.00
NGL	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.07	0.03	0.01	0.05	0.04	0.52	0.16	0.04	0.01	0.01	0.00	0.00
WOM	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.02	0.01	0.04	0.09	0.19	0.48	0.06	0.02	0.00	0.00	0.00
NMS	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.03	0.06	0.79	0.05	0.02	0.00	0.00
BDS	0.00	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.02	0.01	0.02	0.04	0.81	0.00	0.00	0.01	0.00
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.94	0.00	0.00
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.95	0.00

(a) *EmTAN-SVGP*

Table 9.1.: Averaged (mean \pm standard deviation) training and prediction times (in sec) for each studied model computed over nine runs. The averaged training time is for one epoch.

	Gapfilled-SVGP	EmTAN-SVGP	EmTAN-MLP	EmTAN-LTAE	raw-LTAE
Training times	3.37 \pm 0.20	9.67 \pm 0.22	4.02 \pm 0.19	8.40 \pm 0.79	12.79 \pm 0.19
Prediction times	28.17 \pm 1.28	35.66 \pm 0.94	5.59 \pm 0.46	6.16 \pm 0.27	12.03 \pm 0.47

9.1.2. Qualitative results

In the previous section, the quantitative assessment has been conducted, the qualitative study will now follow. Land cover maps were generated on two different tiles (*T31TCJ* and *T31TDJ*) for the following models: *EmTAN-SVGP*, *EmTAN-MLP*, *EmTAN-LTAE* and *raw-LTAE*. The land cover map of the *Gapfilled-SVGP* model was not generated as its performance results in terms of classification accuracy are not good enough. Besides, this model was already studied in Chapters 5 and 6. All the land cover maps are available for download: [10.5281/zenodo.8033902](https://zenodo.org/record/8033902).

Figures 9.4 and 9.5 represent land cover maps obtained with the studied models on two different agricultural areas around Toulouse. In Figure 9.4, and more precisely in the forest areas, it appears that the *raw-LTAE* model does not correctly predict the **BLF** class. In contrast, the predictions are homogeneous for the models using the *EmTAN*: *EmTAN-SVGP*, *EmTAN-MLP* and *EmTAN-LTAE*. For the *mTANe-MLP* and *mTANe-LTAE* models, in both areas from Figures 9.4 and 9.5, the majority of the crops are surrounded by the class **VIN** whereas it would appear to be hedges instead. Finally, for both areas, the results obtained for the *EmTAN-SVGP*, *EmTAN-MLP* and *EmTAN-LTAE* models showed that the main structures of the map are clearly represented (i.e. crop field border). As with the sum or product of covariance functions discussed in Chapitre 5, the classification maps obtained with EmTAN do not exhibit rounded borders. Therefore, these models provide the spatial information in the temporal interpolation method without spatial over-smoothing.

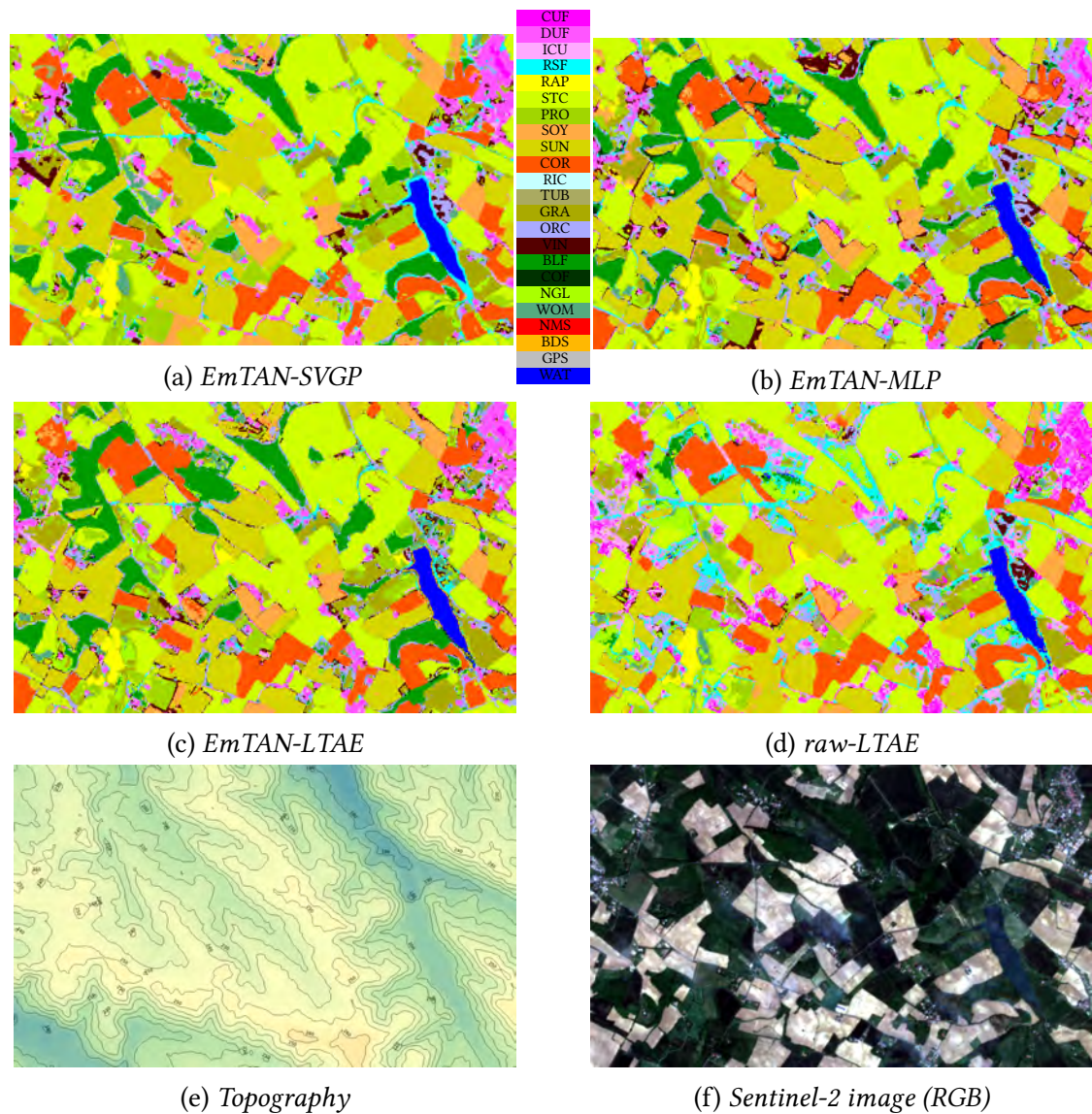


Figure 9.4: Comparison of the land cover maps obtained with each model on an agricultural area around Toulouse (tile T31TCJ). Topography information (30-meter STRM, contours are in meters) and Sentinel-2 image (RGB) (acquisition date: 15/05/18) of the specific zone are provided. Some clouds are visible in the Sentinel-2 image. The studied area is relatively flat (min: 180m, max:260m). There are different types of landscape: towns, crop fields, a lake, forests, etc.

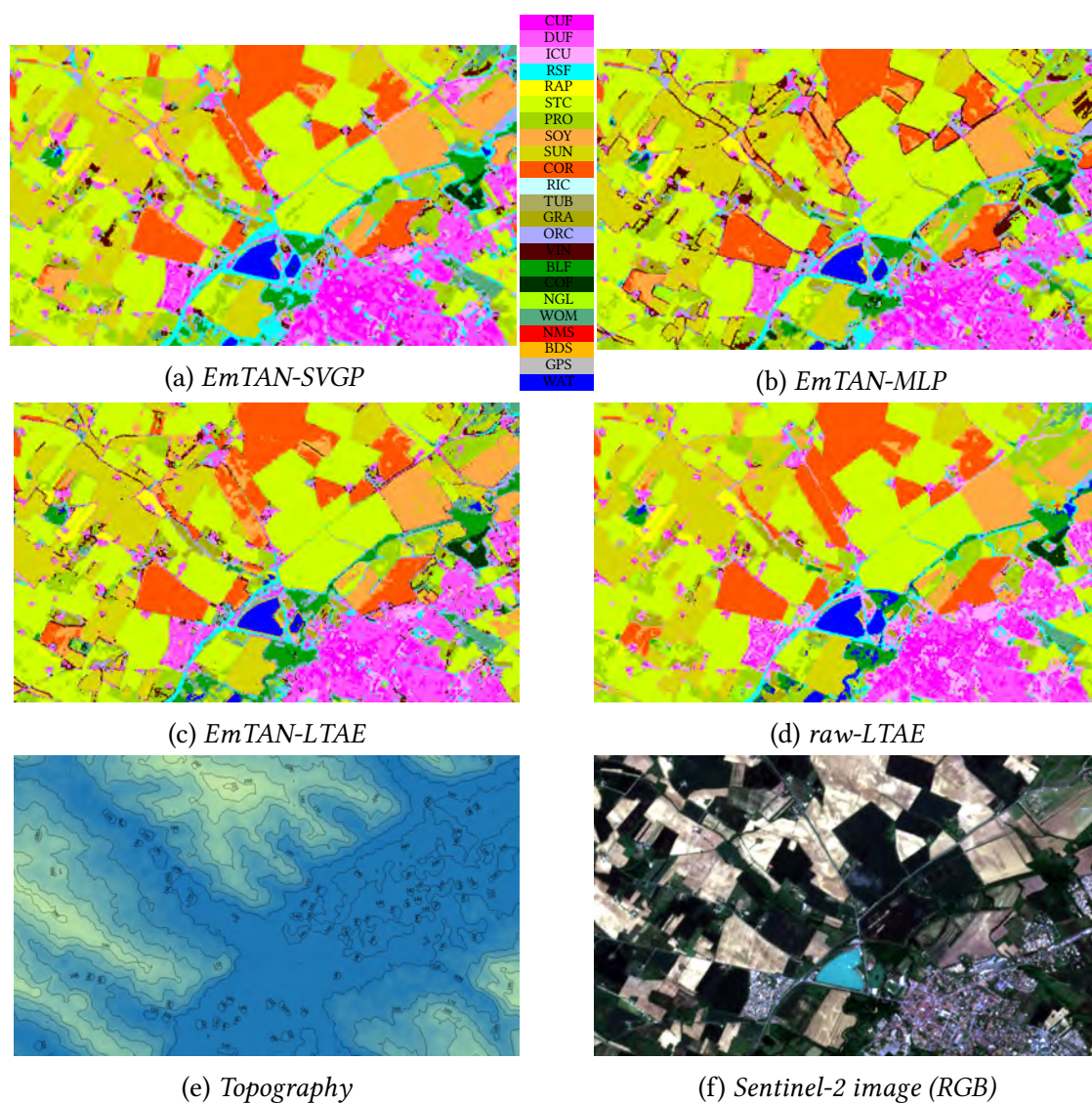


Figure 9.5: Comparison of the land cover maps obtained with each model on an other agricultural area around Toulouse (tile T31TCJ). Topography information (30-meter STRM, contours are in meters) and Sentinel-2 image (RGB) (acquisition date: 15/05/18) of the specific zone are provided. Some clouds are visible in the Sentinel-2 image. The studied area is relatively flat (min: 140m, max:210m). There are different types of landscape: towns, crop fields, a lake, forests, etc.

9.1.3. Robustness to the temporal sampling

As shown in the previous sections, the *raw-LTAE* model has the best classification performances in terms of classification accuracy. However, to compute the inference on a specific area (e.g. on a specific Sentinel-2 tile), the *raw-LTAE* required having seen the whole set of observed dates during the training step. This is not the case for our proposed model which is able to process pixels with any set of observed dates. This can make the time encoding of *raw-LTAE* not robust to variations of the temporal sampling between the train and test sets, with a possible overfit on the training dates. To investigate this possible issue, dates not seen during the training step and used only for the inference were artificially created. They correspond to the original acquisition dates \mathbf{T} from the *training* data set that have been slightly shifted for the *test* data set. Different values for the shift were studied: $\delta = \{0, 1, 2, 3, 5\}$ days. Five days correspond to the maximum number of days between acquisition dates for pixels on two adjacent orbits. In order to have a lighter experiment, the classification metrics were computed on test samples limited to the *T31TCJ* tile for two models *EmTAN-SVGP* and *raw-LTAE* both trained on the 27 tiles.

Figure 9.6 represents the **OA** for the *EmTAN-SVGP* and *raw-LTAE* models computed with artificially shifted acquisition dates. The **OA** of the *EmTAN-SVGP* model is not affected by this temporal shift δ . However, the **OA** of the *raw-LTAE* model is drastically impacted by the temporal shift δ . For a shift of one day, the **OA** is reduced by almost 3 points and it is almost divided by 1.5 with $\delta = 5$ days.

Figure 9.7 represents the F-score per class for the two models *EmTAN-SVGP* and *raw-LTAE*. As found with the **OA**, the *EmTAN-SVGP* model is not affected by this temporal shift. For the *raw-LTAE* model, the most impacted class by the shift is **CUF**: the F-score is divided by approximately 3.6 from 0 to 5 days. In contrast, the classes **SUN** and **RAP** are the least influenced by the shift: the F-score is divided by around 1.2 from 0 to 5 days. The precision and recall per class are presented in Figures C.2 and C.3 in Appendix C.

The use of a time and space informed kernel interpolator makes the *EmTAN-SVGP* model more robust to this shift than the *raw-LTAE* model which uses spectro-temporal attention mechanisms but no interpolation. We conclude that the *raw-LTAE* is more sensitive to dates seen during the training step and may therefore be likely to over-fit.

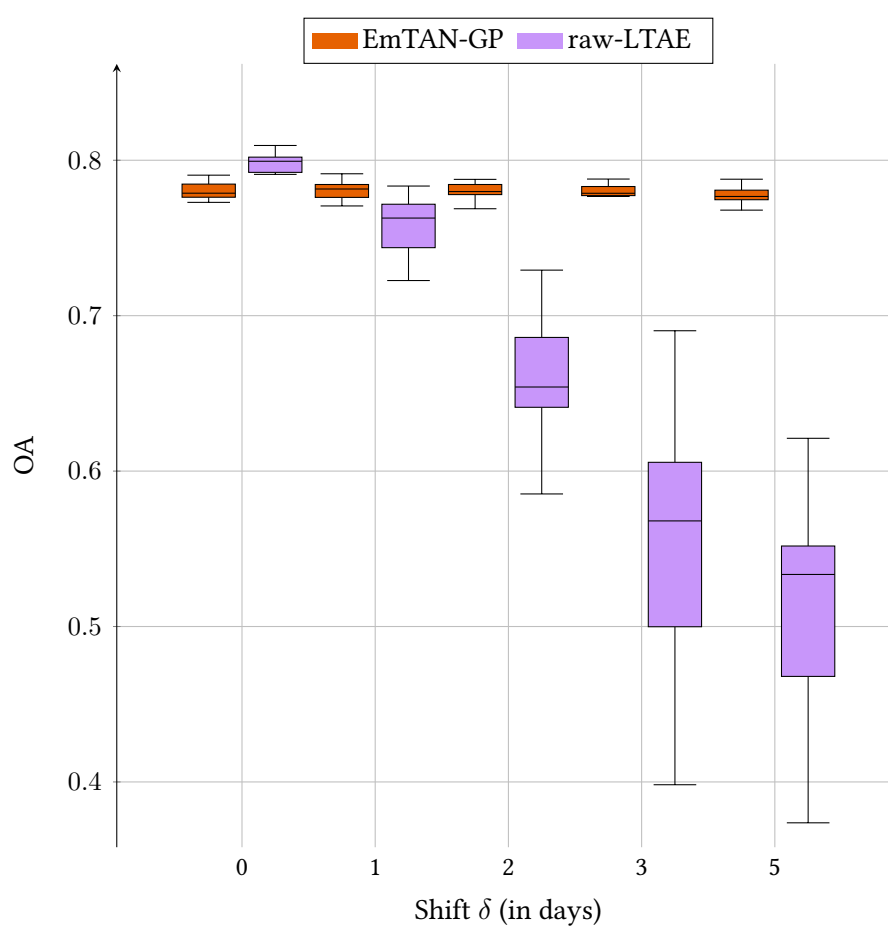


Figure 9.6: Boxplots of the OA for the EmTAN-SVGP and raw-LTAE models computed with the test data set limited to the T31TCJ tile over nine runs. The models were trained and validated on the all 27 tiles. The acquisition dates \mathbf{T} for the test data set were artificially shifted with different values: $\delta = \{0, 1, 2, 3, 5\}$ days.

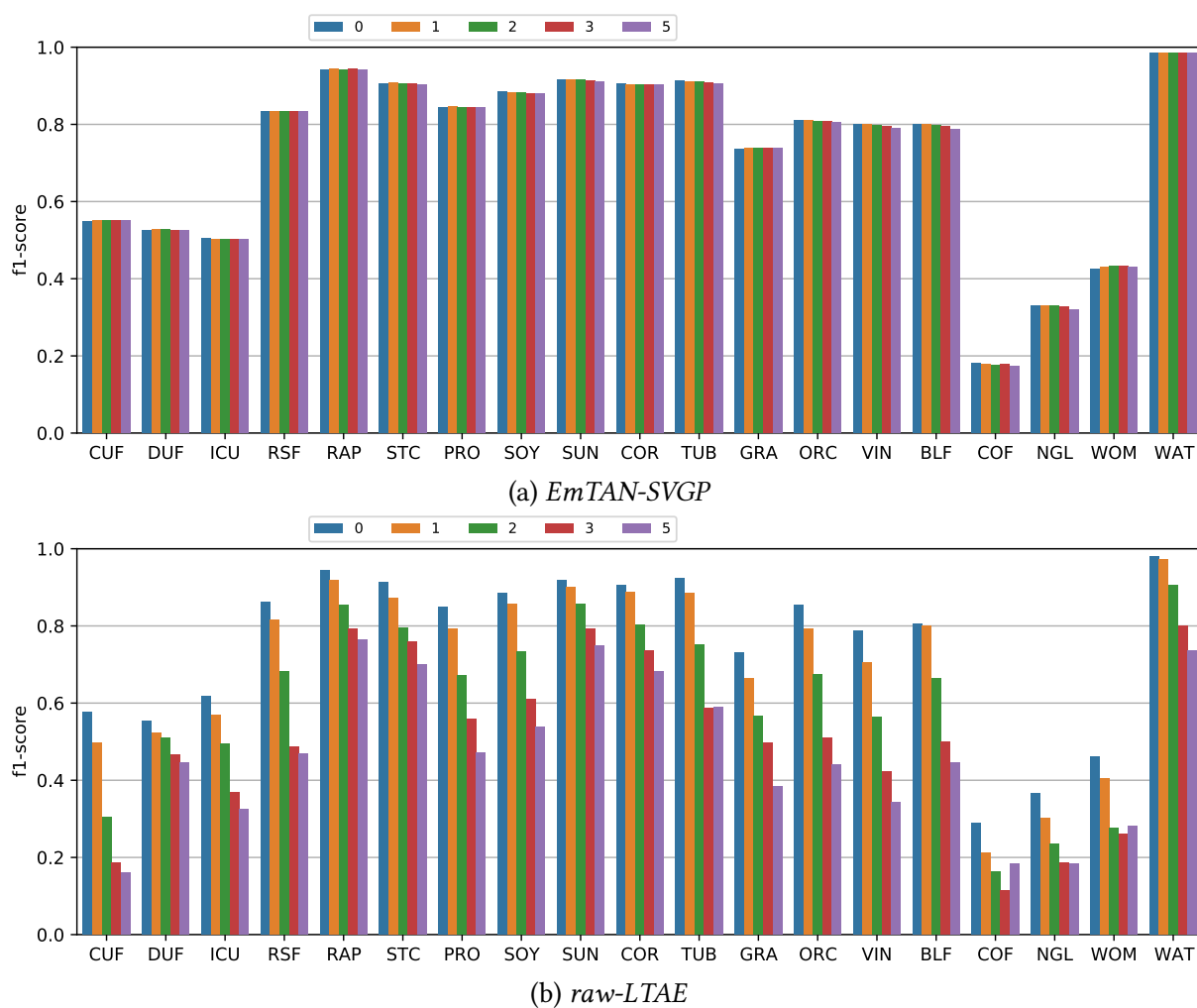


Figure 9.7: Barplots of the F-score per class for the *EmTAN-SVGP* and *raw-LTAE* models computed with the test data set limited to the *T31TCJ* tile over nine runs.

9.2. Model evaluation

This section investigates the influence of latent representation sizes on the classification accuracy and processing times as well as the use of the spatial positional encoding matrix. Then, the latent representation and the similarity kernel learned by the interpolator are discussed.

9.2.1. Spectral and temporal feature reduction

As described in Section 8.1.3, the cost of the estimation of parameters θ_2 of the SVGP is highly dependent on the number of spectro-temporal features $d = R \times D'$ with the following term: $M \times d$. A high number of parameters is time-consuming and reducing the number of features d could be beneficial for the convergence of the algorithm (both in terms of time and quality of the optimum).

Figures 9.8 and 9.9 represent respectively the averaged OA and the averaged training times computed with different number of latent dates $R = \{5, 7, 13, 15, 19, 25, 37\}$, different number of latent spectral¹ features $D' = \{4, 6, 9, 10, 11, 12, 13\}$ and different number of heads $H = \{1, 2, 3\}$ over nine runs.

The number of heads H has a little impact on the classification performances. Indeed, for $D' = 13$ and $R = 37$, the OA goes from 77.44 for $H = 1$ (Figure 9.8a) to 77.79 for $H = 3$ (Figure 9.8c). Similar results are found, for $D' = 4$ and $R = 5$, the OA goes from 73.45 to 73.66. Besides, from $H = 1$ to $H = 3$, the training time can be increased by a factor of two: 1317 seconds to 2644 seconds, for $D' = 13$ and $R = 37$, as shown in Figures 9.9a and 9.9c, respectively.

Hence, we set $H = 1$ for all the remaining experiments. The number of latent dates and latent spectral features has a greater influence on the OA. Indeed, from $D' = 13$ and $R = 37$ to $D' = 4$ and $R = 5$, the OA is reduced by almost four points, as illustrated in Figure 9.8a. However, they correspond to extreme values. It is possible to reduce the number of latent dates and latent spectral features with a negligible effect on the OA. Indeed, reducing R from 37 to 13 and D' from 13 to 9 result to an OA from 77.44 to 77.23. With $R = 13$ and $D' = 9$, the number of parameters θ_2 is divided by a factor four, i.e. from 584 200 to 165 600 parameters. Moreover, the training times is divided by a factor two.

From Figures 9.8 and 9.9, we notice that R has a huge influence on the training times but a slight one on the OA. It is the opposite for D' : its value has a significant influence on the OA and not so much on the training times.

In the next section, using these results, we will focus on the *mTAN-SVGP* model with $R = 13$ latent dates, $D' = 9$ latent spectral features and $H = 1$ head.

¹As in Chapter 8, we take the liberty of using the term "spectral" as a misnomer, as it does not concern the temporal dimension.

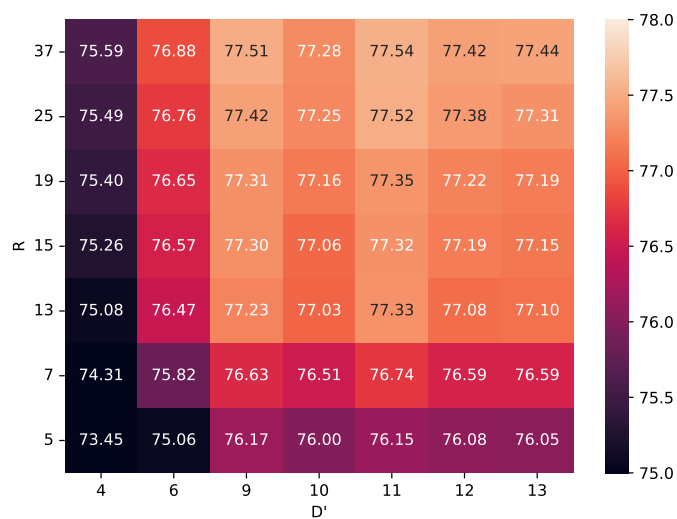
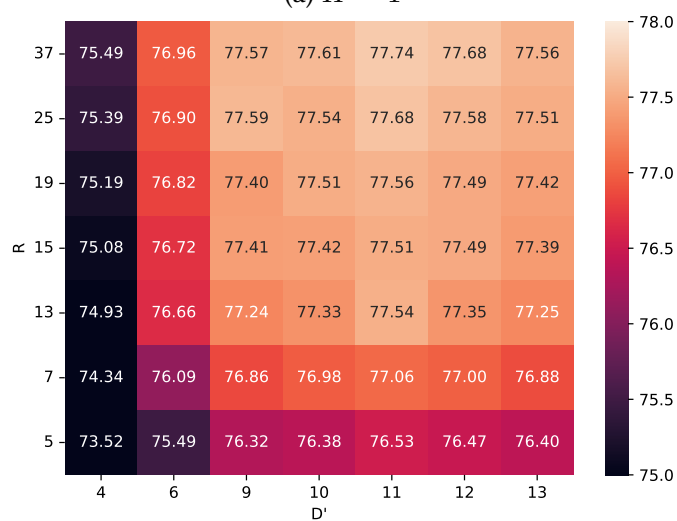
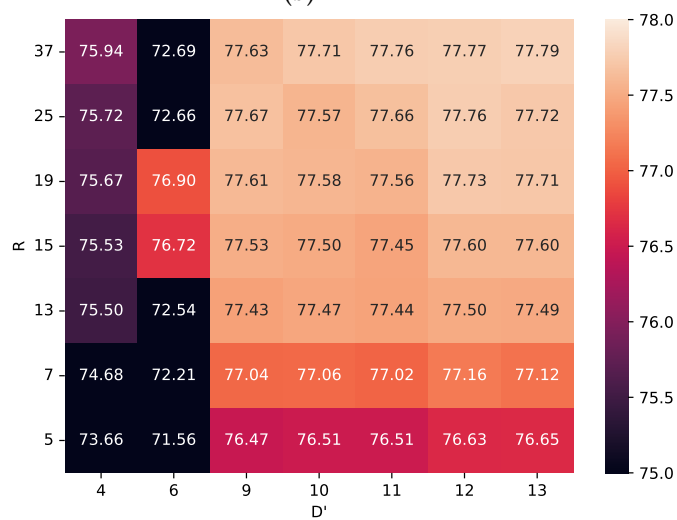
(a) $H = 1$ (b) $H = 2$ (c) $H = 3$

Figure 9.8: Averaged OA (mean in % computed over nine different runs) with R the number of latent dates, D' the number of latent spectral features and H the number of heads.

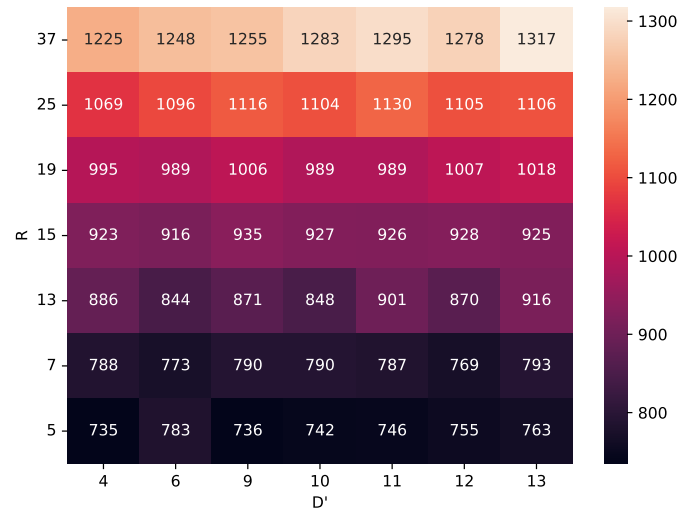
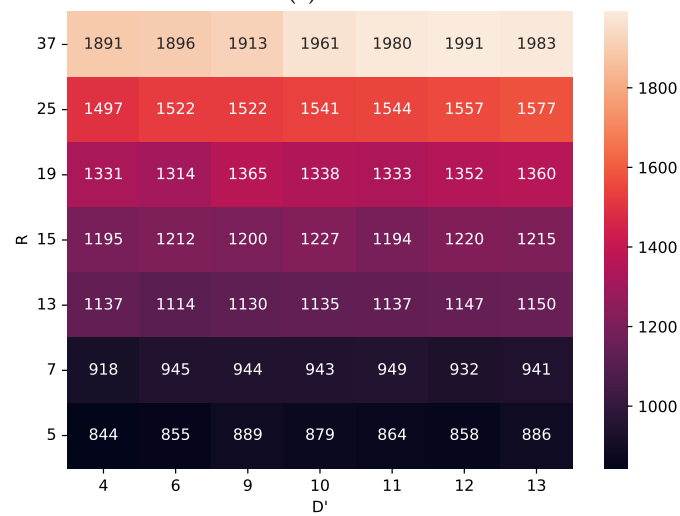
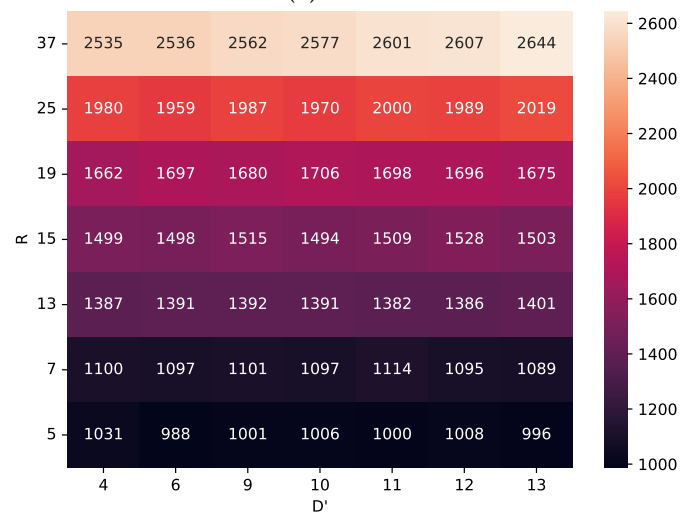
(a) $H = 1$ (b) $H = 2$ (c) $H = 3$

Figure 9.9: Averaged training times in seconds (mean computed over nine different runs) with R the number of latent dates, D' the number of latent spectral features and H the number of heads.

9.2.2. Spatial positional encoding

Table 9.2 represents the averaged OA and the averaged training times with and without the spatial positional encoded matrix \mathbf{P} , defined in Section 8.1.2. The use of the spatial positional encoding increased by nearly 1.5 points the OA. In Part II, for the SVGP classifier with linearly interpolated data, we have shown that the OA was increased by nearly two points by using the spatial information through a spatial covariance function. It is quite comparable to the results obtained with the spatial positional encoding, as we have shown that Equation (8.6) is very similar to Equation (5.10). Regarding the training times, the differences are negligible and are probably due to the HPC (e.g. waiting time, availability of resources, task priority).

Figure 9.10 represents the value of \mathbf{P} for the features number 4 and number 12. This value was computed using spatial coordinates on a regularly spaced grid over the 27 tiles. Figures 9.10a and 9.10b exhibit smooth spatial transitions and anisotropic spatial similarity. However, we did not observe any explainable spatial pattern on the different runs: no latitude effect, for instance. Besides, between different runs, we do not specifically find the same spatial patterns.

Table 9.2.: Averaged OA and averaged training times (in sec) (mean $\% \pm$ standard deviation computed over nine runs) with and without the spatial positional encoded matrix \mathbf{P} .

	Without \mathbf{P}	With \mathbf{P}
Averaged OA	77.23 ± 0.17	78.63 ± 0.16
Training times	870 ± 57	834 ± 45



(a) feature 4



(b) feature 12

Figure 9.10: Spatial positional encoding \mathbf{P} computed over a regular grid of spatial coordinates. Two different features and studied: feature 4 and feature 12 (background map © OpenStreetMap contributors).

9.2.3. Influence of the number of inducing points

It is known that the learning capacity of the SVGP classifier is strongly influenced by the number of IP M , and a trade-off should be found between the computational complexity and the learning capacity [Hensman et al., 2015]. From the results found in Section 9.2.1, the number of spectro-temporal features is drastically reduced with no loss in terms of classification accuracy. Therefore, the number of inducing points can be increased without increasing the number of trainable parameters too much, as illustrated in Figure 8.2.

By benefiting of a reduced computational load thanks to the dimension reduction, we perform several experiments with increasing number of inducing points $M = \{100, 150, 200\}$. Table 9.3 represents averaged OA and training times computed with different number of inducing points. With $M = 200$, the OA is increased by almost one point compared to $M = 50$. Training time is only slightly affected by this increase of the number of inducing points, i.e. 834s to 967s. Hence, spectro-temporal reduction made possible to use higher number of inducing points and thus to increase the performances, while maintaining a reduced computational load.

Table 9.3.: Averaged OA and averaged training times (in sec) (mean $\% \pm$ standard deviation computed over nine runs) for different number of inducing points M .

	Number of inducing points M			
	50	100	150	200
Averaged OA	78.63 \pm 0.16	79.20 \pm 0.21	79.43 \pm 0.29	79.48 \pm 0.17
Training time	834 \pm 45	910 \pm 78	921 \pm 29	967 \pm 22

9.3. Analysis of the spatially informed interpolator

This section analyzes the extended mTAN. The results were computed with the *EmTAN-SVGP* model. Firstly, the latent representation obtained by the interpolator is discussed. Then, a detailed study of the attention weights learned by the interpolator is proposed.

9.3.1. Latent representation

It is possible to visualize the learned latent representation \hat{x}_j . Figure 9.11 represents the comparison of three NDVI time series profiles from one pixel labeled as COR: the raw data, the gapfilled data (i.e. linearly interpolated) and the learned latent representation obtained by our time and space informed kernel interpolator.

The latent representation obtained in Figure 9.11 clearly does not minimize the reconstruction error of the original time series. For instance, the second minimum of the NDVI observed around the day of the year 280 is not reconstructed. Yet, this is the representation that conducts to minimize the classification loss function of the SVGP. Besides, latent dates do not necessarily correspond to real dates, so time distortion can occur. The aim is to align the data in order to maximize classification performance.

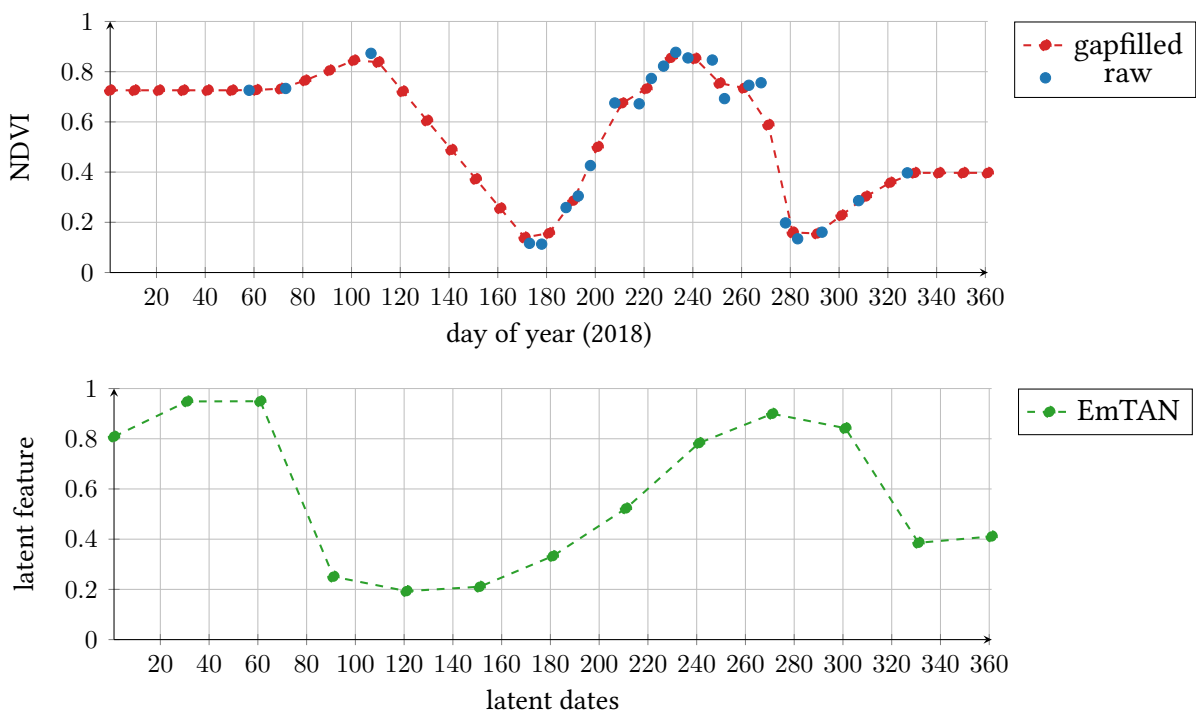


Figure 9.11: NDVI time series profiles for a pixel labeled COR. Blue points \bullet correspond to the raw data, the observations flagged as clouds or cloud shadows have been removed in order to have a comprehensive plot. Red points \bullet correspond to the value obtained with a linear interpolation with an interval of 10 days for a total of 37 dates. Green points \bullet correspond to the latent representation \hat{x}_j with $j = \text{NDVI}$ obtained before the spectral reduction ($D' = 9$).

9.3.2. Versatility of the similarity kernel

By using *attention* and *embedding* mechanisms, the similarity kernel is able to adapt to the pixel temporal sampling. The versatility of the similarity kernel can be shown by computing the attention value γ_{r_l} defined in Equation (7.23) for different latent dates r_l and for different sets of observed dates \mathbf{T} . In Figure 9.12, three different latent dates are studied $r_l = 1$, $r_l = 181$ and $r_l = 361$. For each latent date r_l , two different sets of observed dates \mathbf{T} are considered. Firstly, the attention value was computed with a regular set of observed dates: $\mathbf{T} = \{1, \dots, 365\}$ with an interval of $\tau = 1$ day (in red in Figure 9.12). Then, the attention value was computed with the set of observed dates from the i th pixel with $\mathbf{T} = \mathbf{T}^i$ (in blue in Figure 9.12).

From Figure 9.12, we can see that contrary to conventional RBF kernel, the learned kernel is not centered on the latent date r_l . For instance, with $r_l = 1$, the learned kernel is centered at around day 75. It thus adapts itself according to the latent date r_l and the available observations. Moreover, for the set of observed dates $\mathbf{T} = \{1, \dots, 365\}$ (i.e. continuous red line), the bandwidth is larger for the latent date $r_l = 361$ than for the latent date $r_l = 181$. Such property is referred to as a *variable-bandwidth* kernel in the statistical literature [Terrell and Scott, 1992]. While it has shown to perform well on several cases, such kernel was difficult to optimize with standard statistical models. Using the proposed framework, the optimization is efficient, scales well and can handle any timestamp.

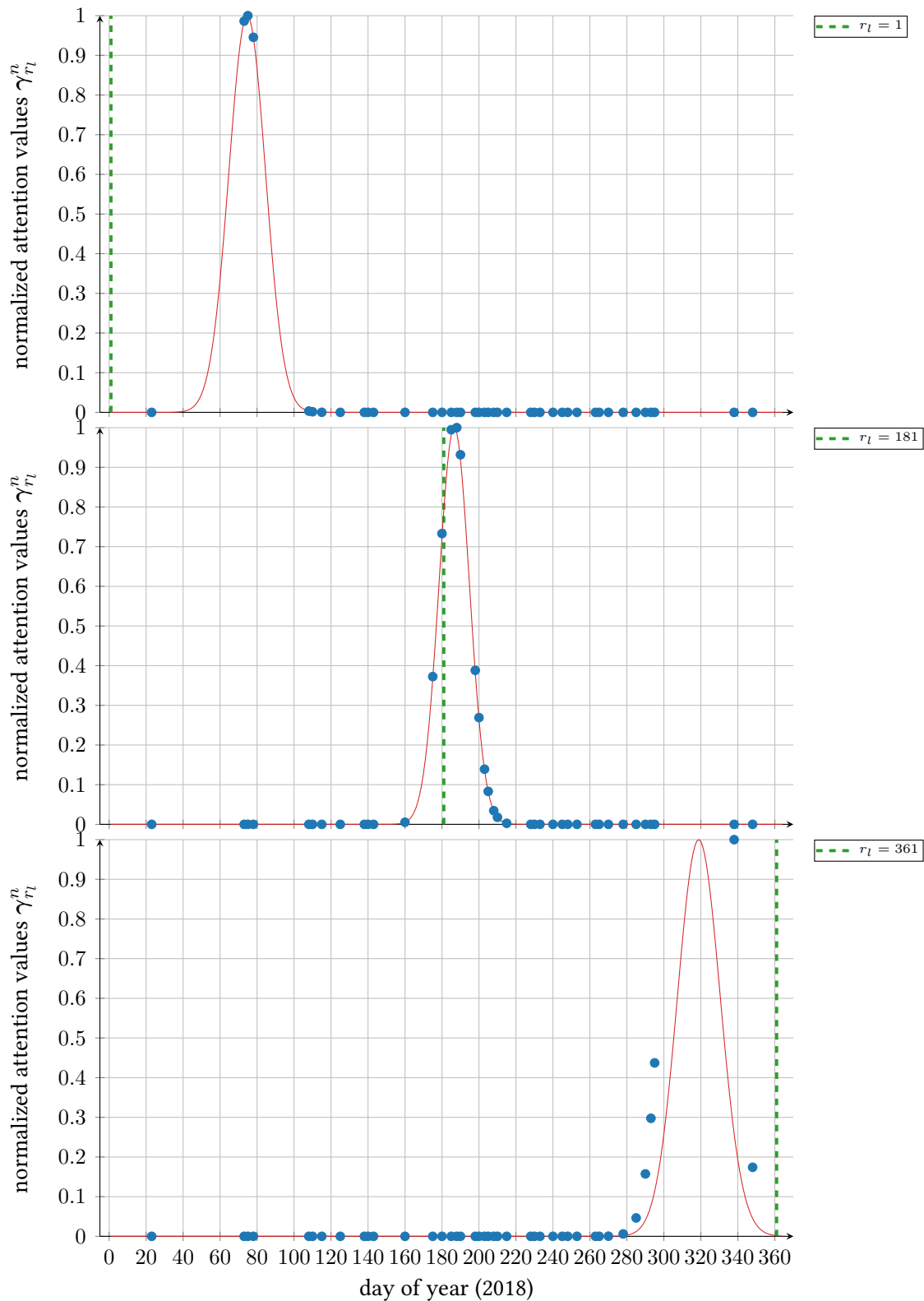


Figure 9.12: Normalized attention values $\gamma_{r_l}^n = \frac{\gamma_{r_l}}{\max(\gamma_{r_l})}$ computed on three different latent dates $r_l = 1, r_l = 181$ and $r_l = 361$. — corresponds to $\gamma_{r_l}^n$ computed with $\mathbf{T} = \{1, \dots, 365\}$ with a regular interval of $\tau = 1$ day. Blue points \bullet correspond to $\gamma_{r_l}^n$ computed with $\mathbf{T} = \mathbf{T}^i$ for the i th pixel. The attention value plotted is normalized in order to have the same vertical scale.

Part IV.
General Conclusion

10.1. Summary

The context of this thesis was to monitor ecosystems using artificial intelligence algorithms applied to remote sensing data. Specifically, this thesis was focused on the development of algorithms for **Land Use and Land Cover (LULC)** pixel-based classification at a large scale. We decided to adopt the same configurations than the **Centre d’Expertise Scientifique Occupation des SOs (CES OSO)** framework i.e. the use of the Sentinel-2 **Satellite Image Time-Series (SITS)** and the use of the same reference data (i.e. 23 classes). The current learning workflow consisted of **Random Forests (RF)** classification with a spatial stratification using linearly interpolated Sentinel-2 **SITS** [Inglada et al., 2017].

The contributions of the thesis are developed in Parts **II** and **III**. In Part **II**, I first investigated the spatial variability issue, and in particular the spatial discontinuities that occur at the boundary of two spatial strata for the **CES OSO** approach. Then, in Part **III**, I proposed to deal with irregular and unaligned time series during the learning step. Experiments were conducted with large scale Sentinel-2 **SITS** of one full year.

In Part **II**, I proposed an approach based on **Stochastic Variational Gaussian Processes (SVGP)**. By combining sparse methods with **Variational Inference (VI)**, this model is able to scale to massive data sets. This is the first time that a model with **Gaussian Processes (GP)** has been implemented under operational **LULC** classification conditions. Better results than the current **CES OSO** based approach have been achieved. Indeed, in terms of accuracy, **GP** models outperformed **RF** and **Multi-layer Perceptron (MLP)** methods. However, they were slightly worse than structured **Deep Learning (DL)** models i.e. the **Lightweight Temporal Attention Encoder (LTAE)** model. Besides, thanks to a spatial covariance function combined with a spectro-temporal covariance function, the spatial variability was taken into account. Therefore, with **GP** models, the spatial stratification was not needed anymore. Moreover, spatial discontinuities between adjacent regions were more severe for **RF** models. Additionally, unlike **DL** methods, it was possible to interpret the parameters of this model (e.g. the spatial values of the **Inducing Points (IP)** or the values of the mixing matrix). Finally, the Bayesian nature of the **GP** has enabled the estimation of the posterior predictive distributions which can be used to assess prediction uncertainties.

In Part **III**, I have developed an end-to-end model that combines a time and space informed kernel interpolator with the **SVGP** classifier. The fixed and reduced size latent representation obtained with the interpolator is given to the **SVGP** classifier and all the parameters are jointly optimized during the training of the classifier. We were able to process irregular

and unaligned **SITS** without any re-sampling preprocessing. This method outperformed the simple **SVGP** classifier with linearly preprocessed interpolated data. In comparison to the previous method, temporal and spectral reductions were performed jointly but independently in each dimension, for the classification task. This constrained spectro-temporal structure has enabled to reduce the number of parameters and therefore has allowed to use more **IP**, resulting in improved classification performance. Furthermore, the spatial information was taken into account but in a different way from the previous method. Indeed, the spatial information was introduced directly in the **SVGP** classifier but with the learned representation through a spatial positional encoding matrix. Finally, for the inference, the end-to-end learning model did not require the common temporal grid used during the training step and was not sensitive to the set of available dates during inference.

In this thesis, I focused on the analysis of classification metrics for the study area, which consists of 27 tiles in the south of France. I also produced **LULC** maps for two tiles. However, I did not apply the model over the entire metropolitan France as it is done for the **CES OSO**. Based on what I have studied, I can conclude that this model provides better performance than the current framework. Firstly, we no longer need to use spatial stratification. Then, irregular and unaligned time series can be directly used without a preprocessing step. Moreover, we are no longer limited by the number of training samples, as it can be the case with **RF**. However, on the other hand, this model is computationally intensive and requires the use of **GPU**. Besides, this model does not require a lot of hyper-parameters tuning as it is the case with **DL** methods, but it does require a bit more than **RF**. Therefore, it might be interesting to know whether the conclusions we have reached are the same as those we might find for the entire area of France. Finally, different aspects such as label noise, pluriannual classification or the estimation of continuous variables, were not treated in this thesis, but they are discussed in Section 10.2.

10.2. Perspectives

The perspectives of this work are multiple and concern both methodological and thematic aspects. Short-term perspectives are presented first, they correspond to developments that I could have done if I had more time during my thesis. They are followed by mid- and long-term ones, which correspond to outlooks requiring more time (another thesis or a postdoc).

10.2.1. Short-term

Methodological

- The simplified version of **Linear Model of Co-regionalization (LMC)** defined in Equation (4.21) was used for the **SVGP** classifier. A perspective could be to modify its definition. For instance, Liu *et al.* [Liu et al., 2022] proposed to add an additional layer in Equation (4.21) which depends on the inputs. The L latent **GP** are mapped into a higher dimension latent space. For several data sets, this method gives better results than the traditional LMC. In our case, this additional layer could depend on the spatial information. Therefore, the latent processes could be specialized over a part of the area.
- Regarding the **Extended multi Time Attention Networks (EmTAN)**, the potential of the multi-head attention has not been fully exploited. Indeed, only one head was used and

as shown in Section 9.2.1 the performance with an increasing number of heads was not satisfying. A perspective could be to inform the different heads with the spatial information: the linear layer β_H in Equation (8.1) could be replaced by the output of a perceptron using the spatial positional encoding. A softmax function could be used after the MLP to produce β_H in order to weight each head. This could help the heads to specialize spatially and differentiate themselves.

- The latent dates \mathbf{R} used in the EmTAN were selected with a regular interval starting with the first day of the year (i.e. $r_1 = 1$). Experiments were also made with random irregular sampling and with selected dates from the histogram of available dates. With these experiments, we did not observe any influence on the model performance. However, a perspective could be to let the EmTAN learn the position of the latent dates which are useful for the classification task. Besides, it could be very interesting to use pluriannual time series instead of a one-year time series (i.e. multitemporal data fusion [Ghamisi et al., 2019]). The EmTAN could learn periodic patterns over the years.

Thematic

- The *EmTAN-SVGP* was applied on 27 tiles in the south of metropolitan France. A perspective could be to apply the model over all metropolitan France, as it is done for OSO. Comparisons with the CES OSO based approach (i.e. RF with spatial stratification) would be interesting.
- The classification in areas with high relief have shown poorer results. Therefore, a perspective could be to extend the spatial information by using the altitude in addition to the longitude and the latitude. Moreover, in addition to spatial data (i.e. altitude, longitude and latitude), a perspective could be to use the other topographic data to construct the spatial positional encoding in order to take better account of climatic, geographical and other differences.
- To improve the classification in areas with high cloud cover, one perspective could be to add radar data, such as Sentinel-1, in addition to the Sentinel-2. Indeed, the main advantage of radar sensors is that they are not limited by the weather conditions such as clouds. The ability of the interpolator to process unaligned time series would make the fusion of radar data and optical data straightforward.

Besides, to improve the classification of certain classes, other types of optical sensors can be used, for instance Landsat 8 with its thermal bands. Results showed that LULC maps can be improved by the use of thermal bands [Sun and Schulz, 2015]. The THRSNA satellite is scheduled to launch in 2025 and will provide thermal data with a finer spatial resolution (i.e. 60m) and with a more frequent revisit cycle (i.e. every 3 days) [Roujean et al., 2021]. Thanks to the interpolator, the fusion can also be straightforward.

Finally, informations from non satellite sources can be added to the time series. For example, the World Clim 2 data set provides averaged monthly climate data from weather stations at a spatial resolution of 1km for the period 1970–2000 [Fick and Hijmans, 2017]. This climate data set corresponds to: temperature (minimum, maximum and average), precipitation, solar radiation, vapour pressure and wind speed. The EmTAN can also be used to merge all the time series.

All this additional information could improve the representation for the classification task.

- In the manuscript, we focused only on the classification task. However, our method can also be applied for a regression task. In this case, the model is considerably simplified: the **Evidence Lower Bound (ELBO)** from Equation (5.2) is used and no **Monte Carlo (MC)** sampling is required, as the likelihood is Gaussian. The estimation of grassland mowing dates is a regression problem of high interest at my current laboratory. Indeed, grassland late mowing helps to maintain biodiversity, as it allows plant and animal species (particularly birds) to complete their reproductive cycles [Smith et al., 2000], [Brown and Nocera, 2017]. Recently, a large number of publications propose to use machine learning algorithms for the estimation of grassland mowing events with Sentinel-2 time series [Garioud et al., 2019], [Holtgrave et al., 2023]. A perspective could be to apply our *EmTAN-SVGP* model for the mapping of grassland mowing events. It could be beneficial for this task as uncertainties are provided with our model. Indeed, the **SVGP** classifier allows to obtain the posterior predictive distribution which we did not use for the classification task as it represents a large amount of data. However, for a regression task, this amount is largely reduced and can be very useful. Uncertainties in this regression problem are a valuable complementary information.

10.2.2. Mid- and long-term

Methodological

- The performance of the **SVGP** is highly limited by the number of **IP** but also by their dimension. Some works have shown that the prediction can be inaccurate for large-scale data that are not inherently low-rank structured, since we assume $M \ll N$ [Wu et al., 2021], [Tran et al., 2021]. Therefore, Wu *et al.* [Wu et al., 2022] proposed a new model: the **Variational Nearest Neighbor Gaussian Processes (VNNGP)**. It performs a mean-field variational approximation instead of full-rank variational approximation. This sparse approximation allows **VNNGP** to use more **IP** than other variational methods. It employs a K nearest-neighbor approximation. A perspective could be to use **VNNGP** instead of **SVGP**. Due to their variational inference nature, **VNNGP** can be applied to classification tasks.
- In this manuscript, a large number of reference data were available. However, some zones contain fewer reference data than others, as illustrated in Figure 3.14. These zones correspond to mountainous areas that are generally more difficult to access but also can be less urbanized or less exploited agriculturally. A perspective could be to use unlabeled data from these limited reference data zones in addition to the labeled data. One possible implementation would be to constrain a similar latent representation with the unlabeled data. Therefore, a loss composed of a reconstruction term and a prediction term could be used.

Thematic

- Currently, the OSO map gives the dominant class over a full year. If two different agricultural classes are successively present on a crop during the year, only the main one

will be identified. A perspective could be to have a classification with a label that varies over time. For instance, information on the label could be provided on a quarterly basis. This might allow to study double cropping (i.e. two different crops cultivated on the same field in the same year: the second crop is seeded after the first has been harvested) or to identify catch crops (i.e. quick-growing crop that grows between two regular crops [Lockhart and Wiseman, 2014]). Therefore, it can provide information about the agricultural practices used.

- The OSO map is currently produced using data from January to December. A perspective could be to produce the map before the end of the year, for example in June. Getting a LULC map in June could make it possible to estimate and/or anticipate crop irrigation in summer, for instance. By producing an early classification, not all the reference data are available and therefore the method must be adapted accordingly. Some works are currently being carried out in this direction [Lin et al., 2022], [Rußwurm et al., 2023a].
- The current trend is to produce LULC maps using newly available satellite images. However, a large number of archives are available (e.g. satellite images archives such as SPOT World heritage¹, but also aerial images archives such as "Remonter le temps"²). In order to better monitor climate change, it is also interesting to monitor large-scale changes over long term periods. A perspective could be to produce LULC using these archives. In general, images from archives have lower resolutions (spatial, temporal and spectral) and can be of very poor quality. Therefore, they are more difficult to analyze and process compared to newly available satellite images. A possible solution is to use recent satellite images in order to process more easily older images i.e. temporal domain adaptation [Chen et al., 2020], [Capliez et al., 2023]. Since it is possible to perform prediction with recent satellite images, the same latent representation for both older and recent time series could be produce (transfer learning [Demir et al., 2013]). For instance, the EmTAN could be used to produce these representations and place them on the same regular grid. My future research will be in this direction, as in January 2024, I am starting a post-doc with the objective of classifying trees in the Pyrenees since the 1960s using aerial images with artificial intelligence algorithms.

¹<https://regards.cnes.fr/user/swh/modules/60>

²<https://remonterletemps.ign.fr/>

CONCLUSION EN FRANÇAIS

L'objectif général de cette thèse a concerné la surveillance des écosystèmes à l'aide d'algorithmes d'intelligence artificielle appliqués aux données de télédétection. Plus précisément, cette thèse s'est concentrée sur le développement d'algorithmes pour la classification supervisée de cartes d'occupation du sol à grande échelle. Les mêmes configurations que celles proposées par Centre d'Expertise Scientifique Occupation des SOls (CES OSO) du pôle Theia³ ont été utilisées, c'est-à-dire l'utilisation de séries temporelles d'images satellites (SITS) Sentinel-2 et l'utilisation des mêmes données de référence (i.e. 23 classes). Le processus opérationnel d'apprentissage actuel consiste en une classification supervisée à l'aide de forêts aléatoires (RF). Une stratification spatiale est appliquée et les SITS Sentinel-2 sont ré-échantillonnées linéairement [Inglada et al., 2017].

Les contributions de la thèse ont été développées dans les Parties II and III. Dans la Partie II, nous avons d'abord étudié la question de la variabilité spatiale, et en particulier les discontinuités spatiales qui se produisent à la limite de deux strates spatiales pour l'approche CES OSO. Ensuite, dans la Partie III, nous avons proposé de traiter directement les séries temporelles irrégulières et non alignées pendant l'étape d'apprentissage. Les expérimentations ont été menées avec des SITS Sentinel-2 d'une année entière dans une zone comprenant tout le sud de la France.

Dans la Partie II, nous avons proposé une approche basée sur les processus gaussiens variationnels stochastiques (SVGP). En combinant des méthodes parcimonieuses avec l'inférence variationnelle, notre modèle a été capable de traiter des gros volumes de données. C'est la première fois qu'un modèle avec des processus gaussiens (GP) a été mis en œuvre dans des conditions opérationnelles de classification de cartes d'occupation du sol. De meilleurs résultats que ceux obtenus avec l'approche actuelle CES OSO (RF avec stratification spatiale) ont été obtenus. En effet, en termes de précision, les modèles basés sur les GP ont surpassé les modèles basés sur des RF et les perceptrons multicouche (MLP). Cependant, ils ont été légèrement moins performants que les modèles structurés d'apprentissage profond, c'est-à-dire le modèle basé sur un encodeur temporel utilisant la auto-attention (LTAE). Grâce à une fonction de covariance spatiale combinée à une fonction de covariance spectro-temporelle, la variabilité spatiale a été prise en compte. Par conséquent, avec les SVGP, la stratification spatiale n'était plus nécessaire. De plus, les discontinuités spatiales entre régions adjacentes étaient plus marquées pour les modèles RF. Contrairement aux méthodes d'apprentissage profond, il a été possible d'interpréter les paramètres de notre modèle (par exemple, les coordonnées spatiales des points induisants ou les valeurs de la matrice de mélange). Enfin, la nature bayésienne des GP a permis d'estimer les distributions prédictives *a posteriori* utilisées pour évaluer les incertitudes de prédiction.

Dans la Partie III, nous avons développé une méthode d'apprentissage de bout en bout: un interpolateur à noyau prenant en compte l'information spatiale combiné avec les SVGP définis

³<https://www.theia-land.fr/>

précédemment. La représentation latente de taille fixe et réduite obtenue avec l'interpolateur a été donnée aux SVGP et tous les paramètres ont été optimisés conjointement par rapport à la tâche de classification. Les SITS irrégulières et non alignées ont été utilisées sans aucun prétraitement. Cette méthode s'est avérée plus performante que les SVGP avec des données ré-échantillonnées par interpolation linéaire dans une étape de prétraitement distincte de la classification. Par rapport à la méthode précédente, les réductions temporelles et spectrales ont été effectuées conjointement, mais indépendamment, pour la tâche de classification. La mise en place d'une structure spectro-temporelle contrainte a permis de réduire le nombre de paramètres et donc d'utiliser plus de points induisants, ce qui a permis d'améliorer les performances de la classification. De plus, l'information spatiale a été prise en compte, mais d'une manière différente de la méthode précédente. En effet, l'information spatiale a été introduite directement dans les SVGP à l'aide de la représentation qui a été apprise par le biais d'une matrice d'encodage positionnel spatial. Nous avons montré que notre modèle d'apprentissage ne nécessite pas pour l'inférence la grille temporelle commune utilisée pendant l'étape d'apprentissage.

Dans cette thèse, nous nous sommes concentrés sur l'analyse des métriques de classification pour notre zone d'étude, qui se compose de 27 tuiles dans le sud de la France. Nous avons également produit des cartes d'occupation du sol pour deux tuiles. Cependant, nous n'avons pas appliqué notre modèle à l'ensemble de la France métropolitaine comme c'est le cas pour le CES OSO. Néanmoins, sur la base de ce que nous avons étudié, nous pouvons conclure que notre modèle offre de meilleures performances que le cadre actuel. Tout d'abord, nous n'avons plus besoin d'utiliser la stratification spatiale. Ensuite, les séries temporelles irrégulières et non alignées peuvent être directement utilisées sans étape de prétraitement. Nous ne sommes plus limités par le nombre d'échantillons d'apprentissage, comme cela peut être le cas avec les RF. En revanche, notre modèle est très gourmand en ressources informatiques et nécessite l'utilisation de GPU. Peu de réglages sont à faire au niveau des hyper-paramètres pour notre modèle, contrairement aux méthodes d'apprentissage profond, mais tout de même un peu plus qu'avec les RF. Il pourrait donc être intéressant de savoir si les conclusions auxquelles nous sommes parvenus sont les mêmes que celles que nous pourrions trouver pour l'ensemble de la France. Enfin, différents aspects tels que le bruit des classes, la classification pluriannuelle ou l'estimation de variables continues, n'ont pas été traités dans cette thèse.

Part V.
Appendices

This appendix describes several tools used in supervised classification.

A.1. Data set selection

Working with supervised methods induces the use of different labeled data sets. A first data set, called *training* data set, is used to train the model. After completing the training, a separate set of data, called *test* data set, is used to estimate the performance of the model in terms of classification accuracy [Bishop, 1995]. Section A.2 details the different methods used to estimate this performance. Algorithms which minimize a loss function need the use of another data set called *validation* data set. Such as for the previous data sets, this data set contains different samples. It helps to validate our model performance during training. Indeed, as described in Section A.2, it is possible to monitor the validation loss and detect over-fitting [Bottou et al., 2018].

Each data set contains separate samples from the other data sets. Different methods are used to select the samples from reference data. The simplest one is called simple random sampling, as shown in Figure A.1a. Every sample has the same chance of being selected. However, if the reference data set is imbalanced, which is often the case in land cover, it can create data sets without samples from minority classes and with samples only from majority classes. Thus, two techniques can be used to resample this data set in order to have a better distribution. The first one is called random oversampling. It duplicates samples in the minority classes. The second one is called random undersampling. It deletes samples in the majority classes. However, this can be tedious. Therefore, with imbalanced data sets, using stratified sampling is more appropriate. Stratified sampling divides the population into strata based on relevant characteristics (for example the class label), as illustrated in Figure A.1b. While maintaining the proportions, random sampling can be used to select the samples in each strata. Thus, the distribution of classes in each of the train, validation, and test data sets is preserved. The sampling methods have a great impact on the performances [Li et al., 2021]. Moreover, the size of the different data sets (number of samples) has an influence on the classification performances. Indeed, large training data sets provide better classification performances than smaller ones [Foody et al., 2006].

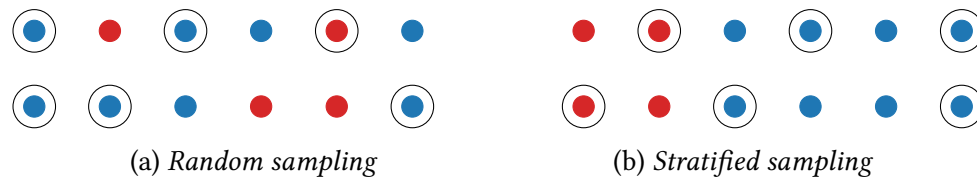


Figure A.1: Examples of sampling methods for two different data sets.

A.2. Validation and accuracy assessment

A.2.1. Training and validation losses

In order to have good performances, it is important to monitor the model during training thanks to training and validation losses. Methods which minimize a loss function, such as DL models, allow the study of performances during the training step. Indeed, training and validation losses are good tools that can indicate if the training goes well. Figure A.2 represents training and validation losses in the case of over-fitting, under-fitting or good-fitting.

In over-fitting, the model works very well with the training data set but poorly with validation and test data sets. It leads to poor generalization. The model learns too much the training data. In this case, the validation loss will diverge from the training loss after being close to it.

In under-fitting, the model performs poorly not only on the training set data but also with the validation and test data sets. The model is too simple. In this case, the training loss does not stabilize.

When the model is well trained (optimum or good-fitting), the training and validation losses are close to each other and the training loss is stable. Thus, the model performs well with the training, validation and test data sets.

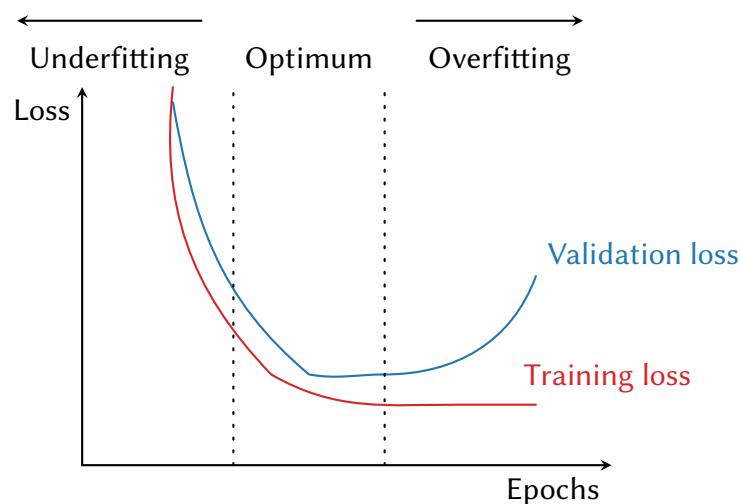


Figure A.2: Training and validation losses in over-fitting, under-fitting and good-fitting (optimum).

A.2.2. Confusion matrices and metrics

After the training step, confusion matrices and metrics are used in order to evaluate the performances of the model. They are computed with the test data set. It is also possible to compute them during the training step with the validation data set. The confusion matrix, also called error matrix, is a technique for summarizing the performance of a supervised classification algorithm. It is a square matrix with the number of rows and columns equal to the number of classes. In classification, different metrics are widely used such as: overall accuracy, F-score, recall, precision, kappa. The confusion matrix and the metrics are first defined in the case of binary classification and then in the case of multi-class classification.

Binary classification

In binary classification, the classes are labeled either positive (1) or negative (0). The confusion matrix is defined such as:

		Predicted Class	
		Positive	Negative
True Class	Positive	TP	FN
	Negative	FP	TN

with True Positive (TP): number of predictions where the classifier correctly classifies the positive class as positive; False Negative (FN): number of predictions where the classifier incorrectly classifies the positive class as negative; False Positive (FP): number of predictions where the classifier incorrectly classifies the negative class as positive; True Negative (TN): number of predictions where the classifier correctly classifies the negative class as negative.

From this confusion matrix, it is possible to compute different metrics used to assess the classification accuracies [Congalton, 1991]:

- **Overall accuracy (OA).** It represents the ratio of correct predictions and can be computed as

$$OA = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (A.1)$$

If the OA reaches 1, it means that all the predictions are correct.

- **Precision.** It is defined as the proportion of the positive class predictions that were actually correct. The precision can be computed as:

$$P = \frac{TP}{(TP + FP)} \quad (A.2)$$

It corresponds to the first row of the confusion matrix. If there are no false positives, the precision is equal to 1.

- **Recall/Sensitivity.** It is defined as the proportion of actual positive class samples that were identified by the model. The recall can be computed as:

$$R = \frac{TP}{(TP + FN)} \quad (A.3)$$

It corresponds to the first column of the confusion matrix. If there are no false negatives, the recall is equal to 1.

- **Specificity.** It is the ability of a classification model to measure the rate of actual negatives identified correctly. The specificity can be computed as:

$$\text{Sp} = \frac{TN}{(TN + FP)} \quad (\text{A.4})$$

- **F-score.** It is the harmonic mean of the model's precision and recall. The F-score can be computed as:

$$\text{F-score} = \frac{2TP}{(2TP + FP + FN)} = \frac{2P \times R}{P + R} \quad (\text{A.5})$$

It is useful to find the best trade-off between these two metrics [Sasaki et al., 2007]. Such as the previous metrics, it ranges from 0 to 1, with 1 the best value.

- **Kappa.** It is a metric that takes into account correct classifications due to random chance. Kappa can be written such as:

$$\text{Kappa} = \frac{2 \times (TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)} \quad (\text{A.6})$$

Kappa values range from -1 to 1 . Different interpretations of the Kappa values can be found. [Landis and Koch, 1977] suggests that if Kappa is equal to zero the agreement is no better than what would be obtained by chance. Negative values imply that the agreement is worse than what would be expected by chance. From 0 to 0.2 , it is a slight agreement, from 0.21 to 0.4 it is a fair agreement, from 0.41 to 0.6 it is a moderate agreement, from 0.61 to 0.8 it is a substantial agreement and finally from 0.81 to 1 it is a perfect agreement. [Ranganathan et al., 2017] proposed an other interpretation suggesting that Kappa values below 0.6 indicate a significant level of disagreement. These two interpretations are among a large number of other possible interpretations [Monserud and Leemans, 1992, Fleiss et al., 2013].

Multi-class classification

In multi-class classification with C classes, the confusion matrix is defined as:

		Predicted Class		
		1	...	C
True Class	1	n_{11}	...	n_{1C}
	n_{ij}	...
	C	n_{C1}	...	n_{CC}

where n_{ij} is the number of predictions where the classifier classify the true class i as j . A normalized confusion matrix is defined as:

		Predicted Class		
		1	...	C
True Class	1	$\frac{n_{11}}{\sum_{j=1}^C n_{1j}}$...	$\frac{n_{1C}}{\sum_{j=1}^C n_{1j}}$
	$\frac{n_{ij}}{\sum_{j=1}^C n_{ij}}$...
	C	$\frac{n_{C1}}{\sum_{j=1}^C n_{Cj}}$...	$\frac{n_{CC}}{\sum_{j=1}^C n_{Cj}}$

Indeed, the total of each row i is equal to 1 such as $\sum_{j=1}^C \left(\frac{n_{ij}}{\sum_{j=1}^C n_{ij}} \right) = 1$.

The different metrics defined in the case of binary classification can also be calculated in the multi-class classification:

- **Overall accuracy (OA)**. It can be computed as:

$$OA = \frac{1}{N} \sum_{i=1}^C n_{ii} \quad (\text{A.7})$$

with $N = \sum_{i,j=1}^C n_{ij}$ the total number of predictions. If a class is under-represented (very few samples), all the predictions can be attributed to the most abundant class. This would produce a very good overall accuracy but the under-represented class would have a very low accuracy. Thus, it is also important to calculate metrics for each class such as precision, recall or F-score.

- **Precision** for the class i , also called user's accuracy, can be computed as

$$UA_i = \frac{n_{ii}}{\sum_{j=1}^C n_{ji}} \quad (\text{A.8})$$

It corresponds to the i th column of the confusion matrix.

- **Recall** for the class i , also called producer's accuracy, can be computed as

$$PA_i = \frac{n_{ii}}{\sum_{j=1}^C n_{ij}} \quad (\text{A.9})$$

It corresponds to the i th row of the confusion matrix.

- **F-score** for the class i can be calculated from the precision and recall as

$$F\text{-score}_i = \frac{2 \times UA_i \times PA_i}{UA_i + PA_i} \quad (\text{A.10})$$

- **Kappa**. In multi class classification, the computation of this metric is very different from the binary case. Indeed, it can be computed as:

$$\text{Kappa} = \frac{N \sum_{i=1}^C n_{ii} - \sum_{i=1}^C \left(\sum_{j=1}^C n_{ij} \times \sum_{j=1}^C n_{ji} \right)}{N^2 - \sum_{i=1}^C \left(\sum_{j=1}^C n_{ij} \times \sum_{j=1}^C n_{ji} \right)} \quad (\text{A.11})$$

$$= \frac{OA - p_h}{1 - p_h} \quad (\text{A.12})$$

with $p_h = \frac{1}{N^2} \sum_{i=1}^C \left(\sum_{j=1}^C n_{ij} \times \sum_{j=1}^C n_{ji} \right)$ the measure of agreement. Whereas it is widely used in multi class classification, [Foody, 2020] suggests to remove the Kappa metric from the analysis in land cover classification. The main reasons are that p_h does not represent correctly the rate of correct classification due to chance [Foody, 1992]. Moreover, [Pontius Jr and Millones, 2011] showed that it can be difficult to compare Kappa values between different classification systems. For these reasons, in this thesis, Kappa will not be used.

Metrics are quantitative measurements used to assess the performances of one model. The previous list is not exhaustive, other metrics can be defined [Grandini et al., 2020]. However, these previous metrics are the most used in land cover classification. In the following, these metrics are used to evaluate the performances of the classification models.

Computing the mean and the variance of these metrics over multiple data sets, provides a more accurate and robust estimate of the model's performance. Different re-sampling methods are used to compute these multiple data sets. If the reference data is limited, k -fold cross validation is used. It consists in splitting the reference data set into k equal parts called folds. The model is trained k times with $k - 1$ folds used for the training data set and 1 fold used for the test data set. Each time, the test data set has a different fold, as represented in Figure A.3a. If enough reference data are available, k training and k test data sets are selected with different random seeds, as illustrated in Figure A.3b.

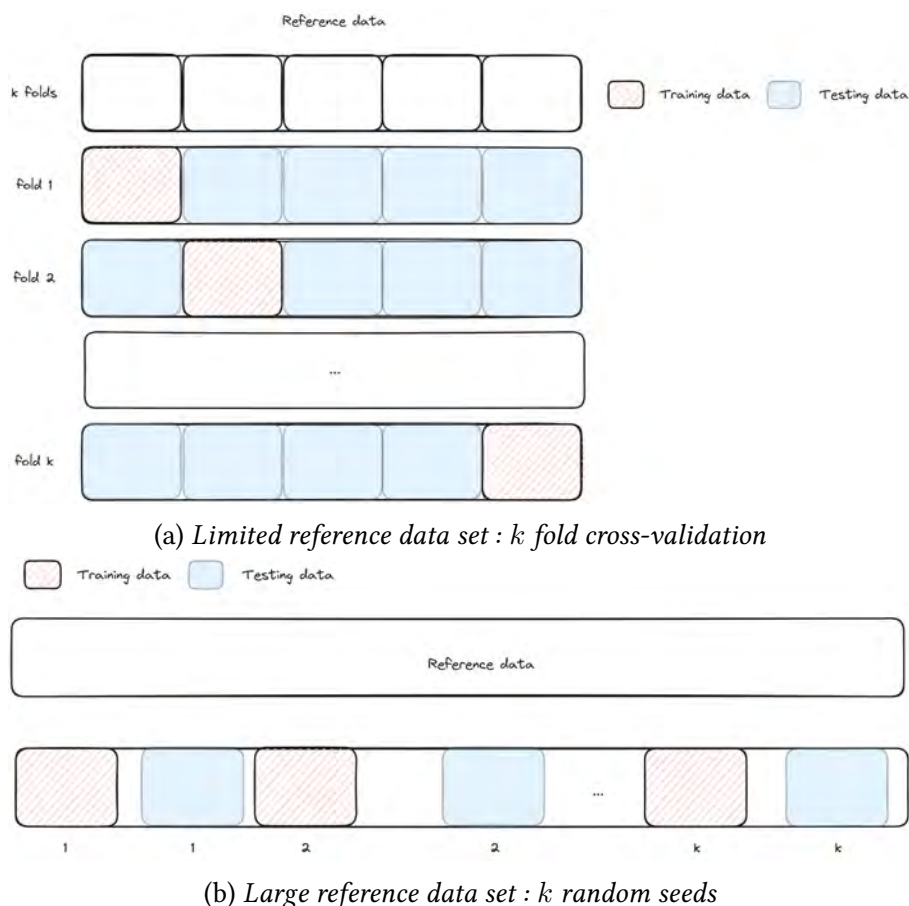


Figure A.3: Comparison between two selection methods for multiple data sets.

A.2.3. Statistical tests

Statistical tests are used to compare different models and assess the significance of observed differences [Foody, 2004]. They allow to answer to the following question: "Are the observed differences statistically significant or could they be due to random chance?". In this manuscript, we decided to use the Wilcoxon test [Conover, 1999] because it allows to know if the mean values of the metrics (i.e. F-score or OA) of two different classifiers differ significantly from each other.

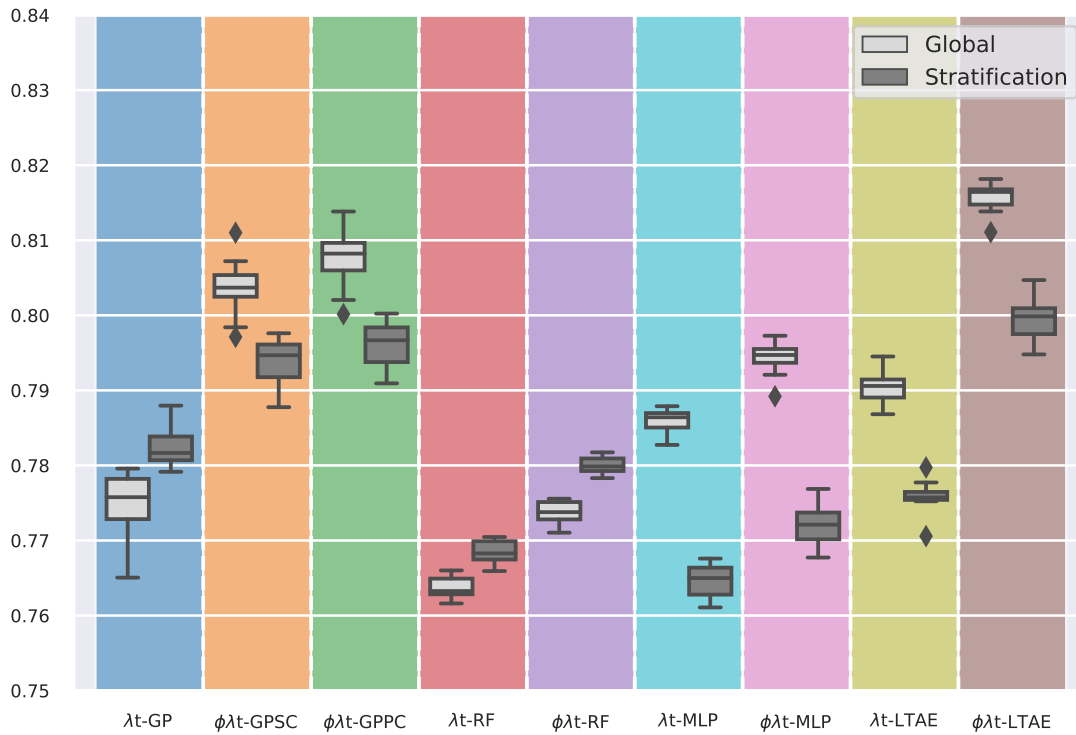
These are additional results for Chapter 6.

B.1. Additional results: Comparison with competitive methods

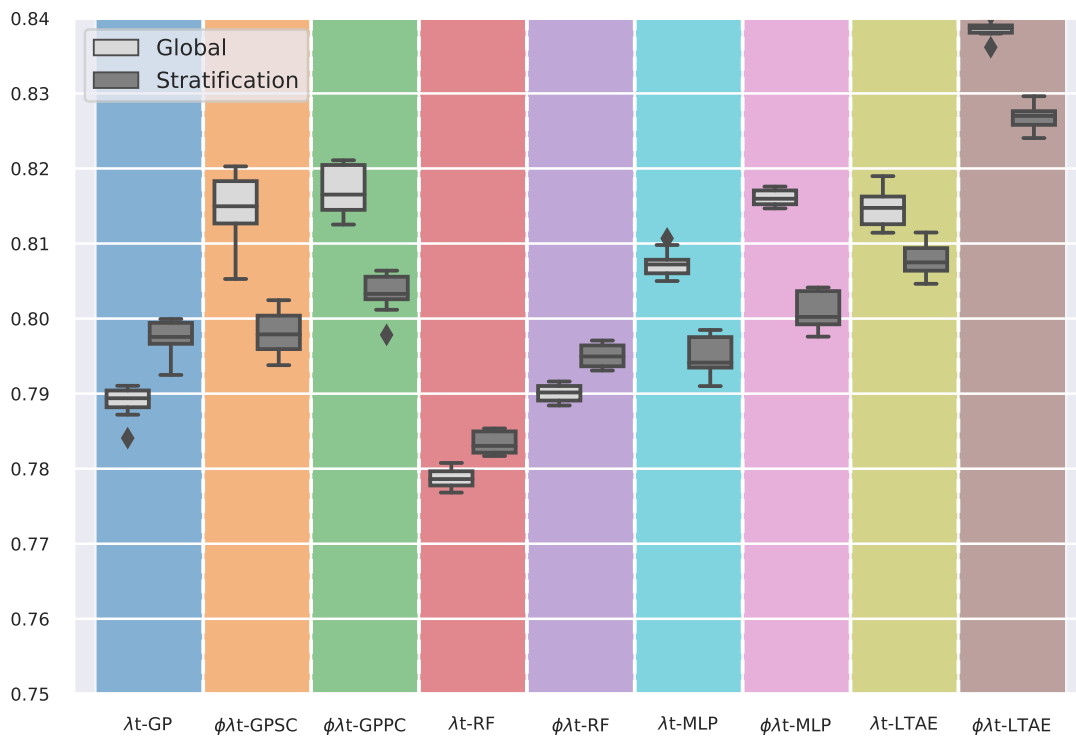
In the following, we present additional results for the Section 6.1.

B.1.1. F-score

The F-score for each model (λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE) is presented in the following. Both data sets DS-A and DS-B are considered for each configuration: *global* and *stratification*.



(a) data set DS-A



(b) data set DS-B

Figure B.1: Boxplots of the F-score for each model: λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE. Both data sets DS-A and DS-B are considered for each configuration: global and stratification.

B.1.2. F-score per class

Two tables are provided in the following. They correspond to the global averaged OA, global averaged F-score and averaged F-score per class (mean $\%$ \pm standard deviation) computed over the 11 runs of each model trained with the *classification* data set. The first line corresponds to models trained with the *training* data set DS-A and the second line corresponds to models trained with the *training* data set DS-B. The studied models are: λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPCC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE. Table B.1 corresponds to the *global* configuration and Table B.2 corresponds to the *stratification* configuration. The nomenclature of the classes is presented in Table 3.3. The darkest grey corresponds to the best F-score per class and the lightest grey corresponds to the third best one.

Table B.1.: Global configuration

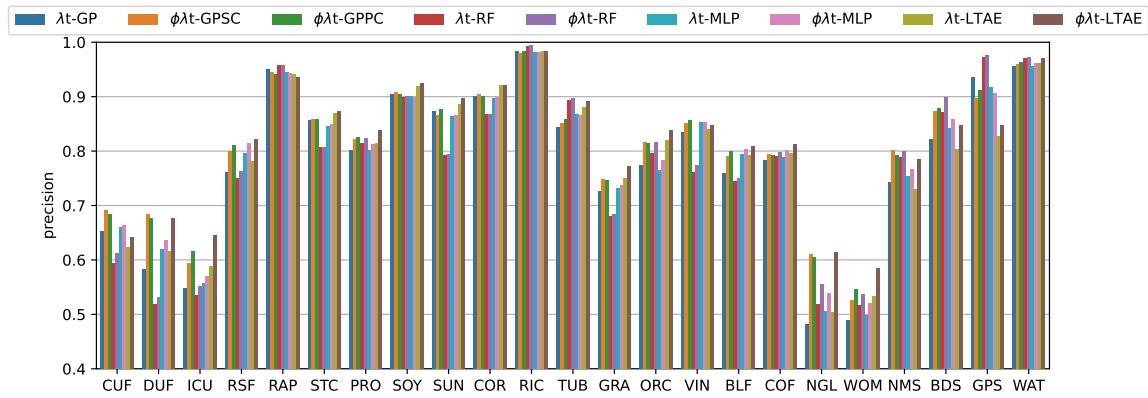
	λt -GP	$\phi \lambda t$ -GPSC	$\phi \lambda t$ -GPPC	λt -RF	$\phi \lambda t$ -RF	λt -MLP	$\phi \lambda t$ -MLP	λt -LTAE	$\phi \lambda t$ -LTAE
CUF	62.4 \pm 3.4	69.1 \pm 0.8	69.0 \pm 0.9	60.8 \pm 0.2	62.5 \pm 0.2	65.4 \pm 0.2	66.6 \pm 0.3	64.7 \pm 0.4	67.9 \pm 0.3
	62.2 \pm 1.7	68.1 \pm 0.9	67.9 \pm 1.1	60.1 \pm 0.2	62.3 \pm 0.2	65.4 \pm 0.3	66.8 \pm 0.2	64.0 \pm 0.3	68.4 \pm 0.3
DUF	61.3 \pm 1.1	69.1 \pm 0.8	68.8 \pm 1.0	57.3 \pm 0.2	58.6 \pm 0.2	63.7 \pm 0.4	65.0 \pm 0.3	62.7 \pm 0.5	66.7 \pm 0.6
	63.9 \pm 0.8	70.3 \pm 0.8	70.1 \pm 0.6	60.5 \pm 0.1	62.2 \pm 0.2	67.3 \pm 0.2	68.8 \pm 0.3	66.3 \pm 0.3	71.8 \pm 0.4
ICU	50.8 \pm 1.5	58.3 \pm 1.5	59.1 \pm 1.2	49.8 \pm 0.2	51.7 \pm 0.3	54.1 \pm 0.5	55.7 \pm 0.3	53.4 \pm 0.7	58.7 \pm 0.4
	55.0 \pm 0.9	62.0 \pm 1.6	62.7 \pm 1.1	53.8 \pm 0.1	56.0 \pm 0.1	60.5 \pm 0.4	62.2 \pm 0.3	59.6 \pm 0.3	65.4 \pm 0.3
RSF	77.5 \pm 0.6	81.2 \pm 0.9	81.6 \pm 0.8	76.2 \pm 0.3	77.1 \pm 0.3	80.3 \pm 0.4	81.2 \pm 0.3	80.3 \pm 0.5	83.4 \pm 0.4
	78.4 \pm 1.1	81.8 \pm 0.7	82.3 \pm 0.5	77.2 \pm 0.2	78.4 \pm 0.2	82.1 \pm 0.3	83.4 \pm 0.3	82.3 \pm 0.2	85.5 \pm 0.2
RAP	94.7 \pm 0.7	94.6 \pm 0.5	94.5 \pm 0.7	94.5 \pm 0.3	94.5 \pm 0.3	94.5 \pm 0.6	94.5 \pm 0.6	94.9 \pm 0.5	94.8 \pm 0.3
	95.2 \pm 0.5	95.3 \pm 0.5	95.4 \pm 0.3	95.2 \pm 0.2	95.3 \pm 0.2	95.3 \pm 0.4	95.4 \pm 0.4	95.8 \pm 0.3	96.0 \pm 0.4
STC	87.5 \pm 0.5	87.7 \pm 0.3	87.9 \pm 0.3	85.6 \pm 0.1	85.7 \pm 0.1	87.5 \pm 0.5	87.8 \pm 0.2	89.2 \pm 0.3	89.6 \pm 0.2
	89.4 \pm 0.4	89.7 \pm 0.4	89.7 \pm 0.6	87.7 \pm 0.1	87.9 \pm 0.1	89.7 \pm 0.3	89.9 \pm 0.2	91.2 \pm 0.2	91.5 \pm 0.2
PRO	74.3 \pm 1.3	76.2 \pm 1.1	76.2 \pm 1.0	70.3 \pm 0.3	71.0 \pm 0.2	75.1 \pm 1.0	75.8 \pm 0.8	76.8 \pm 1.0	78.9 \pm 0.8
	77.0 \pm 0.9	78.5 \pm 1.2	78.3 \pm 1.2	73.7 \pm 0.2	74.6 \pm 0.2	77.8 \pm 0.6	78.9 \pm 0.5	80.2 \pm 0.5	81.9 \pm 0.8
SOY	89.3 \pm 0.5	89.6 \pm 0.4	89.5 \pm 0.4	87.3 \pm 0.2	87.5 \pm 0.3	89.5 \pm 0.4	89.6 \pm 0.6	91.2 \pm 0.6	91.9 \pm 0.4
	90.3 \pm 0.4	90.6 \pm 0.6	90.2 \pm 0.5	88.6 \pm 0.1	88.8 \pm 0.1	90.9 \pm 0.4	91.1 \pm 0.5	92.5 \pm 0.4	92.9 \pm 0.4
SUN	88.2 \pm 0.7	88.2 \pm 0.8	88.6 \pm 0.8	84.4 \pm 0.2	84.6 \pm 0.2	88.3 \pm 0.4	88.3 \pm 0.4	89.7 \pm 0.6	90.5 \pm 0.4
	89.6 \pm 1.1	90.0 \pm 0.8	89.9 \pm 0.9	86.4 \pm 0.1	86.5 \pm 0.2	90.5 \pm 0.2	90.5 \pm 0.3	91.5 \pm 0.3	91.9 \pm 0.3
COR	91.4 \pm 0.5	91.5 \pm 0.5	91.3 \pm 0.5	89.5 \pm 0.2	89.5 \pm 0.2	91.3 \pm 0.3	91.3 \pm 0.2	93.3 \pm 0.1	93.3 \pm 0.3
	92.0 \pm 0.6	92.3 \pm 0.5	92.0 \pm 0.5	90.6 \pm 0.1	90.7 \pm 0.1	92.5 \pm 0.2	92.4 \pm 0.3	94.1 \pm 0.2	94.0 \pm 0.2
RIC	98.2 \pm 0.1	98.2 \pm 0.4	98.3 \pm 0.2	98.4 \pm 0.1	98.5 \pm 0.1	97.9 \pm 0.1	97.9 \pm 0.2	98.2 \pm 0.1	98.4 \pm 0.1
	98.3 \pm 0.2	98.5 \pm 0.2	98.3 \pm 0.7	98.6 \pm 0.0	98.7 \pm 0.0	98.3 \pm 0.1	98.2 \pm 0.3	98.5 \pm 0.2	98.7 \pm 0.1
TUB	83.0 \pm 0.8	83.1 \pm 0.9	83.8 \pm 0.8	79.0 \pm 0.3	79.5 \pm 0.6	83.4 \pm 0.6	83.5 \pm 0.7	86.8 \pm 0.8	87.9 \pm 0.5
	85.0 \pm 1.0	85.0 \pm 0.8	84.6 \pm 1.2	80.6 \pm 0.4	81.0 \pm 0.4	85.5 \pm 0.8	86.0 \pm 0.6	89.0 \pm 0.4	89.7 \pm 0.4
GRA	71.7 \pm 0.8	73.7 \pm 0.8	73.7 \pm 0.5	70.2 \pm 0.2	70.6 \pm 0.2	72.3 \pm 0.3	73.0 \pm 0.5	73.5 \pm 0.2	75.2 \pm 0.2
	73.0 \pm 0.5	74.8 \pm 0.4	75.0 \pm 0.5	71.7 \pm 0.2	72.2 \pm 0.2	73.6 \pm 0.3	74.6 \pm 0.4	75.1 \pm 0.2	76.7 \pm 0.2
ORC	74.0 \pm 0.6	78.3 \pm 0.7	78.4 \pm 1.1	72.8 \pm 0.2	74.1 \pm 0.3	75.0 \pm 0.6	76.2 \pm 0.3	78.1 \pm 0.5	80.2 \pm 0.5
	75.6 \pm 0.7	79.5 \pm 0.5	79.6 \pm 0.7	74.6 \pm 0.2	76.0 \pm 0.1	79.1 \pm 0.5	80.3 \pm 0.3	81.4 \pm 0.4	83.6 \pm 0.3
VIN	86.6 \pm 0.8	88.5 \pm 0.6	88.6 \pm 0.5	81.7 \pm 0.1	82.8 \pm 0.2	87.9 \pm 0.3	88.1 \pm 0.3	87.7 \pm 0.3	88.6 \pm 0.5
	88.1 \pm 0.6	89.6 \pm 0.6	89.6 \pm 0.6	83.7 \pm 0.1	84.8 \pm 0.1	89.7 \pm 0.2	90.1 \pm 0.3	89.7 \pm 0.4	90.9 \pm 0.2
BLF	81.7 \pm 0.8	83.0 \pm 0.8	83.5 \pm 0.9	81.0 \pm 0.2	81.5 \pm 0.2	83.4 \pm 0.3	84.0 \pm 0.3	83.4 \pm 0.2	84.5 \pm 0.3
	83.3 \pm 0.6	85.0 \pm 0.4	85.1 \pm 0.4	82.4 \pm 0.1	83.0 \pm 0.1	85.3 \pm 0.2	85.7 \pm 0.1	85.4 \pm 0.3	87.1 \pm 0.1
COF	81.2 \pm 0.7	82.0 \pm 0.8	82.5 \pm 0.6	81.4 \pm 0.2	82.4 \pm 0.2	82.0 \pm 0.3	82.8 \pm 0.2	83.1 \pm 0.3	84.6 \pm 0.3
	82.3 \pm 0.5	83.0 \pm 0.6	83.5 \pm 0.7	82.6 \pm 0.2	83.6 \pm 0.2	83.8 \pm 0.2	84.4 \pm 0.2	84.9 \pm 0.2	86.3 \pm 0.1
NGL	46.5 \pm 1.4	57.5 \pm 2.2	58.0 \pm 1.3	47.8 \pm 1.4	51.3 \pm 1.2	48.0 \pm 1.6	52.5 \pm 1.2	46.4 \pm 1.8	56.6 \pm 0.9
	48.7 \pm 1.5	58.7 \pm 1.5	59.1 \pm 1.2	50.2 \pm 1.2	54.2 \pm 1.0	52.8 \pm 1.5	55.7 \pm 1.3	51.8 \pm 1.9	60.3 \pm 0.9
WOM	48.4 \pm 3.0	55.8 \pm 1.1	56.5 \pm 2.4	52.9 \pm 0.4	55.1 \pm 0.5	51.6 \pm 0.7	53.4 \pm 0.9	53.0 \pm 1.1	59.5 \pm 0.5
	50.0 \pm 1.9	56.9 \pm 1.6	57.1 \pm 1.5	54.4 \pm 0.5	56.8 \pm 0.5	55.3 \pm 0.7	57.6 \pm 0.7	56.8 \pm 1.0	62.1 \pm 0.4
NMS	72.8 \pm 0.9	78.9 \pm 2.1	78.3 \pm 1.7	73.5 \pm 0.6	75.1 \pm 0.9	73.7 \pm 0.8	75.0 \pm 1.3	74.5 \pm 0.9	79.7 \pm 0.7
	73.7 \pm 0.8	79.1 \pm 1.7	78.7 \pm 1.9	74.2 \pm 0.7	76.2 \pm 0.8	75.8 \pm 0.7	77.8 \pm 0.5	77.1 \pm 0.7	81.4 \pm 0.6
BDS	77.0 \pm 0.7	78.0 \pm 4.5	81.4 \pm 3.4	77.0 \pm 0.5	80.0 \pm 0.7	79.9 \pm 0.7	81.9 \pm 0.6	79.8 \pm 1.2	85.1 \pm 0.7
	78.8 \pm 0.9	78.4 \pm 5.3	81.8 \pm 2.5	77.9 \pm 0.5	80.7 \pm 0.6	81.3 \pm 0.5	82.8 \pm 0.4	83.2 \pm 0.7	86.8 \pm 0.7
GPS	88.6 \pm 1.0	90.4 \pm 1.5	91.9 \pm 0.8	89.9 \pm 1.5	90.4 \pm 1.5	88.2 \pm 1.2	87.7 \pm 1.7	82.3 \pm 2.2	83.6 \pm 2.8
	89.1 \pm 0.9	90.8 \pm 1.4	92.1 \pm 1.3	91.1 \pm 0.9	91.8 \pm 0.9	89.1 \pm 0.7	89.2 \pm 0.9	87.5 \pm 1.2	89.3 \pm 0.9
WAT	95.1 \pm 0.2	95.5 \pm 0.5	95.9 \pm 0.6	95.2 \pm 0.1	95.4 \pm 0.1	94.7 \pm 0.2	95.3 \pm 0.2	95.2 \pm 0.3	96.4 \pm 0.2
	95.5 \pm 0.2	95.8 \pm 0.5	95.8 \pm 0.5	95.3 \pm 0.1	95.5 \pm 0.1	95.0 \pm 0.2	95.3 \pm 0.3	95.8 \pm 0.1	96.8 \pm 0.3
oa	76.6 \pm 0.5	79.4 \pm 0.3	79.8 \pm 0.4	75.3 \pm 0.1	76.3 \pm 0.1	77.7 \pm 0.1	78.6 \pm 0.2	78.5 \pm 0.2	81.0 \pm 0.1
	77.8 \pm 0.2	80.5 \pm 0.4	80.7 \pm 0.3	76.7 \pm 0.1	77.8 \pm 0.1	79.8 \pm 0.2	80.7 \pm 0.1	80.6 \pm 0.2	83.1 \pm 0.1
F-score	77.5 \pm 0.4	80.4 \pm 0.4	80.8 \pm 0.4	76.4 \pm 0.1	77.4 \pm 0.2	78.6 \pm 0.2	79.4 \pm 0.2	79.0 \pm 0.2	81.6 \pm 0.2
	78.9 \pm 0.2	81.5 \pm 0.5	81.7 \pm 0.3	77.9 \pm 0.1	79.0 \pm 0.1	80.7 \pm 0.2	81.6 \pm 0.1	81.5 \pm 0.2	83.9 \pm 0.1

Table B.2.: Stratification configuration

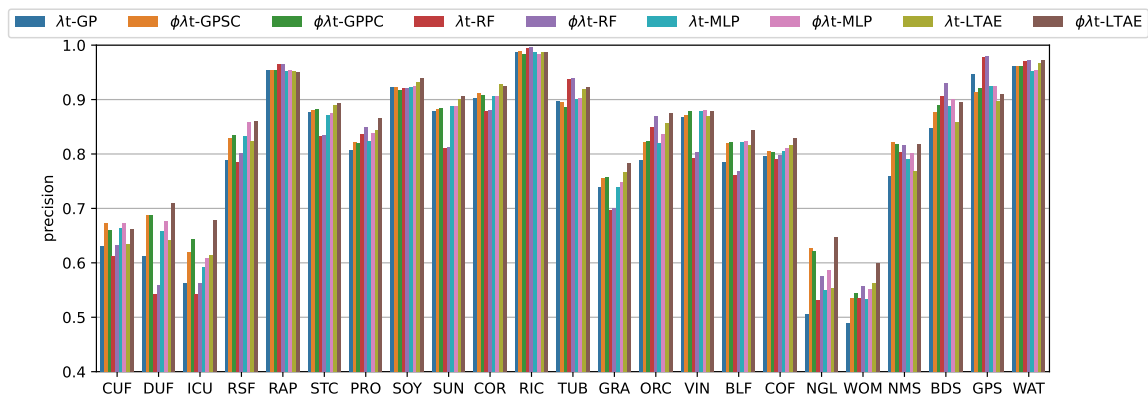
	λt -GP	$\phi\lambda t$ -GPSC	$\phi\lambda t$ -GPPC	λt -RF	$\phi\lambda t$ -RF	λt -MLP	$\phi\lambda t$ -MLP	λt -LTAE	$\phi\lambda t$ -LTAE
CUF	66.8 ± 0.5	71.2 ± 0.3	71.3 ± 0.4	63.6 ± 0.2	66.7 ± 0.2	65.1 ± 0.4	66.2 ± 0.3	64.6 ± 0.3	68.1 ± 0.2
	66.3 ± 0.4	70.5 ± 0.7	70.4 ± 0.5	63.4 ± 0.2	66.6 ± 0.2	67.2 ± 0.1	68.2 ± 0.2	65.4 ± 0.3	69.4 ± 0.3
DUF	63.6 ± 0.9	66.4 ± 0.8	67.0 ± 1.1	58.8 ± 0.4	61.1 ± 0.4	61.1 ± 0.5	61.4 ± 0.8	61.8 ± 0.4	65.6 ± 0.4
	66.7 ± 0.6	66.6 ± 0.8	67.4 ± 0.8	62.0 ± 0.2	64.4 ± 0.2	67.9 ± 0.4	68.8 ± 0.5	66.8 ± 0.4	70.9 ± 0.4
ICU	55.1 ± 0.4	60.0 ± 1.0	60.8 ± 0.4	53.0 ± 0.3	55.8 ± 0.4	53.1 ± 0.3	54.0 ± 0.3	51.4 ± 0.5	56.6 ± 0.3
	59.9 ± 0.4	64.0 ± 1.0	64.5 ± 0.5	57.1 ± 0.2	60.2 ± 0.2	60.5 ± 0.3	61.6 ± 0.4	59.1 ± 0.3	64.4 ± 0.5
RSF	80.3 ± 0.4	82.4 ± 0.8	82.6 ± 0.8	77.9 ± 0.3	79.3 ± 0.4	79.2 ± 0.3	79.8 ± 0.4	78.3 ± 0.4	80.5 ± 0.4
	81.1 ± 0.6	82.4 ± 1.1	82.6 ± 0.9	79.2 ± 0.2	80.4 ± 0.3	81.8 ± 0.4	82.3 ± 0.3	82.2 ± 0.3	84.3 ± 0.3
RAP	93.3 ± 0.7	93.6 ± 0.9	93.6 ± 0.7	94.5 ± 0.3	94.5 ± 0.4	91.8 ± 0.7	92.0 ± 0.7	92.4 ± 0.9	92.3 ± 0.5
	94.6 ± 0.5	94.6 ± 0.7	94.8 ± 0.5	94.9 ± 0.3	94.9 ± 0.3	93.6 ± 0.4	93.6 ± 0.4	94.6 ± 0.4	94.7 ± 0.5
STC	86.3 ± 0.4	86.2 ± 0.5	86.0 ± 0.5	84.1 ± 0.2	84.3 ± 0.2	84.2 ± 0.4	84.5 ± 0.5	87.1 ± 0.3	87.4 ± 0.4
	88.3 ± 0.4	88.1 ± 0.4	88.2 ± 0.5	86.7 ± 0.2	86.8 ± 0.3	87.8 ± 0.4	87.8 ± 0.4	90.1 ± 0.2	90.2 ± 0.3
PRO	71.6 ± 0.9	73.8 ± 0.6	72.7 ± 0.7	66.6 ± 0.5	67.5 ± 0.4	68.9 ± 0.7	69.4 ± 0.5	71.2 ± 1.3	74.0 ± 0.8
	73.8 ± 0.5	75.4 ± 0.8	75.2 ± 0.6	70.2 ± 0.3	70.8 ± 0.4	73.2 ± 0.8	73.7 ± 1.1	77.0 ± 0.7	78.7 ± 0.5
SOY	86.6 ± 0.6	86.6 ± 0.7	86.6 ± 0.8	86.0 ± 0.3	86.2 ± 0.3	84.6 ± 0.5	84.6 ± 0.9	87.3 ± 0.8	88.5 ± 0.7
	87.8 ± 0.5	87.5 ± 0.7	87.8 ± 0.5	87.5 ± 0.3	87.8 ± 0.3	87.0 ± 0.4	87.2 ± 0.5	89.7 ± 0.5	90.4 ± 0.7
SUN	85.3 ± 0.5	85.6 ± 0.4	85.8 ± 0.4	83.3 ± 0.3	83.6 ± 0.3	83.2 ± 0.5	83.5 ± 0.4	86.3 ± 0.7	87.1 ± 0.5
	87.4 ± 0.3	87.5 ± 0.4	87.7 ± 0.6	85.6 ± 0.3	85.9 ± 0.2	86.8 ± 0.3	86.8 ± 0.3	89.3 ± 0.3	89.7 ± 0.4
COR	89.0 ± 0.2	88.9 ± 0.4	89.2 ± 0.4	87.4 ± 0.2	87.4 ± 0.2	87.3 ± 0.5	87.2 ± 0.6	90.5 ± 0.7	90.9 ± 0.7
	90.3 ± 0.4	90.3 ± 0.4	90.5 ± 0.3	89.2 ± 0.2	89.3 ± 0.1	89.4 ± 0.5	89.6 ± 0.3	92.4 ± 0.4	92.7 ± 0.3
RIC	98.2 ± 0.3	98.1 ± 0.4	98.2 ± 0.2	97.8 ± 0.1	97.9 ± 0.1	97.8 ± 0.2	97.8 ± 0.3	97.6 ± 0.2	98.2 ± 0.2
	98.3 ± 0.2	98.1 ± 0.4	98.4 ± 0.2	97.9 ± 0.2	98.1 ± 0.2	98.0 ± 0.5	98.1 ± 0.5	98.3 ± 0.1	98.5 ± 0.2
TUB	79.0 ± 1.3	79.3 ± 1.0	80.5 ± 0.9	75.3 ± 0.7	75.8 ± 0.8	74.3 ± 2.0	75.3 ± 1.5	80.7 ± 1.3	82.2 ± 1.3
	79.9 ± 1.6	80.5 ± 1.6	81.2 ± 1.4	76.6 ± 1.0	77.2 ± 0.8	78.8 ± 1.5	79.2 ± 1.6	84.2 ± 1.0	85.1 ± 1.1
GRA	72.3 ± 0.5	73.8 ± 0.7	73.6 ± 0.9	70.6 ± 0.2	71.0 ± 0.2	71.0 ± 0.5	71.4 ± 0.4	72.2 ± 0.2	73.6 ± 0.2
	73.8 ± 0.5	74.9 ± 0.8	75.2 ± 0.6	72.2 ± 0.1	72.8 ± 0.1	73.3 ± 0.4	73.6 ± 0.4	74.8 ± 0.2	75.8 ± 0.3
ORC	75.2 ± 0.5	77.4 ± 0.7	77.1 ± 1.1	73.8 ± 0.1	75.3 ± 0.2	72.7 ± 0.7	73.9 ± 0.5	74.9 ± 0.6	77.6 ± 0.4
	78.4 ± 0.4	79.8 ± 0.8	79.7 ± 1.1	75.1 ± 0.1	76.7 ± 0.2	77.4 ± 0.4	78.3 ± 0.5	80.2 ± 0.3	81.6 ± 0.3
VIN	86.9 ± 0.3	88.3 ± 0.6	88.5 ± 0.6	83.5 ± 0.2	84.6 ± 0.3	83.9 ± 0.9	85.1 ± 0.8	85.4 ± 0.3	87.1 ± 0.3
	88.1 ± 0.8	89.3 ± 0.7	89.7 ± 0.4	84.6 ± 0.3	85.9 ± 0.2	88.1 ± 0.6	88.4 ± 0.6	88.7 ± 0.3	89.9 ± 0.2
BLF	82.2 ± 0.4	83.3 ± 0.6	83.4 ± 0.7	81.5 ± 0.2	82.2 ± 0.2	80.9 ± 0.3	81.6 ± 0.3	81.1 ± 0.3	82.2 ± 0.7
	83.9 ± 0.3	84.5 ± 0.5	84.9 ± 0.5	82.9 ± 0.1	83.6 ± 0.1	84.0 ± 0.3	84.6 ± 0.2	84.4 ± 0.3	85.6 ± 0.3
COF	80.9 ± 0.5	80.3 ± 0.9	80.7 ± 0.7	81.0 ± 0.4	82.0 ± 0.4	79.7 ± 0.6	80.4 ± 0.5	81.6 ± 0.4	82.7 ± 0.6
	82.3 ± 0.3	80.6 ± 1.0	81.2 ± 0.9	82.4 ± 0.2	83.4 ± 0.3	82.4 ± 0.5	82.9 ± 0.3	84.0 ± 0.3	84.6 ± 0.5
NGL	49.6 ± 2.6	56.0 ± 2.1	54.4 ± 1.5	50.9 ± 1.2	54.0 ± 1.3	48.8 ± 2.5	51.4 ± 2.2	48.0 ± 1.6	56.7 ± 2.0
	52.7 ± 1.9	54.8 ± 1.3	55.2 ± 1.3	53.0 ± 1.4	56.3 ± 1.4	53.3 ± 2.0	55.4 ± 2.1	52.6 ± 1.6	59.4 ± 1.4
WOM	55.4 ± 1.0	57.5 ± 0.9	58.4 ± 0.9	55.2 ± 0.6	57.1 ± 0.5	52.7 ± 1.1	53.7 ± 1.2	54.4 ± 1.0	58.8 ± 0.7
	56.1 ± 1.0	57.8 ± 1.0	58.9 ± 0.5	56.2 ± 0.6	58.2 ± 0.5	56.1 ± 0.9	57.0 ± 0.8	57.4 ± 1.1	61.3 ± 0.5
NMS	78.0 ± 1.4	79.8 ± 1.2	80.5 ± 2.0	78.6 ± 1.3	79.9 ± 1.5	77.6 ± 1.3	79.1 ± 1.0	78.8 ± 0.7	81.3 ± 0.9
	78.6 ± 0.9	79.5 ± 1.3	80.2 ± 1.7	79.1 ± 0.9	80.1 ± 1.1	78.1 ± 0.9	79.2 ± 0.9	80.0 ± 0.6	82.2 ± 0.3
BDS	77.5 ± 1.7	69.5 ± 4.8	72.2 ± 3.6	75.8 ± 1.1	79.1 ± 1.5	76.1 ± 1.6	78.0 ± 1.4	74.8 ± 1.2	81.5 ± 0.7
	78.7 ± 1.2	63.5 ± 4.6	68.0 ± 4.6	77.1 ± 0.8	79.6 ± 1.0	78.9 ± 0.9	79.7 ± 1.5	80.1 ± 0.8	83.5 ± 0.8
GPS	91.0 ± 2.9	93.3 ± 0.5	93.7 ± 1.3	92.2 ± 1.8	92.8 ± 1.6	89.9 ± 3.4	90.9 ± 1.2	89.4 ± 1.0	90.6 ± 1.5
	91.3 ± 2.1	92.3 ± 1.4	92.0 ± 2.2	93.0 ± 1.4	93.5 ± 1.5	90.5 ± 2.2	91.7 ± 1.5	90.8 ± 1.3	92.4 ± 1.7
WAT	95.4 ± 0.5	94.6 ± 1.1	94.7 ± 0.7	95.9 ± 0.3	96.0 ± 0.4	94.5 ± 0.4	94.5 ± 0.6	94.6 ± 0.4	95.4 ± 0.5
	95.6 ± 0.5	93.2 ± 1.0	94.1 ± 0.8	96.1 ± 0.2	96.1 ± 0.3	94.9 ± 0.5	94.8 ± 0.5	95.9 ± 0.4	96.0 ± 0.4
oa	77.2 ± 0.2	78.4 ± 0.2	78.6 ± 0.2	75.7 ± 0.2	76.8 ± 0.1	75.4 ± 0.2	76.1 ± 0.3	76.8 ± 0.2	79.1 ± 0.3
	78.7 ± 0.2	78.9 ± 0.2	79.4 ± 0.2	77.2 ± 0.1	78.4 ± 0.1	78.5 ± 0.2	79.1 ± 0.2	79.9 ± 0.2	81.8 ± 0.2
F-score	78.2 ± 0.3	79.4 ± 0.3	79.6 ± 0.3	76.8 ± 0.2	78.0 ± 0.1	76.5 ± 0.2	77.2 ± 0.3	77.6 ± 0.2	80.0 ± 0.3
	79.8 ± 0.2	79.8 ± 0.3	80.3 ± 0.3	78.3 ± 0.1	79.5 ± 0.1	79.5 ± 0.2	80.1 ± 0.3	80.8 ± 0.2	82.7 ± 0.2

B.1.3. Precision and recall per class

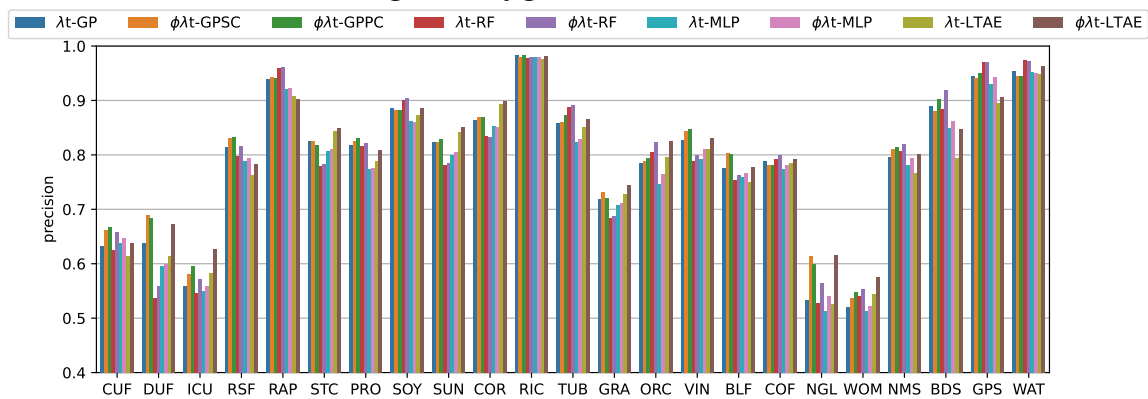
Two figures are provided in the following. They correspond to the barplots of the precision and recall per class, respectively, for each model: λt -GP, $\phi\lambda t$ -GPSC, $\phi\lambda t$ -GPPC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP, λt -LTAE and $\phi\lambda t$ -LTAE. Both data sets DS-A and DS-B are considered for each configuration: *global* and *stratification*.



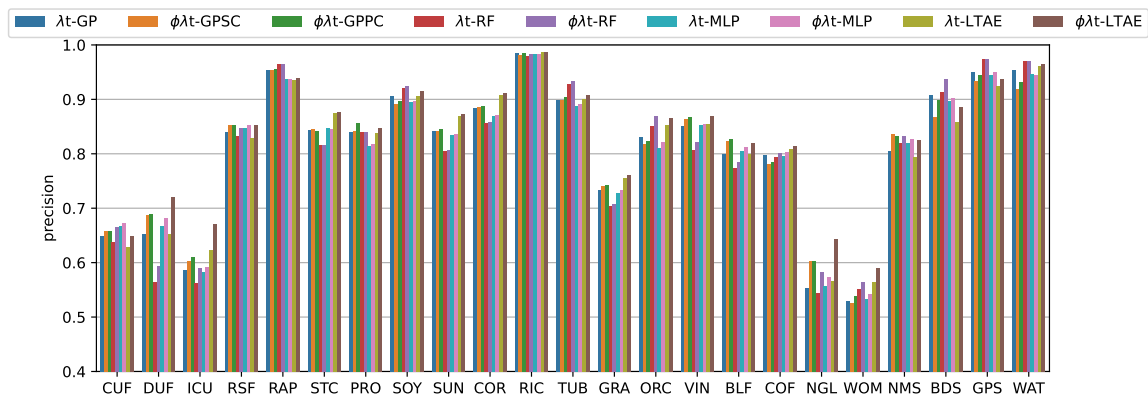
(a) global configuration (data set DS-A)



(b) global configuration (data set DS-B)

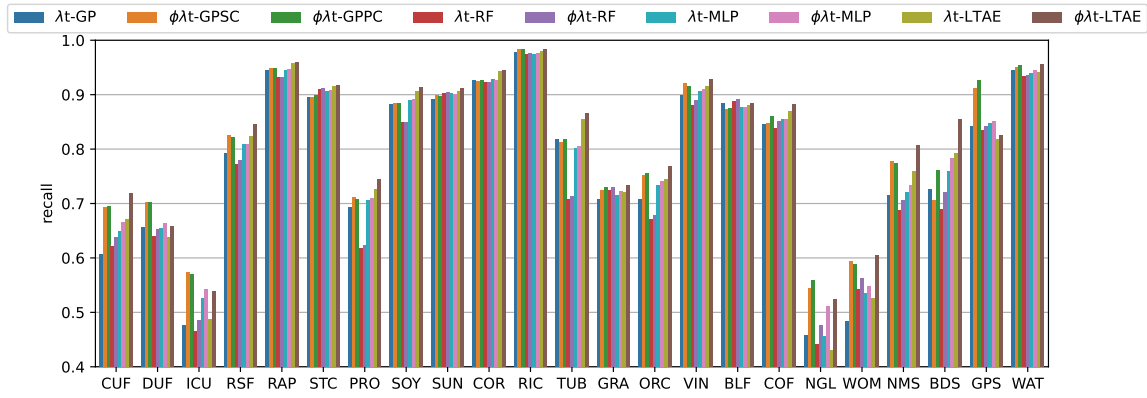


(c) stratification configuration (data set DS-A)

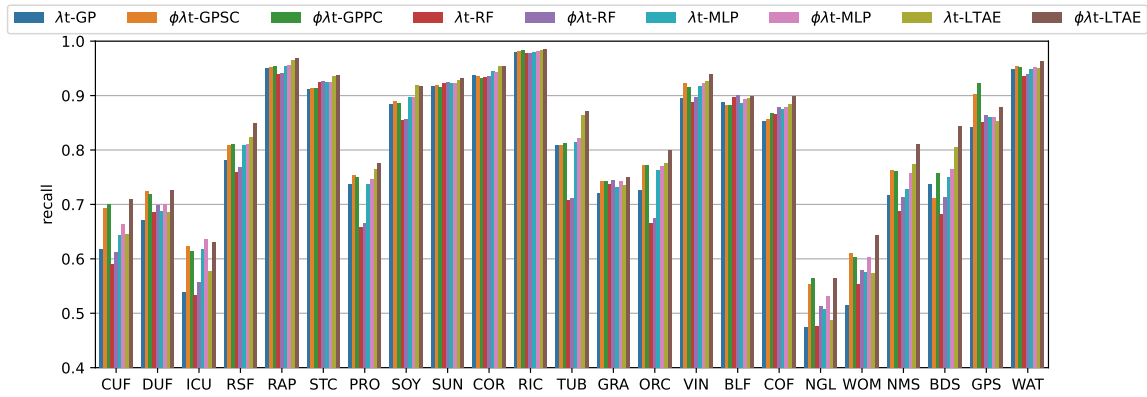


(d) stratification configuration (data set DS-B)

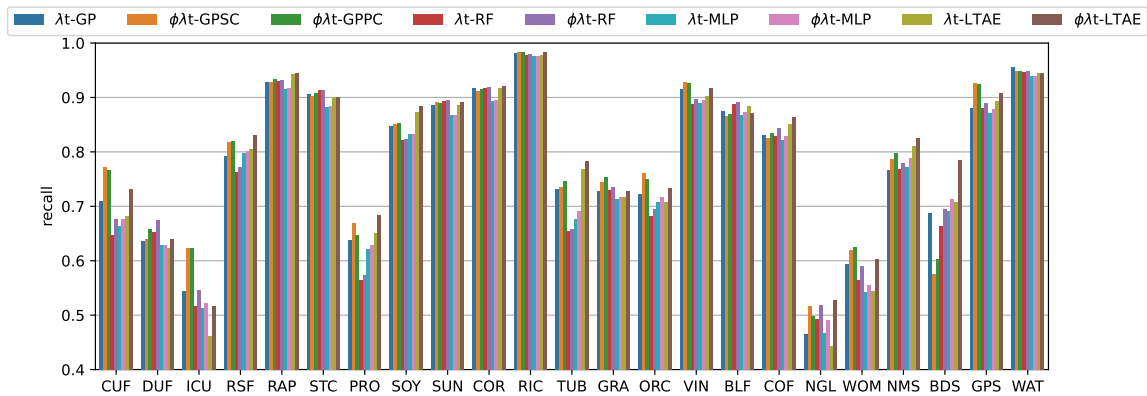
Figure B.2: Precision



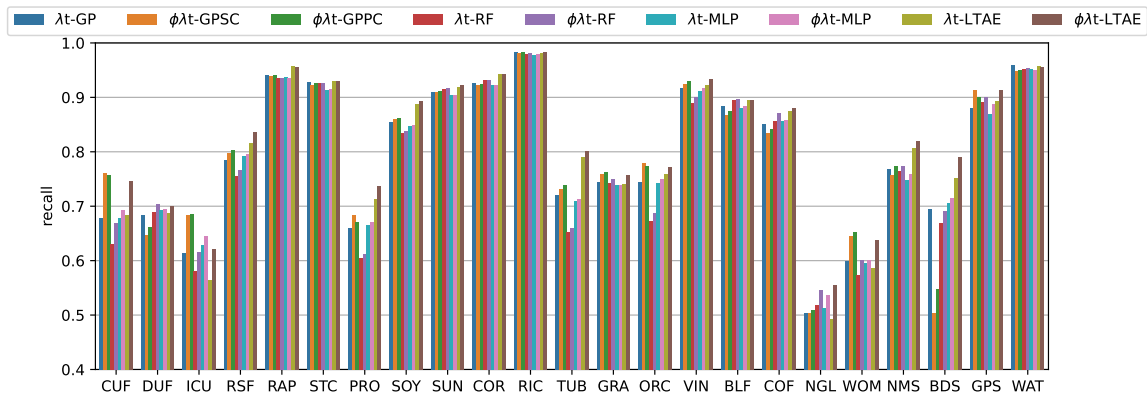
(a) global configuration (data set DS-A)



(b) global configuration (data set DS-B)



(c) stratification configuration (data set DS-A)

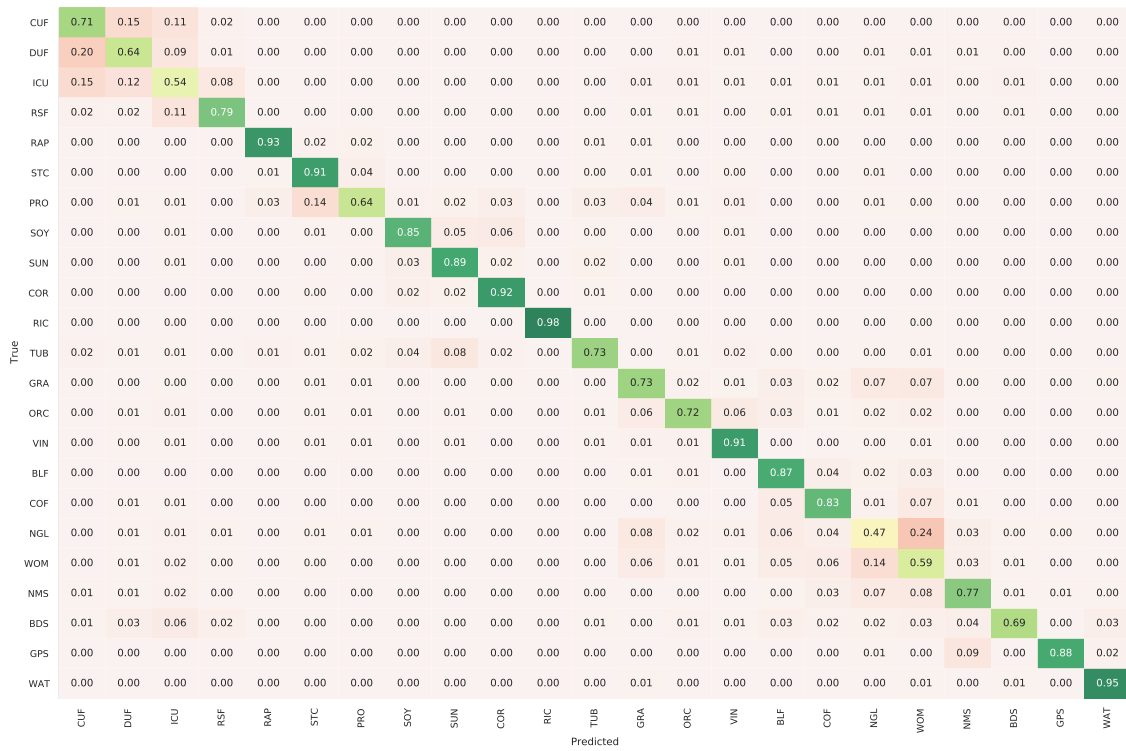


(d) stratification configuration (data set DS-B)

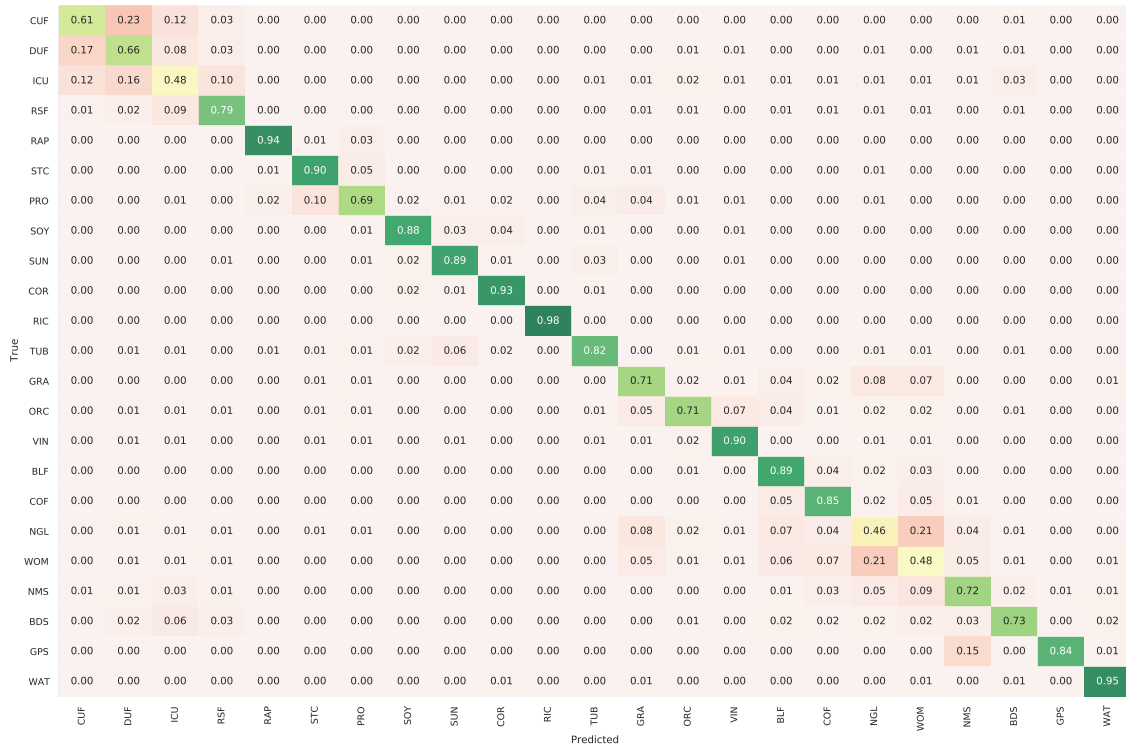
Figure B.3: Recall

B.1.4. Confusion matrices

Normalized confusion matrices for λt -GP, $\phi\lambda t$ -GPSC, λt -RF, $\phi\lambda t$ -RF, λt -MLP, $\phi\lambda t$ -MLP and λt -LTAE are represented in the following. Regarding the models $\phi\lambda t$ -GPSC and $\phi\lambda t$ -LTAE, they are represented in Figure 6.4. Only the data set DS-A is considered for each configuration: *global* and *stratification*

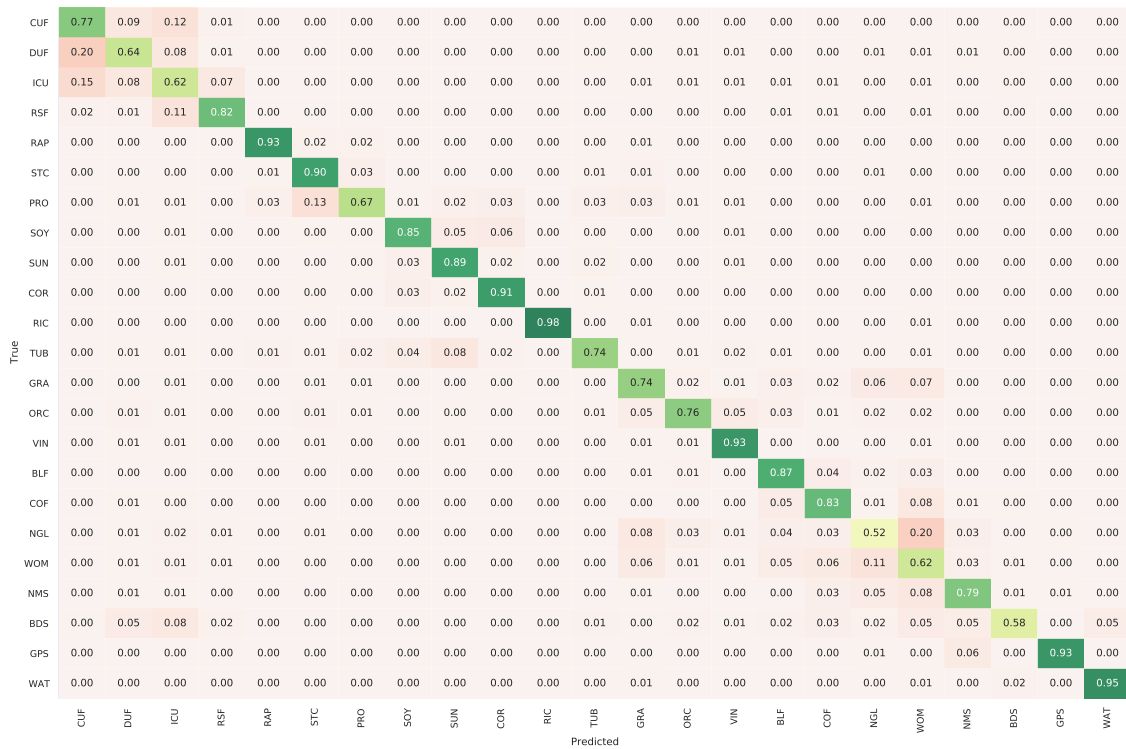


(1) stratification configuration

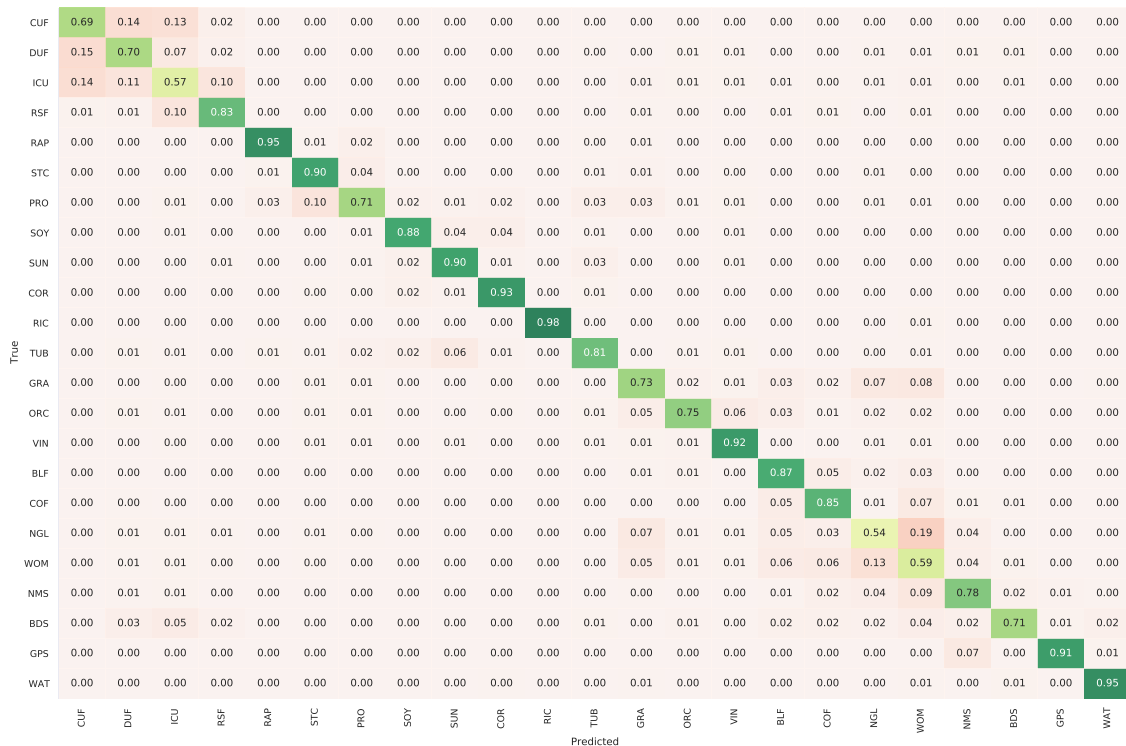


(2) global configuration

(a) λt -GP model

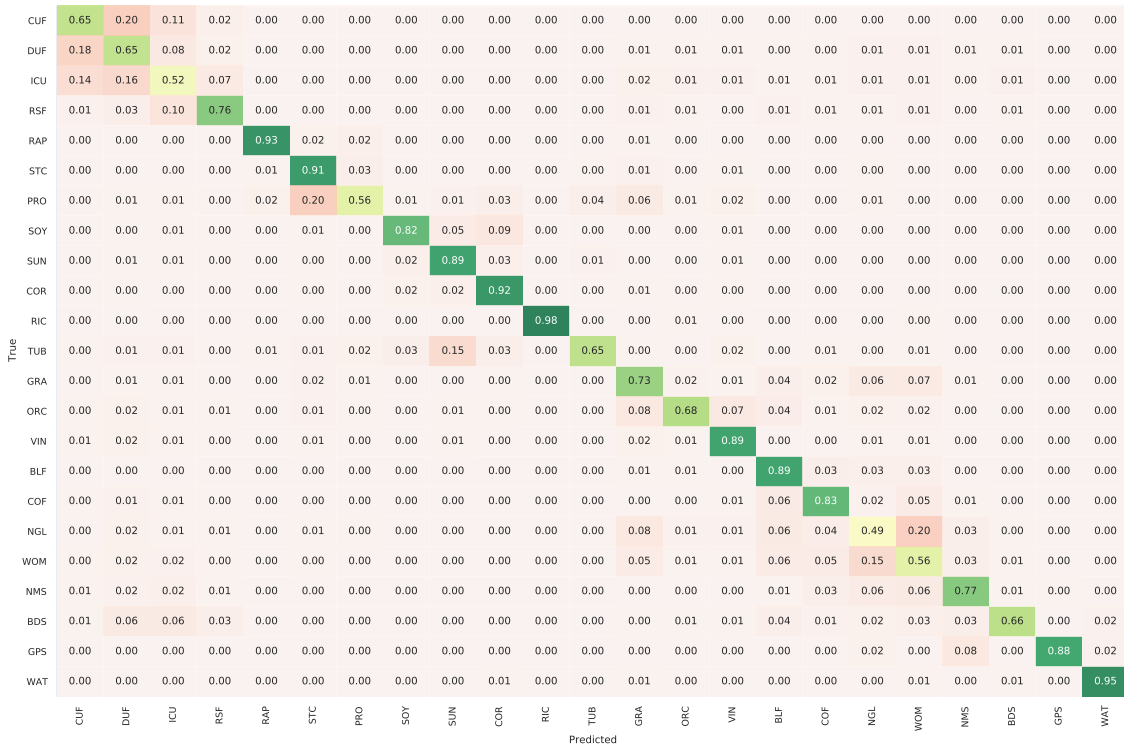


(1) stratification configuration



(2) global configuration

(b) $\phi\lambda t$ -GPSC model



(1) stratification configuration



(2) global configuration

(c) λt -RF model

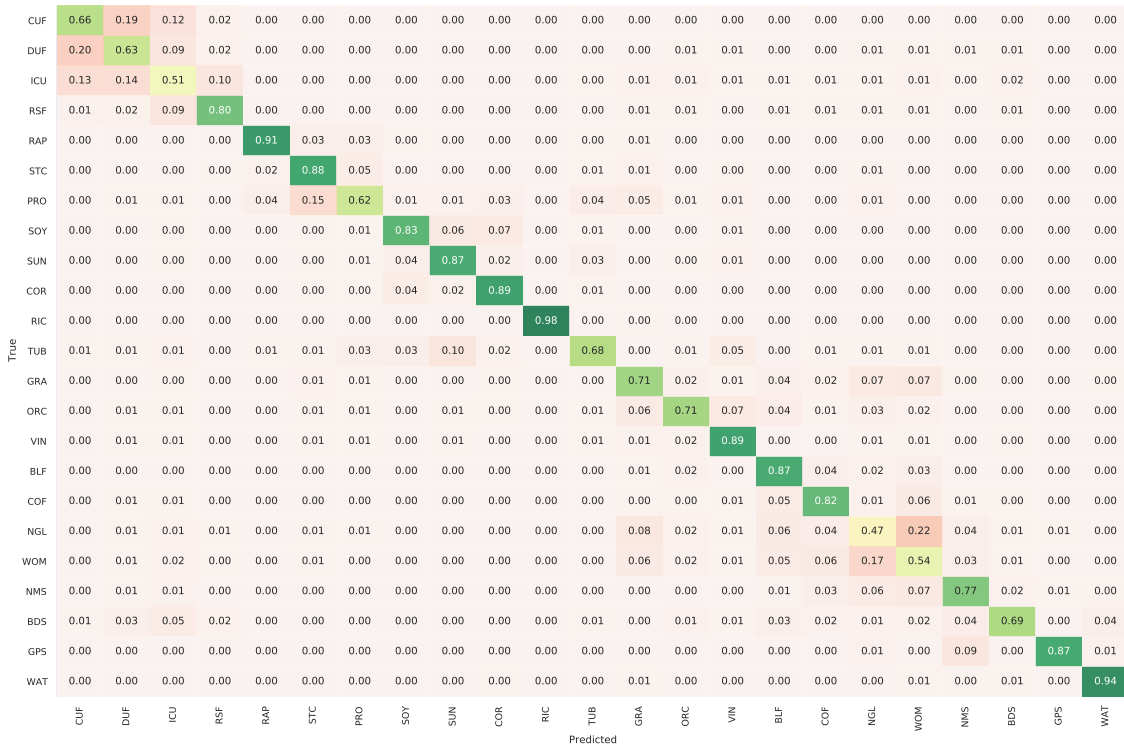
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT
CUF	0.68	0.18	0.11	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUF	0.16	0.67	0.08	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00
ICU	0.13	0.15	0.55	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00	0.00
RSF	0.01	0.03	0.11	0.77	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.93	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.01	0.91	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.01	0.01	0.00	0.02	0.20	0.57	0.01	0.01	0.03	0.00	0.04	0.06	0.01	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.82	0.05	0.09	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.02	0.89	0.03	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.92	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
TUB	0.00	0.01	0.01	0.00	0.01	0.01	0.02	0.03	0.15	0.04	0.00	0.66	0.00	0.00	0.02	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00
GRA	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.74	0.02	0.01	0.03	0.02	0.06	0.07	0.00	0.00	0.00	0.00
ORC	0.00	0.02	0.01	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.08	0.69	0.07	0.04	0.01	0.02	0.02	0.00	0.00	0.00	0.00
VIN	0.01	0.02	0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.01	0.90	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.89	0.03	0.02	0.03	0.00	0.00	0.00	0.00
COF	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.84	0.01	0.05	0.01	0.00	0.00	0.00
NGL	0.00	0.02	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.01	0.01	0.06	0.04	0.52	0.19	0.03	0.00	0.00	0.00
WOM	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.01	0.01	0.05	0.05	0.14	0.59	0.03	0.01	0.00	0.00
NMS	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.05	0.07	0.78	0.01	0.00	0.00
BDS	0.01	0.06	0.05	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.04	0.01	0.02	0.03	0.03	0.69	0.00	0.02
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.07	0.00	0.89	0.02
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.95
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT

(1) stratification configuration

	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT
CUF	0.64	0.22	0.10	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUF	0.18	0.65	0.07	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00
ICU	0.14	0.18	0.49	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00	0.00
RSF	0.01	0.03	0.09	0.78	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.93	0.01	0.03	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.00	0.91	0.03	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.01	0.01	0.00	0.02	0.16	0.62	0.02	0.01	0.02	0.00	0.03	0.06	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.85	0.05	0.06	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.02	0.91	0.02	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TUB	0.00	0.01	0.01	0.00	0.01	0.01	0.01	0.03	0.14	0.03	0.00	0.71	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GRA	0.00	0.01	0.00	0.00	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.73	0.01	0.01	0.04	0.02	0.06	0.07	0.00	0.00	0.00	0.01
ORC	0.00	0.03	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.07	0.68	0.08	0.05	0.01	0.02	0.02	0.00	0.00	0.00	0.00
VIN	0.01	0.02	0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.01	0.89	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.89	0.03	0.02	0.03	0.00	0.00	0.00	0.00
COF	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.05	0.85	0.01	0.05	0.00	0.00	0.00	0.00
NGL	0.00	0.02	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.01	0.02	0.07	0.04	0.48	0.19	0.03	0.00	0.00	0.00
WOM	0.01	0.02	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.01	0.01	0.06	0.06	0.15	0.56	0.03	0.01	0.00	0.00
NMS	0.03	0.02	0.05	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.04	0.08	0.71	0.01	0.00	0.00
BDS	0.01	0.04	0.06	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.03	0.01	0.02	0.02	0.02	0.72	0.00	0.01
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.00	0.84	0.01
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.94
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT

(2) global configuration

(d) $\phi\lambda t$ -RF model



(1) stratification configuration



(2) global configuration

(e) λt -MLP model

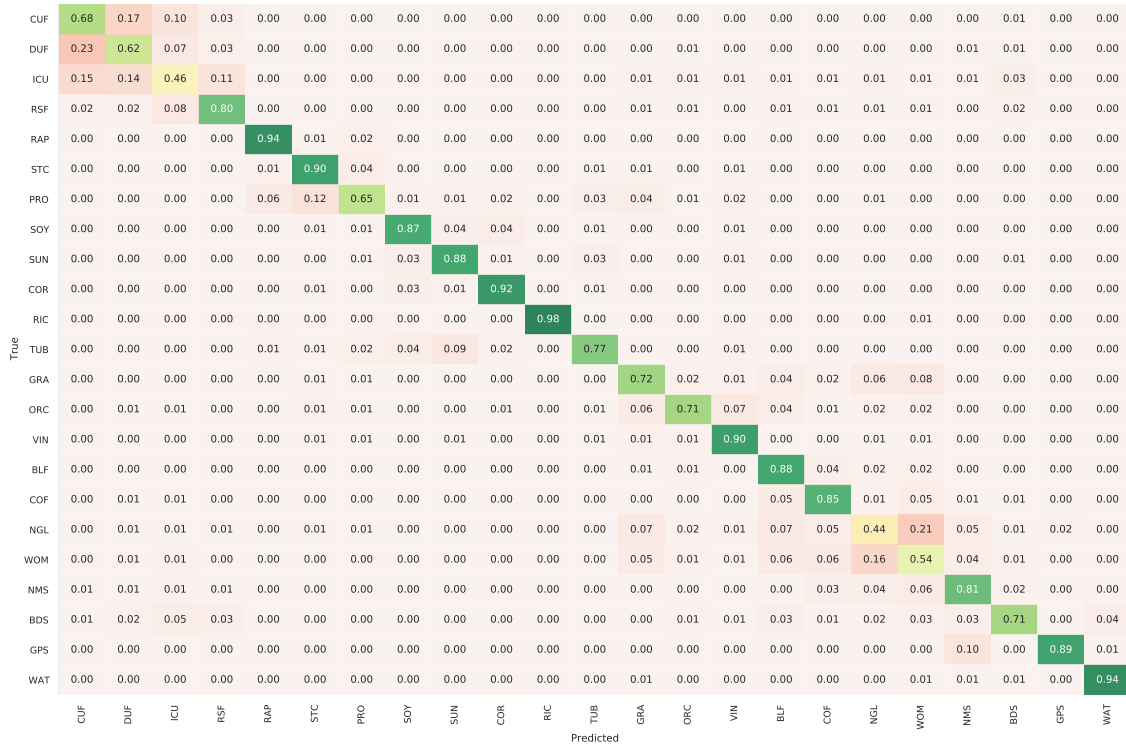
True	Predicted																					
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS
CUF	0.68	0.18	0.11	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUF	0.19	0.63	0.10	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.00
ICU	0.14	0.14	0.52	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.02	0.00	0.00
RSF	0.01	0.02	0.09	0.80	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01	0.01	0.00	0.01	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.92	0.03	0.03	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.02	0.88	0.05	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.01	0.01	0.00	0.03	0.15	0.63	0.01	0.01	0.03	0.00	0.04	0.04	0.01	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.83	0.06	0.07	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.04	0.87	0.02	0.00	0.03	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.02	0.89	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TUB	0.01	0.02	0.01	0.00	0.01	0.01	0.03	0.04	0.10	0.02	0.00	0.69	0.00	0.01	0.03	0.00	0.01	0.01	0.01	0.00	0.00	0.00
GRA	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.72	0.02	0.01	0.03	0.02	0.07	0.07	0.00	0.00	0.00
ORC	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.06	0.72	0.06	0.04	0.01	0.02	0.02	0.00	0.00	0.00
VIN	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.01	0.02	0.90	0.00	0.00	0.01	0.01	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.87	0.04	0.02	0.03	0.00	0.00	0.00
COF	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.05	0.83	0.01	0.06	0.01	0.00	0.00
NGL	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.08	0.02	0.01	0.06	0.04	0.49	0.22	0.03	0.01	0.00
WOM	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.02	0.01	0.05	0.06	0.16	0.55	0.03	0.01	0.00
NMS	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.05	0.07	0.79	0.02	0.01	0.00
BDS	0.00	0.03	0.04	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.03	0.02	0.01	0.02	0.04	0.71	0.00
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.09	0.00	0.88	0.02
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.94

(1) stratification configuration

True	Predicted																					
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS
CUF	0.67	0.19	0.11	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUF	0.18	0.66	0.09	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.00
ICU	0.13	0.13	0.54	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.00
RSF	0.01	0.02	0.10	0.81	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01	0.01	0.00	0.01	0.00
RAP	0.00	0.00	0.00	0.00	0.95	0.01	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.01	0.91	0.04	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.00	0.01	0.00	0.03	0.11	0.71	0.01	0.02	0.02	0.00	0.03	0.03	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.89	0.03	0.04	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.90	0.01	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.93	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TUB	0.00	0.00	0.01	0.00	0.01	0.01	0.02	0.03	0.07	0.01	0.00	0.81	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GRA	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.72	0.02	0.01	0.03	0.02	0.07	0.07	0.00	0.00	0.01
ORC	0.00	0.01	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.01	0.00	0.01	0.05	0.74	0.06	0.03	0.01	0.02	0.02	0.00	0.01	0.00
VIN	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.01	0.02	0.91	0.00	0.00	0.01	0.01	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.88	0.03	0.02	0.03	0.00	0.00	0.00
COF	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.85	0.01	0.05	0.01	0.00	0.00	0.00
NGL	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.08	0.02	0.01	0.05	0.04	0.51	0.19	0.04	0.01	0.00
WOM	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.01	0.01	0.05	0.06	0.18	0.55	0.04	0.01	0.00
NMS	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.05	0.11	0.73	0.02	0.01	0.01
BDS	0.00	0.02	0.04	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.03	0.03	0.78	0.00
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.00	0.85	0.01
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.94

(2) global configuration

(f) $\phi\lambda t$ -MLP model



(1) stratification configuration



(2) global configuration

(g) λt -LTAE model

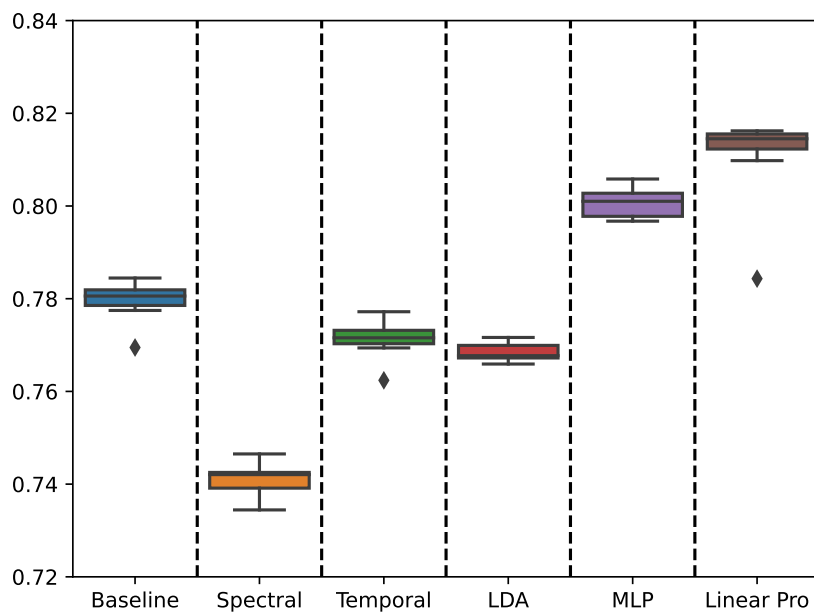
Figure B.4: Normalized confusion matrices for each model.

B.2. Additional results: Feature extraction

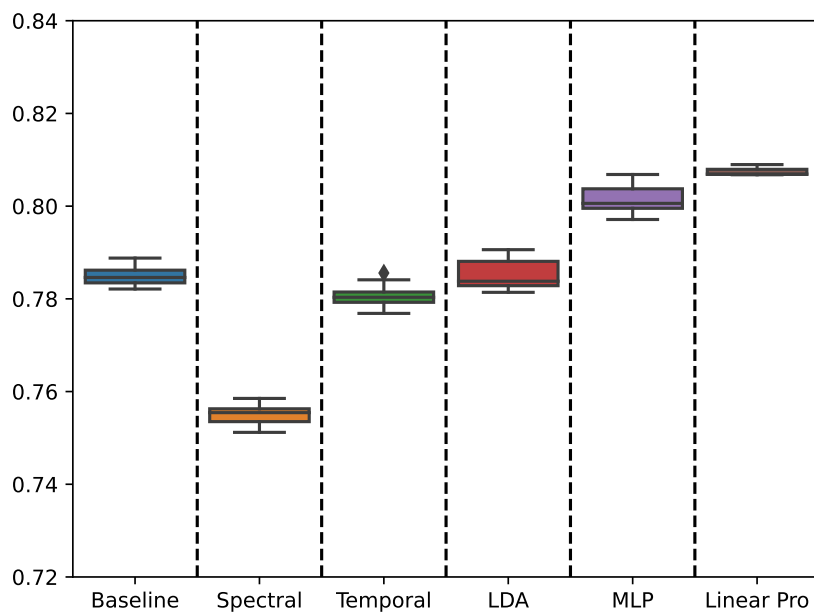
In the following, we present additional results for the Section 6.5.

B.2.1. F-score

Comparison of the Fscore for the different feature extraction methods in both configurations: *global* and *stratification*. The nomenclature of the methods is described in Table 5.8.



(a) Global configuration



(b) Stratification configuration

Figure B.5: Comparison of the Fscore for the different feature extraction methods.

These are additional results for Chapter 9.

C.1. Additional results: Comparison with competitive methods

In the following, we present additional results for the Section 9.1.

C.1.1. Confusion matrices

Normalized confusion matrices for the *Gapfilled-SVGP*, *EmTAN-MLP* and *EmTAN-LTAE* models are represented in the following. Concerning the *EmTAN-SVGP* and *raw-LTAE* models, they are represented in Figure 9.3.

CUF	0.67	0.15	0.10	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.02	0.01	0.00	0.00	
DUF	0.29	0.45	0.07	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.03	0.00	0.00	0.03	0.01	0.02	0.02	0.00	0.00
ICU	0.25	0.10	0.34	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.05	0.02	0.00	0.00	0.04	0.02	0.03	0.06	0.00	0.00
RSF	0.03	0.02	0.10	0.66	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.04	0.01	0.01	0.00	0.02	0.02	0.02	0.04	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.79	0.09	0.08	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.01	0.01	0.85	0.05	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.03	0.01	0.00	0.00	0.00	0.00
PRO	0.00	0.00	0.00	0.01	0.02	0.13	0.59	0.02	0.02	0.01	0.00	0.03	0.06	0.01	0.02	0.00	0.00	0.04	0.01	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.83	0.07	0.04	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.86	0.01	0.00	0.03	0.00	0.01	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.02	0.85	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TUB	0.01	0.01	0.00	0.00	0.01	0.01	0.03	0.03	0.09	0.01	0.00	0.73	0.00	0.02	0.02	0.00	0.00	0.01	0.01	0.01	0.01	0.00	0.00
GRA	0.00	0.00	0.00	0.01	0.02	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.58	0.05	0.01	0.03	0.02	0.15	0.06	0.00	0.01	0.00	0.01
ORC	0.00	0.01	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.05	0.72	0.07	0.02	0.01	0.05	0.02	0.00	0.01	0.00	0.00
VIN	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.04	0.85	0.00	0.00	0.03	0.02	0.01	0.01	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.10	0.00	0.73	0.04	0.04	0.06	0.00	0.01	0.00	0.00
COF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.01	0.07	0.66	0.02	0.15	0.01	0.02	0.00	0.00
NGL	0.00	0.02	0.01	0.01	0.13	0.02	0.01	0.01	0.00	0.01	0.00	0.00	0.06	0.06	0.03	0.05	0.04	0.32	0.18	0.03	0.01	0.00	0.00
WOM	0.00	0.01	0.01	0.01	0.05	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.04	0.05	0.03	0.04	0.04	0.23	0.37	0.06	0.02	0.00	0.00
NMS	0.02	0.03	0.05	0.03	0.35	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.05	0.06	0.28	0.05	0.03	0.00
BDS	0.04	0.01	0.07	0.04	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01	0.00	0.04	0.04	0.01	0.01	0.01	0.03	0.03	0.60	0.00	0.01
GPS	0.00	0.00	0.00	0.01	0.19	0.00	0.00	0.00	0.04	0.00	0.02	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.01
WAT	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.00	0.89
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT

(a) Gapfilled-SVGP

True																								
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT	
CUF	0.59	0.21	0.12	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	
DUF	0.20	0.55	0.08	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.03	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.00	0.00
ICU	0.15	0.17	0.37	0.11	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.03	0.03	0.00	0.01	0.01	0.01	0.02	0.04	0.00	0.00	
RSF	0.02	0.04	0.06	0.77	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.00	
RAP	0.00	0.00	0.00	0.00	0.88	0.02	0.07	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
STC	0.00	0.00	0.00	0.01	0.02	0.82	0.09	0.00	0.00	0.00	0.00	0.01	0.02	0.01	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	
PRO	0.00	0.01	0.01	0.01	0.05	0.12	0.63	0.02	0.01	0.02	0.00	0.04	0.03	0.01	0.04	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.79	0.05	0.08	0.00	0.03	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
SUN	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.04	0.80	0.02	0.00	0.07	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
COR	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.07	0.02	0.84	0.01	0.01	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
TUB	0.01	0.01	0.01	0.00	0.01	0.01	0.04	0.04	0.09	0.01	0.00	0.71	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	
GRA	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.64	0.04	0.02	0.04	0.02	0.10	0.06	0.01	0.01	0.00	0.01	
ORC	0.00	0.02	0.01	0.02	0.00	0.02	0.02	0.00	0.00	0.01	0.00	0.01	0.05	0.64	0.10	0.03	0.01	0.03	0.02	0.00	0.01	0.00	0.00	
VIN	0.00	0.03	0.01	0.00	0.00	0.01	0.02	0.01	0.01	0.01	0.00	0.02	0.01	0.03	0.79	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00	
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.81	0.05	0.03	0.05	0.00	0.01	0.00	0.00	
COF	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.05	0.80	0.01	0.06	0.02	0.01	0.00	0.00	
NGL	0.00	0.01	0.01	0.01	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.08	0.03	0.02	0.07	0.05	0.41	0.20	0.06	0.02	0.02	0.00	
WOM	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.05	0.03	0.02	0.06	0.09	0.26	0.32	0.08	0.03	0.00	0.00	
NMS	0.00	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.04	0.04	0.73	0.06	0.03	0.00	
BDS	0.01	0.03	0.05	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.01	0.01	0.01	0.02	0.02	0.05	0.72	0.00	0.01	
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.92	0.00	
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.02	0.00	0.93	

(b) EmTAN-MLP

True																							
	CUF	DUF	ICU	RSF	RAP	STC	PRO	SOY	SUN	COR	RIC	TUB	GRA	ORC	VIN	BLF	COF	NGL	WOM	NMS	BDS	GPS	WAT
CUF	0.64	0.18	0.10	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
DUF	0.22	0.58	0.07	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00
ICU	0.15	0.15	0.42	0.11	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.00	0.01	0.01	0.01	0.02	0.04	0.00	0.00
RSF	0.02	0.03	0.07	0.79	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.00
RAP	0.00	0.00	0.00	0.00	0.91	0.01	0.05	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
STC	0.00	0.00	0.00	0.00	0.02	0.85	0.07	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
PRO	0.00	0.01	0.00	0.00	0.05	0.10	0.64	0.02	0.01	0.02	0.00	0.05	0.03	0.01	0.04	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00
SOY	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.83	0.04	0.05	0.00	0.03	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.03	0.83	0.01	0.00	0.06	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
COR	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.04	0.01	0.89	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RIC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TUB	0.01	0.02	0.01	0.00	0.01	0.01	0.03	0.03	0.09	0.01	0.00	0.74	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GRA	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.67	0.03	0.01	0.04	0.02	0.09	0.07	0.01	0.00	0.00	0.01
ORC	0.00	0.02	0.01	0.01	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.01	0.05	0.67	0.09	0.04	0.01	0.02	0.02	0.00	0.01	0.00	0.00
VIN	0.01	0.02	0.01	0.00	0.00	0.01	0.02	0.01	0.01	0.00	0.00	0.01	0.01	0.02	0.84	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00
BLF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.03	0.00	0.83	0.05	0.03	0.03	0.00	0.00	0.00	0.00
COF	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.05	0.83	0.01	0.04	0.01	0.01	0.00	0.00
NGL	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.07	0.02	0.02	0.07	0.05	0.43	0.20	0.05	0.01	0.02	0.00
WOM	0.00	0.01	0.02	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.05	0.02	0.01	0.06	0.08	0.20	0.43	0.06	0.02	0.00	0.00
NMS	0.00	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.04	0.06	0.74	0.05	0.03	0.01
BDS	0.01	0.02	0.03	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.01	0.02	0.02	0.03	0.04	0.77	0.00	0.01
GPS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.92	0.00
WAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.01	0.00	0.93

(c) EmTAN-LTAE

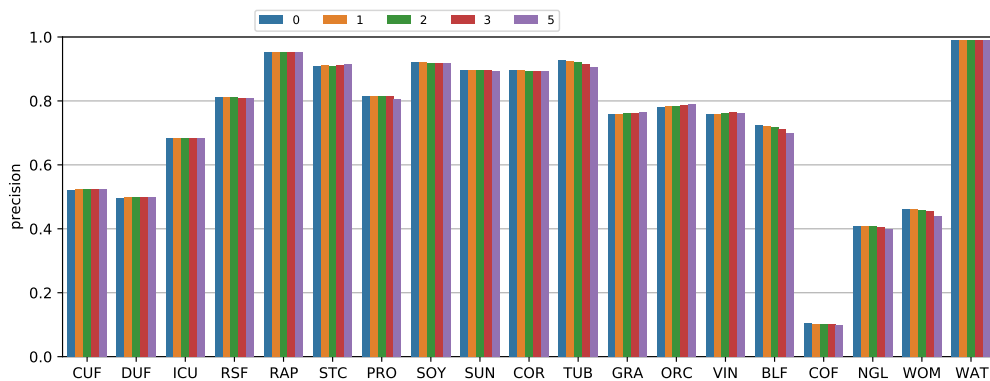
Figure C.1: Normalized confusion matrices for the Gapfilled-SVGP, EmTAN-MLP and EmTAN-LTAE models.

C.2. Additional results: Robustness to the temporal sampling

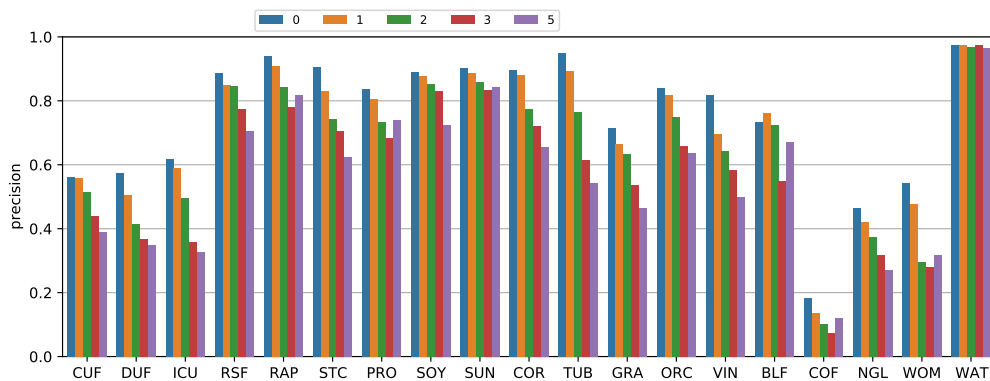
In the following, we present additional results for the Section 9.1.3.

C.2.1. Precision and recall per class

Two figures are provided in the following. They correspond to the barplots of the precision and recall per class, respectively, computed with artificially shifted acquisition dates from the *test* data for the two models *mTANe-SVGP* and *raw-LTAE*. The nomenclature of the classes is presented in Table 3.3.

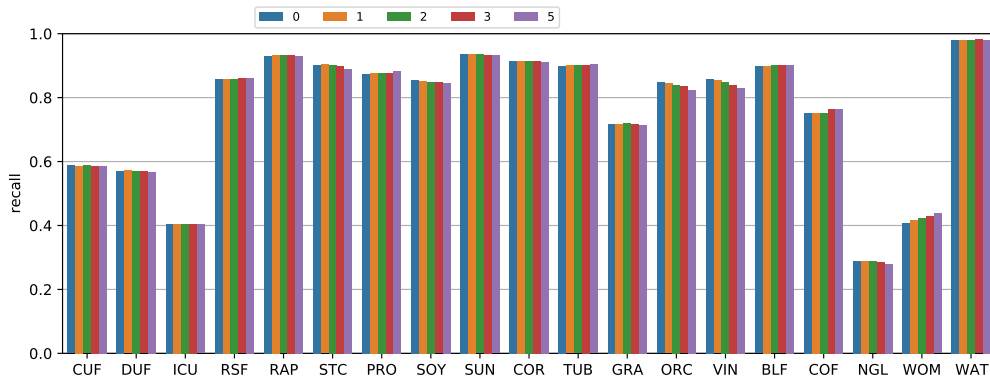


(a) *EmTAN-SVGP*

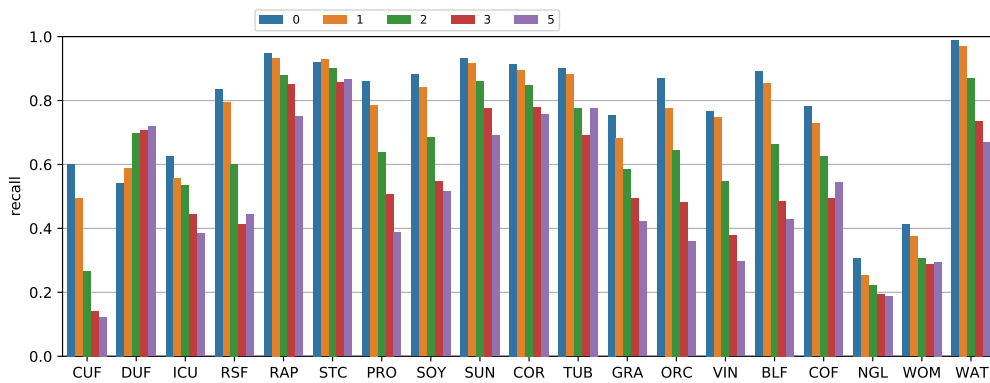


(b) *raw-LTAE*

Figure C.2: Barplots of the precision per class for the *EmTAN-SVGP* and *raw-LTAE* models computed with the test data set only on the *T31TCJ* tile over nine runs.



(a) *EmTAN-SVGP*



(b) *raw-LTAE*

Figure C.3: Barplots of the recall per class for the *EmTAN-SVGP* and *raw-LTAE* models computed with the test data set only on the *T31TCJ* tile over nine runs.

In the interest of reproducible research, the data sets, the trained models and the land cover maps are made available. Moreover, the implementation of all the models is provided. They are given for each part: Part **II** and **III**.

D.1. Part II

D.1.1. Data sets

The classification data set preprocessed for each region can be downloaded at the following link: [10.5281/zenodo.7099785](https://zenodo.org/record/7099785). The boundary data set preprocessed for different sizes of boundary can be downloaded at the following link: [10.5281/zenodo.7099783](https://zenodo.org/record/7099783).

D.1.2. Best trained models

Best trained models for each region and in global configuration are available here: [10.5281/zenodo.7104552](https://zenodo.org/record/7104552). These models were used to produce the land cover maps of the next section.

D.1.3. Land cover maps

The land cover maps for all studied models on two different tiles (*T31TCJ* and *T31TDJ*) in both configurations are available for download: [10.5281/zenodo.7077887](https://zenodo.org/record/7077887).

D.1.4. Code

The implementation of the models is made available in the following repository: https://gitlab.com/Valentine-Bellet/land_cover_southfrance_gp.

D.2. Part III

D.2.1. Data sets

The classification data set preprocessed can be downloaded at the following link: [10.5281/zenodo.8033058](https://zenodo.org/record/8033058).

D.2.2. Best trained models

Best trained models (mTAN-GP, mTAN-MLP, mTAN-LTAE, and raw-LTAE) are available here: [10.5281/zenodo.8033364](https://doi.org/10.5281/zenodo.8033364). These models were used to produce the land cover maps of the next section.

D.2.3. Land cover maps

The land cover maps for all studied models on two different tiles (*T31TCJ* and *T31TDJ*) are available for download: [10.5281/zenodo.8033902](https://doi.org/10.5281/zenodo.8033902).

D.2.4. Code

The implementation of the models is made available in the following repository: https://gitlab.com/Valentine-Bellet/land_cover_southfrance_mt看_gp_irregular_sits.

- ANITI** Natural Intelligence Toulouse Institute. 22, 24, 32–34
- ANN** Artificial Neural Networks. 72
- AR6** Sixth Assessment Report. 37
- ASP** Agence de Services et de Paiement. 94
- BISE** Best Index Slope Extraction. 202
- BNN** Bayesian neural networks. 78
- BPTT** Backpropagation through time. 75
- CES OSO** Centre d’Expertise Scientifique Occupation des SOIs. 86, 193, 247–249
- CESBIO** Centre d’Études Spatiales de la Biosphère. 22–24, 32–34, 47
- CLC** CORINE Land Cover. 60, 94
- CNES** Centre National d’Études Spatiales. 22–24, 32, 33, 46
- CNN** Convolutional Neural Networks. 72–76, 79, 168
- CNRS** Centre national de la recherche scientifique. 22, 32
- CPU** Central Processing Unit. 132
- DL** Deep Learning. 72, 167, 172, 247, 248, 258
- DTC** Deterministic Training Conditional. 134
- DTW** Dynamic Time Warping. 77
- ECV** Essential Climate Variables. 39
- EEA** European Environment Agency. 60, 94
- ELBO** Evidence Lower Bound. 136, 137, 140, 142, 250
- EmTAN** Extended multi Time Attention Networks. 211, 219, 220, 224, 248–251
- EO** Earth Observation. 21, 39, 41, 46, 47, 49

- EP** Expectation Propagation. 122, 131
- ESA** European Space Agency. 82
- EVI** Enhanced Vegetation Index. 77
- FAO** Food and Agriculture Organization. 39
- FCN** Fully Convolutional Network. 73, 75
- FITC** Fully Independent Training Conditional. 134
- GBDT** Gradient Boosted Decision Tree. 61
- GCOS** Global Climate Observing System. 39
- GD** Gradient descent. 72
- GLIMS** Global Land Ice Measurements from Space. 94
- GMM** Gaussian Mixture Models. 57
- GP** Gaussian Processes. 21, 22, 69, 71, 78–80, 106, 111, 112, 116, 117, 121–125, 127, 130, 131, 134, 135, 137, 138, 140, 142–145, 151–153, 160, 162, 163, 167, 168, 172, 181, 183, 184, 187, 189, 214, 218, 247, 248
- GPU** Graphics Processing Unit. 72, 152, 248
- GRU** Gated Recurrent Unit. 75
- HANTS** Harmonic ANalysis of Time Series. 202
- HPC** High-Performance Computing. 23, 72, 238
- ICM** Intrinsic Co-regionalization Model. 125
- IDR** Iterative interpolation for Data Reconstruction. 202
- IGBP** International Geosphere Biosphere Programme. 40
- IGN** Institut Géographique National. 94
- INRAe** Institut National de Recherche pour l’agriculture, l’alimentation et l’environnement. 22, 24, 32, 33
- IP** Inducing Points. 134, 135, 145, 154, 156, 177, 181, 183, 187, 221, 240, 247, 248, 250
- IPBES** Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services. 37
- IPCC** Intergovernmental Panel on Climate Change. 37

- IRD** Institut de recherche pour le développement. 22, 32
- ISPRS** International Society for Photogrammetry and Remote Sensing. 82
- JSTARS** Journal of Selected Topics in Applied Earth Observations. 82
- KL** Kullback–Leibler. 109, 142
- LA** Laplace Approximation. 122, 131
- LAI** Leaf area index. 71, 130
- LCCS** Land Cover Classification System. 39, 40
- LDA** Linear Discriminant Analysis. 155, 156, 189, 190
- LIDAR** Laser Imaging, Detection, And Ranging. 42, 46
- LMC** Linear Model of Co-regionalization. 124–127, 130, 131, 140, 142, 151, 248
- LSTM** Long Short Term Memory. 75
- LTAE** Lightweight Temporal Attention Encoder. 76, 153, 160, 162, 163, 167, 168, 172, 173, 220, 224, 247
- LUCAS** Land Use/Cover Area frame statistical Survey. 60, 61
- LULC** Land Use and Land Cover. 21, 39, 49, 50, 55, 60, 61, 65, 71, 72, 82, 247–249, 251
- MAJA** MACCS-ATCOR Joint Algorithm. 86
- MAP** *maximum a posteriori*. 116
- MC** Monte Carlo. 137, 142, 143, 160, 167, 179, 184, 228, 250
- MCMC** Markov Chain Monte Carlo. 122, 131
- MEA** Millennium Ecosystem Assessment. 36, 37
- ML** Machine Learning. 64, 68, 71, 74, 77, 111, 123
- MLC** Maximum Likelihood Classification. 57
- MLP** Multi-layer Perceptron. 72–76, 78, 117, 153, 156, 160, 162, 163, 167, 168, 172, 173, 189, 190, 217, 218, 220, 224, 247, 249
- MMU** Minimum Mapping Unit. 39, 94
- MODIS** Moderate Resolution Imaging Spectroradiometer. 46, 49, 50, 130
- MSI** Multi-Spectral Instrument. 82
- mTAN** multi Time Attention Networks. 209–211, 214, 216

- NASA** National Aeronautics and Space Administration. 46
- NDBI** Normalized Difference Built-up Index. 77
- NDVI** Normalized Difference Vegetation Index. 42, 46, 56, 86, 87, 90, 154, 199–202, 206, 241
- NDWI** Normalized Difference Water Index. 77, 86, 87, 154
- NLP** Natural Language Processing. 75
- OA** Overall Accuracy. 153, 154, 160, 163, 172, 177, 179, 181, 189, 221, 224, 232, 235, 238, 240, 263, 267
- OB** Object-Based. 58
- OCS GE** OCcupation des Sols Grande Échelle. 60
- OMP** Observatoire Midi-Pyrénées. 24, 33
- OSM** OpenStreetMap. 56
- OSO** Occupation des SOls. 61, 64, 65, 79, 86, 87, 94, 249
- PB** Pixel-Based. 58
- PDF** Probability Density Function. 57, 106, 107
- PITC** Partially Independent Training Conditional. 134
- RAM** Random-Access Memory. 132
- RBF** Radial Basis Function. 69, 112, 117, 144, 152, 177, 183, 206, 217–219, 242
- ReLU** Rectified Linear Unit. 153, 217
- RF** Random Forests. 57, 61, 71, 72, 74, 76–79, 117, 147, 152, 160, 162, 163, 167, 168, 172, 173, 193, 247–249
- RGB** Red, Green and Blue. 87
- RGI** Randolph Glacier Inventory. 94
- RMSE** Root Mean Squared Error. 127
- RNN** Recurrent Neural Networks. 72, 74, 75, 79
- RPG** Registre Parcellaire Graphique. 94
- RQ** Rational Quadratic. 112
- RSE** Remote Sensing of Environment. 82
- RVI** Radar Vegetation Index. 130

- SAR** Synthetic Aperture Radar. 42
- SAVI** Soil Adjusted Vegetation Index. 77
- SDU2E** Sciences de l'Univers, de l'Environnement et de l'Espace. 24, 33
- SGD** Stochastic Gradient Descent. 72
- SITS** Satellite Image Time-Series. 21, 22, 49, 58, 61, 64, 66, 67, 71, 72, 74–76, 78, 79, 206, 211, 247, 248
- SLFM** Semiparametric Latent Factor Model. 125, 126, 130
- SVGP** Stochastic Variational Gaussian Processes. 21, 22, 138, 140, 156, 193, 214, 217–221, 224, 238, 240, 241, 247, 248, 250
- SVM** Support Vector Machine. 57, 64, 69, 71, 72, 74, 76, 121, 122, 133
- TAE** Temporal Attention Encoder. 75
- TERUTI** UTILisation du TERritoire. 61
- TGRS** Transactions on Geoscience and Remote Sensing. 82
- TPE** Thermal Positional Encoding. 76
- TWDTW** Time-Weighted Dynamic Time Warping. 77
- UA** Urban Atlas. 60, 94
- UAV** Unmanned Aerial Vehicles. 46
- USGS** United States Geological Survey. 46
- UT3** Université Toulouse III. 22, 24, 32, 34
- VAE** Variational Auto-encoder. 142
- VGI** Volunteered Geographic Information. 55, 56
- VI** Variational Inference. 135, 138, 247
- VNNGP** Variational Nearest Neighbor Gaussian Processes. 250

REFERENCES

- [Abadi et al., 2015] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. <https://www.tensorflow.org/>.
- [Abdikan et al., 2016] Abdikan, S., Balik Sanli, F., Üstüner, M., and Calò, F. (2016). Land cover mapping using sentinel-1 sar data. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B7:757–761.
- [Adam et al., 2014] Adam, E., Mutanga, O., Odindi, J., and Abdel-Rahman, E. M. (2014). Land-use/cover classification in a heterogeneous coastal landscape using rapideye imagery: evaluating the performance of random forest and support vector machines classifiers. *International Journal of Remote Sensing*, 35(10):3440–3458.
- [Alexander and Fairbridge, 1999] Alexander, D. and Fairbridge, R. (1999). *Encyclopedia of Environmental Science*. Encyclopedia of Environmental Science. Springer Netherlands.
- [Álvarez et al., 2012] Álvarez, M. A., Rosasco, L., and Lawrence, N. D. (2012). Kernels for Vector-Valued Functions: A Review. *Found. Trends Mach. Learn.*, 4(3):195–266.
- [Anderson, 1976] Anderson, J. R. (1976). *A land use and land cover classification system for use with remote sensor data*, volume 964. US Government Printing Office.
- [ASP, 2018] ASP, I. (2018). RPG Version 2.0 - Registre Parcellaire Graphique. Technical report, ASP, IGN.
- [Azzimonti et al., 2016] Azzimonti, D., Bect, J., Chevalier, C., and Ginsbourger, D. (2016). Quantifying uncertainties on excursion sets under a gaussian random field prior. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):850–874.
- [Badrinarayanan et al., 2017] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495.
- [Baetens et al., 2019] Baetens, L., Desjardins, C., and Hagolle, O. (2019). Validation of Copernicus Sentinel-2 Cloud Masks Obtained from MAJA, Sen2Cor, and FMask Processors Using Reference Cloud Masks Generated with a Supervised Active Learning Procedure. *Remote Sensing*, 11(4):433.

- [Bahdanau et al., 2014] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.
- [Baldassarre et al., 2012] Baldassarre, L., Rosasco, L., Barla, A., and Verri, A. (2012). Multi-output learning via spectral filtering. *Machine learning*, 87:259–301.
- [Baldrige et al., 2009] Baldrige, A. M., Hook, S. J., Grove, C., and Rivera, G. (2009). The aster spectral library version 2.0. *Remote sensing of environment*, 113(4):711–715.
- [Baudoux et al., 2021] Baudoux, L., Inglada, J., and Mallet, C. (2021). Toward a Yearly Country-Scale CORINE Land-Cover Map without Using Images: A Map Translation Approach. *Remote Sensing*, 13(6):1060.
- [Bazi and Melgani, 2006] Bazi, Y. and Melgani, F. (2006). Toward an Optimal SVM Classification System for Hyperspectral Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, 44(11):3374–3385.
- [Bazi and Melgani, 2008] Bazi, Y. and Melgani, F. (2008). Classification of hyperspectral remote sensing images using gaussian processes. In *IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium*, volume 2, pages II–1013. IEEE.
- [Bazi and Melgani, 2010] Bazi, Y. and Melgani, F. (2010). Gaussian Process Approach to Remote Sensing Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 48(1):186–197.
- [Bazi et al., 2012] Bazi, Y., Alajlan, N., and Melgani, F. (2012). Improved estimation of water chlorophyll concentration with semisupervised gaussian process regression. *IEEE Transactions on Geoscience and Remote Sensing*, 50(7):2733–2743.
- [Beck et al., 2006] Beck, P. S., Atzberger, C., Høgda, K. A., Johansen, B., and Skidmore, A. K. (2006). Improved monitoring of vegetation dynamics at very high latitudes: A new method using modis ndvi. *Remote sensing of Environment*, 100(3):321–334.
- [Belgiu and Drăguț, 2016] Belgiu, M. and Drăguț, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114:24–31.
- [Belward, 1996] Belward, A. (1996). The igbp-dis global 1 km land cover data set “discover”: Proposal and implementation plans. *IGB-DIS Working Paper No. 13, IGBP-DIS Office, Meteo-France*.
- [Benediktsson et al., 1990] Benediktsson, J., Swain, P., and Ersoy, O. (1990). Neural network approaches versus statistical methods in classification of multisource remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 28(4):540–552.
- [Bertini et al., 2012] Bertini, F., Brand, O., Carlier, S., Del Bello, U., Drusch, M., Duca, R., Fernandez, V., Ferrario, C., Ferreira, M. H., Isola, C., Kirschner, V., Laberinti, P., Lambert, M., Mandorlo, G., Marcos, P., Martimort, P., Moon, S., Oldeman, P., Palomba, M., and Pineiro, J. (2012). Sentinel-2 esa’s optical high-resolution mission for gmes operational services. *ESA bulletin. Bulletin ASE. European Space Agency*, SP-1322.

- [Biau and Scornet, 2016] Biau, G. and Scornet, E. (2016). A random forest guided tour. *Test*, 25:197–227.
- [Bishop, 1995] Bishop, C. M. (1995). *Neural networks for pattern recognition*. Clarendon Press ; Oxford University Press, Oxford : New York.
- [Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.
- [Blaschke et al., 2000] Blaschke, T., Lang, S., Lorup, E., Strobl, J., and Zeil, P. (2000). Object-oriented image processing in an integrated gis/remote sensing environment and perspectives for environmental applications. *Environmental information for planning, politics and the public*, 2(1995):555–570.
- [Bonilla et al., 2007] Bonilla, E. V., Chai, K. M. A., and Williams, C. K. I. (2007). Multi-task gaussian process prediction. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3–6, 2007*, pages 153–160. Curran Associates, Inc.
- [Borra et al., 2019] Borra, S., Thanki, R., and Dey, N. (2019). *Satellite image analysis: clustering and classification*. Springer.
- [Bossard et al., 2000] Bossard, M., Feranec, J., Otahel, J., and others (2000). *CORINE land cover technical guide: Addendum 2000*, volume 40. European Environment Agency Copenhagen.
- [Bottou et al., 2007] Bottou, L., Lin, C.-J., et al. (2007). Support vector machine solvers. *Large scale kernel machines*, 3(1):301–320.
- [Bottou et al., 2018] Bottou, L., Curtis, F. E., and Nocedal, J. (2018). Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, 60(2):223–311.
- [Boyle and Frean, 2004] Boyle, P. and Frean, M. R. (2004). Dependent gaussian processes. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13–18, 2004, Vancouver, British Columbia, Canada]*, pages 217–224.
- [Brando and Dekker, 2003] Brando, V. E. and Dekker, A. G. (2003). Satellite hyperspectral remote sensing for estimating estuarine and coastal water quality. *IEEE transactions on geoscience and remote sensing*, 41(6):1378–1387.
- [Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.
- [Brown et al., 2022] Brown, C., Brumby, S., Guzder-Williams, B., Birch, T., Hyde, S., Mazziariello, J., Czerwinski, W., Pasquarella, V., Haertel, R., Ilyushchenko, S., Schwehr, K., Weisse, M., Stolle, F., Hanson, C., Guinan, O., Moore, R., and Tait, A. (2022). Dynamic world, near real-time global 10 m land use land cover mapping. *Scientific Data*, 9:251.
- [Brown and Nocera, 2017] Brown, L. J. and Nocera, J. J. (2017). Conservation of breeding grassland birds requires local management strategies when hay maturation and nutritional quality differ among regions. *Agriculture, Ecosystems & Environment*, 237:242–249.

- [Buchhorn et al., 2020] Buchhorn, M., Lesiv, M., Tsendbazar, N.-E., Herold, M., Bertels, L., and Smets, B. (2020). Copernicus global land cover layers—collection 2. *Remote Sensing*, 12(6):1044.
- [Büttner and Maucha, 2006] Büttner, G. and Maucha, G. (2006). The thematic accuracy of corine land cover 2000. assessment using lucas (land use/cover area frame statistical survey). *European Environment Agency: Copenhagen, Denmark*, 7.
- [Büttner, 2014] Büttner, G. (2014). Corine land cover and land cover change products. In *Land use and land cover mapping in Europe: practices & trends*, pages 55–74. Springer.
- [Caballero et al., 2023] Caballero, G., Pezzola, A., Winschel, C., Sanchez Angonova, P., Casella, A., Orden, L., Salinero-Delgado, M., Reyes-Muñoz, P., Berger, K., Delegido, J., and Verrelst, J. (2023). Synergy of sentinel-1 and sentinel-2 time series for cloud-free vegetation water content mapping with multi-output gaussian processes. *Remote Sensing*, 15(7).
- [Cai et al., 2023] Cai, X., Bi, Y., Nicholl, P. N., and Sterritt, R. (2023). Rethinking the encoding of satellite image time series. *CoRR*, abs/2305.02086.
- [Camargo et al., 2019] Camargo, F. F., Sano, E. E., Almeida, C. M., Mura, J. C., and Almeida, T. (2019). A comparative assessment of machine-learning techniques for land use and land cover classification of the brazilian tropical savanna using alos-2/palsar-2 polarimetric images. *Remote Sensing*, 11(13):1600.
- [Camps-Valls et al., 2004] Camps-Valls, G., Gomez-Chova, L., Calpe-Maravilla, J., Martin-Guerrero, J., Soria-Olivas, E., Alonso-Chorda, L., and Moreno, J. (2004). Robust support vector method for hyperspectral data classification and knowledge discovery. *IEEE Transactions on Geoscience and Remote Sensing*, 42(7):1530–1542.
- [Camps-Valls et al., 2006] Camps-Valls, G., Gomez-Chova, L., Munoz-Mari, J., Vila-Frances, J., and Calpe-Maravilla, J. (2006). Composite kernels for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 3(1):93–97.
- [Camps-Valls et al., 2012] Camps-Valls, G., Tuia, D., Gómez-Chova, L., Jiménez, S., and Malo, J. (2012). Remote sensing from earth observation satellites. In *Remote Sensing Image Processing*, pages 1–25. Springer.
- [Camps-Valls et al., 2014] Camps-Valls, G., Tuia, D., Bruzzone, L., and Benediktsson, J. A. (2014). Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods. *IEEE Signal Processing Magazine*, 31(1):45–54.
- [Camps-Valls et al., 2016] Camps-Valls, G., Verrelst, J., Munoz-Mari, J., Laparra, V., Mateo-Jimenez, F., and Gomez-Dans, J. (2016). A Survey on Gaussian Processes for Earth-Observation Data Analysis: A Comprehensive Investigation. *IEEE Geoscience and Remote Sensing Magazine*, 4(2):58–78.
- [Cano et al., 2017] Cano, E., Denux, J.-P., Bisquert, M., Hubert-Moy, L., and Chéret, V. (2017). Improved forest-cover mapping based on modis time series and landscape stratification. *International Journal of Remote Sensing*, 38(7):1865–1888.

- [Cantelaube and Carles, 2014] Cantelaube, P. and Carles, M. (2014). Le registre parcellaire graphique: des données géographiques pour décrire la couverture du sol agricole. *Cahier des Techniques de l'INRA*, (Méthodes et techniques GPS et SIG pour la conduite de dispositifs expérimentaux):58–64.
- [Capliez et al., 2023] Capliez, E., Ienco, D., Gaetano, R., Baghdadi, N., and Salah, A. H. (2023). Temporal-domain adaptation for satellite image time-series land-cover mapping with adversarial learning and spatially aware self-training. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:3645–3675.
- [Carranza-García et al., 2019] Carranza-García, M., García-Gutiérrez, J., and Riquelme, J. C. (2019). A framework for evaluating land use and land cover classification using convolutional neural networks. *Remote Sensing*, 11(3):274.
- [Carrasco et al., 2019] Carrasco, L., O'Neil, A. W., Morton, R. D., and Rowland, C. S. (2019). Evaluating combinations of temporally aggregated sentinel-1, sentinel-2 and landsat 8 for land cover mapping with google earth engine. *Remote Sensing*, 11(3):288.
- [Chamorro-Martinez et al., 2021] Chamorro-Martinez, J. A., Cué La Rosa, L. E., Feitosa, R. Q., Sanches, I. D., and Happ, P. N. (2021). Fully convolutional recurrent networks for multirate crop recognition from multitemporal image sequences. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:188–201.
- [Chang and Bai, 2018] Chang, N.-B. and Bai, K. (2018). *Multisensor data fusion and machine learning for environmental remote sensing*. CRC Press.
- [Chapin et al., 2011] Chapin, F., Chapin, M., Matson, P., and Vitousek, P. (2011). *Principles of Terrestrial Ecosystem Ecology*. Biomedical and Life Sciences. Springer New York.
- [Chen et al., 2004] Chen, J., Jönsson, P., Tamura, M., Gu, Z., Matsushita, B., and Eklundh, L. (2004). A simple method for reconstructing a high-quality ndvi time-series data set based on the savitzky–golay filter. *Remote Sensing of Environment*, 91(3):332–344.
- [Chen et al., 2020] Chen, M.-H., Li, B., Bao, Y., AlRegib, G., and Kira, Z. (2020). Action segmentation with joint self-supervised temporal domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Chung et al., 2014] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.
- [Cihlar, 2000] Cihlar, J. (2000). Land cover mapping of large areas from satellites: status and research priorities. *International journal of remote sensing*, 21(6-7):1093–1114.
- [Comber et al., 2005] Comber, A., Fisher, P., and Wadsworth, R. (2005). What is land cover? *Environment and Planning B: Planning and Design*, 32(2):199–209.
- [Congalton, 1991] Congalton, R. G. (1991). A review of assessing the accuracy of classifications of remotely sensed data. *Remote sensing of environment*, 37(1):35–46.

- [Congalton et al., 2014] Congalton, R. G., Gu, J., Yadav, K., Thenkabail, P. S., and Ozdogan, M. (2014). Global land cover mapping: A review and uncertainty analysis. *Remote Sens.*, 6:12070–12093.
- [Conover, 1999] Conover, W. (1999). *Practical nonparametric statistics*. Wiley series in probability and statistics. Wiley, New York, NY [u.a.], 3. ed edition.
- [Constantin et al., 2021] Constantin, A., Fauvel, M., and Girard, S. (2021). Joint Supervised Classification and Reconstruction of Irregularly Sampled Satellite Image Times Series. *IEEE Transactions on Geoscience and Remote Sensing*, 60:4403913.
- [Constantin et al., 2022] Constantin, A., Fauvel, M., and Girard, S. (2022). Mixture of multivariate gaussian processes for classification of irregularly sampled satellite image time-series. *Statistics and Computing*, 32(5):79.
- [Costa et al., 2022] Costa, H., Benevides, P., Moreira, F. D., Moraes, D., and Caetano, M. (2022). Spatially stratified and multi-stage approach for national land cover mapping based on sentinel-2 data and expert knowledge. *Remote Sensing*, 14(8).
- [Cristianini and Shawe-Taylor, 2000] Cristianini, N. and Shawe-Taylor, J. (2000). *Support Vector Machines*, page 93–124. Cambridge University Press.
- [Curran and Atkinson, 1998] Curran, P. J. and Atkinson, P. M. (1998). Geostatistics and remote sensing. *Progress in Physical Geography: Earth and Environment*, 22(1):61–78.
- [Dalimier et al., 2021] Dalimier, J., Claverie, M., Goffart, B., Jungers, Q., Lamarche, C., De Maet, T., and Defourny, P. (2021). Characterizing the Congo Basin Forests by a Detailed Forest Typology Enriched with Forest Biophysical Variables. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 673–676.
- [Davi et al., 2011] Davi, H., Gillmann, M., Ibanez, T., Cailleret, M., Bontemps, A., Fady, B., and Lefèvre, F. (2011). Diversity of leaf unfolding dynamics among tree species: New insights from a study along an altitudinal gradient. *Agricultural and Forest Meteorology*, 151(12):1504–1513.
- [de Groot et al., 2010] de Groot, R., Alkemade, R., Braat, L., Hein, L., and Willemen, L. (2010). Challenges in integrating the concept of ecosystem services and values in landscape planning, management and decision making. *Ecological Complexity*, 7(3):260–272. Ecosystem Services – Bridging Ecology, Economy and Social Sciences.
- [de Oliveira et al., 2019] de Oliveira, S. S. T., Pascoal, L. M. L., Cardoso, M. d. C., Bueno, E. F., Rodrigues, V. J. S., and Martins, W. S. (2019). A parallel and distributed approach to the analysis of time series on remote sensing big data. *Journal of Information and Data Management*, 10(1):16–34.
- [De Oliveira et al., 2009] De Oliveira, T., de Oliveira, L. T., de Carvalho, L. M. T., Martinhago, A. Z., and de Freitas, S. G. (2009). Comparison of modis ndvi time series filtering by wavelets and fourier analysis to generate vegetation signatures. In *Proc. Anais XIV Simposio Brasileiro de Sensoramento Remoto, Natal, Brazil, 25– 30 April*, pages 1465–1472.

- [Deines et al., 2017] Deines, J. M., Kendall, A. D., and Hyndman, D. W. (2017). Annual irrigation dynamics in the us northern high plains derived from landsat satellite data. *Geophysical Research Letters*, 44(18):9350–9360.
- [Dell’Aglia et al., 2020] Dell’Aglia, D. A. G., Gargiulo, M., Iodice, A., Riccio, D., and Ruello, G. (2020). Fire Risk Analysis by using Sentinel-2 Data: The Case Study of the Vesuvius in Campania, Italy. In *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, pages 6806–6809.
- [Demir et al., 2013] Demir, B., Bovolo, F., and Bruzzone, L. (2013). Updating land-cover maps by classification of image time series: A novel change-detection-driven transfer learning approach. *IEEE Transactions on Geoscience and Remote Sensing*, 51(1):300–312.
- [Derksen, 2019] Derksen, D. (2019). *Contextual classification of large volumes of satellite imagery for the production of land cover maps over wide areas*. Theses, Université Paul Sabatier - Toulouse III.
- [Derksen et al., 2020] Derksen, D., Inglada, J., and Michel, J. (2020). Geometry aware evaluation of handcrafted superpixel-based features and convolutional neural networks for land cover mapping using satellite imagery. *Remote Sensing*, 12(3).
- [Devadas et al., 2012] Devadas, R., Denham, R. J., and Pringle, M. (2012). Support vector machine classification of object-based data for crop mapping, using multi-temporal landsat imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIX-B7:185–190.
- [Di Gregorio and Jansen, 1998] Di Gregorio, A. and Jansen, L. (1998). A new concept for a land-cover classification system. *The Land*, 2:55–65.
- [Dorogush et al., 2017] Dorogush, A. V., Gulin, A., Gusev, G., Kazeev, N., Prokhorenkova, L. O., and Vorobev, A. (2017). Fighting biases with dynamic boosting. *CoRR*, abs/1706.09516.
- [Dosovitskiy et al., 2021] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- [Dumeur et al., 2024] Dumeur, I., Valero, S., and Inglada, J. (2024). Self-supervised spatio-temporal representation learning of satellite image time series. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:4350–4367.
- [Durrande et al., 2010] Durrande, N., Ginsbourger, D., and Roustant, O. (2010). Additive Kernels for Gaussian Process Modeling.
- [Dusseux et al., 2013] Dusseux, P., Corpetti, T., and Hubert-Moy, L. (2013). Temporal kernels for the identification of grassland management using time series of high spatial resolution satellite images. *International Geoscience and Remote Sensing Symposium (IGARSS)*.

- [Duvenaud, 2014] Duvenaud, D. (2014). *Automatic Model Construction with Gaussian Processes*. PhD Thesis, Computational and Biological Learning Laboratory, University of Cambridge.
- [d'Andrimont et al., 2020] d'Andrimont, R., Yordanov, M., Martinez-Sanchez, L., Eiselt, B., Palmieri, A., Dominici, P., Gallego, J., Reuter, H. I., Joebges, C., Lemoine, G., et al. (2020). Harmonised lucas in-situ land cover and use database for field surveys from 2006 to 2018 in the european union. *Scientific data*, 7(1):352.
- [Eardley et al., 2016] Eardley, C., Freitas, B., Kevan, P., Rader, R., Gikungu, M., Klein, A., and Wiantoro, S. (2016). Background to pollinators, pollination and food production. *The Assessment Report on Pollinators, Pollination and Food Production; Potts, SG, Imperatriz-Fonseca, VL, Ngo, HT, Eds*, pages 1–25.
- [ESRI, 2021] ESRI (2021). Ai enables rapid creation of global land cover map.
- [Estrada-Carmona et al., 2022] Estrada-Carmona, N., Sánchez, A. C., Remans, R., and Jones, S. K. (2022). Complex agricultural landscapes host more biodiversity than simple ones: A global meta-analysis. *Proceedings of the National Academy of Sciences*, 119(38):e2203385119.
- [Everaerts et al., 2008] Everaerts, J. et al. (2008). The use of unmanned aerial vehicles (uavs) for remote sensing and mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(2008):1187–1192.
- [Fahrig et al., 2011] Fahrig, L., Baudry, J., Brotons, L., Burel, F. G., Crist, T. O., Fuller, R. J., Sirami, C., Siriwardena, G. M., and Martin, J.-L. (2011). Functional landscape heterogeneity and animal biodiversity in agricultural landscapes. *Ecology letters*, 14(2):101–112.
- [Fauvel, 2007] Fauvel, M. (2007). *Méthodes spatiales et spectrales pour la classification de zones urbaines en imagerie satellitaire*. PhD thesis. Thèse de doctorat dirigée par Chanussot, Jocelyn et Jón Atli Benediktsson, Signal, image, parole et télécom Grenoble INPG 2007.
- [Fauvel et al., 2012] Fauvel, M., Chanussot, J., and Benediktsson, J. A. (2012). A Spatial-Spectral Kernel-Based Approach for the Classification of Remote-Sensing Images. *Pattern Recogn.*, 45(1):381–392.
- [Fauvel et al., 2015] Fauvel, M., Bouveyron, C., and Girard, S. (2015). Parsimonious Gaussian Process Models for the Classification of Hyperspectral Remote Sensing Images. *IEEE Geoscience and Remote Sensing Letters*, 12(12):2423–2427.
- [Fauvel et al., 2020] Fauvel, M., Lopes, M., Dubo, T., Rivers-Moore, J., Frison, P.-L., Gross, N., and Ouin, A. (2020). Prediction of plant diversity in grasslands using Sentinel-1 and -2 satellite image time series. *Remote Sensing of Environment*, 237:111536.
- [Feng et al., 2019] Feng, S., Zhao, J., Liu, T., Zhang, H., Zhang, Z., and Guo, X. (2019). Crop Type Identification and Mapping Using Machine Learning Algorithms and Sentinel-2 Time Series Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9):3295–3306.

- [Feranec et al., 2016] Feranec, J., Hazeu, G., Kosztra, B., and Arnold, S. (2016). Corine land cover nomenclature. *European Landscape Dynamics: CORINE Land Cover Data*; Feranec, J., Soukup, T., Hazeu, G., Jaffrain, G., Eds, pages 17–25.
- [Ferraty et al., 2019] Ferraty, F., Zullo, A., and Fauvel, M. (2019). Nonparametric regression on contaminated functional predictor with application to hyperspectral data. *Econometrics and Statistics*, 9:95–107.
- [Fick and Hijmans, 2017] Fick, S. E. and Hijmans, R. J. (2017). Worldclim 2: new 1-km spatial resolution climate surfaces for global land areas. *International journal of climatology*, 37(12):4302–4315.
- [Fleiss et al., 2013] Fleiss, J. L., Levin, B., and Paik, M. C. (2013). *Statistical methods for rates and proportions*. John Wiley & Sons.
- [Foody, 2004] Foody, G. (2004). Thematic map comparison: Evaluating the statistical significance of differences in classification accuracy. *Photogrammetric Engineering and Remote Sensing*, 70:627–633.
- [Foody, 1992] Foody, G. M. (1992). On the compensation for chance agreement in image classification accuracy assessment. *Photogrammetric engineering and remote sensing*, 58(10):1459–1460.
- [Foody et al., 2006] Foody, G. M., Mathur, A., Sanchez-Hernandez, C., and Boyd, D. S. (2006). Training set size requirements for the classification of a specific class. *Remote Sensing of Environment*, 104(1):1–14.
- [Foody, 2020] Foody, G. M. (2020). Explaining the unsuitability of the kappa coefficient in the assessment and comparison of the accuracy of thematic maps obtained by image classification. *Remote Sensing of Environment*, 239:111630.
- [Foster et al., 2020] Foster, T., Mieno, T., and Brozović, N. (2020). Satellite-based monitoring of irrigation water use: Assessing measurement errors and their implications for agricultural water management policy. *Water Resources Research*, 56(11):e2020WR028378.
- [Franklin and Wulder, 2002] Franklin, S. and Wulder, M. (2002). Remote sensing methods in medium spatial resolution satellite data land cover classification of large areas. *Progress in Physical Geography*, 26(2):173–205.
- [Friedl and Brodley, 1997] Friedl, M. and Brodley, C. (1997). Decision tree classification of land cover from remotely sensed data. *Remote Sensing of Environment*, 61(3):399–409.
- [Furfaro et al., 2006] Furfaro, R., Morris, R. D., Kottas, A., Taddy, M., and Ganapol, B. D. (2006). A Gaussian Process Approach to Quantifying the Uncertainty of Vegetation Parameters from Remote Sensing Observations. 2006:B43A–0261.
- [G. J. Roerink and Verhoef, 2000] G. J. Roerink, M. M. and Verhoef, W. (2000). Reconstructing cloudfree ndvi composites using fourier analysis of time series. *International Journal of Remote Sensing*, 21(9):1911–1917.

- [Gallego-Elvira et al., 2013] Gallego-Elvira, B., Oliosio, A., Mira, M., Reyes-Castillo, S., Boulet, G., Marloie, O., Garrigues, S., Courault, D., Weiss, M., Chauvelon, P., and Boutron, O. (2013). Evaspa (evapotranspiration assessment from space) tool: An overview. *Procedia Environmental Sciences*, 19:303–310. Four Decades of Progress in Monitoring and Modeling of Processes in the Soil-Plant-Atmosphere System: Applications and Challenges.
- [Galy-Fajou and Opper, 2021] Galy-Fajou, T. and Opper, M. (2021). Adaptive inducing points selection for gaussian processes. *CoRR*, abs/2107.10066.
- [Gardner et al., 2018] Gardner, J. R., Pleiss, G., Weinberger, K. Q., Bindel, D., and Wilson, A. G. (2018). Gpytorch: Blackbox matrix-matrix gaussian process inference with GPU acceleration. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7587–7597.
- [Garioud et al., 2019] Garioud, A., Giordano, S., Valero, S., and Mallet, C. (2019). Challenges in grassland mowing event detection with multimodal sentinel images. In *2019 10th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp)*, pages 1–4.
- [Garnot and Landrieu, 2020] Garnot, V. S. F. and Landrieu, L. (2020). Lightweight temporal self-Attention for classifying satellite images time series. In *Workshop on Advanced Analytics and Learning on Temporal Data, AALTD*.
- [Garnot et al., 2020] Garnot, V. S. F., Landrieu, L., Giordano, S., and Chehata, N. (2020). Satellite image time series classification with pixel-set encoders and temporal self-attention. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12322–12331. IEEE.
- [Garnot and Landrieu, 2021] Garnot, V. S. F. and Landrieu, L. (2021). Panoptic segmentation of satellite image time series with convolutional temporal attention networks. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 4852–4861. IEEE.
- [Gascoin et al., 2019] Gascoin, S., Grizonnet, M., Bouchet, M., Salgues, G., and Hagolle, O. (2019). Theia snow collection: High-resolution operational snow cover maps from sentinel-2 and landsat-8 data. *Earth System Science Data*, 11(2):493–514.
- [Geirhos et al., 2019] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2019). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.
- [Ghamisi et al., 2019] Ghamisi, P., Rasti, B., Yokoya, N., Wang, Q., Hofle, B., Bruzzone, L., Bovolo, F., Chi, M., Anders, K., Gloaguen, R., Atkinson, P. M., and Benediktsson, J. A. (2019). Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 7(1):6–39.

- [Ghimire et al., 2010] Ghimire, B., Rogan, J., and Miller, J. (2010). Contextual land-cover classification: incorporating spatial dependence in land-cover classification models using random forests and the getis statistic. *Remote Sensing Letters*, 1(1):45–54.
- [Gislason et al., 2006] Gislason, P. O., Benediktsson, J. A., and Sveinsson, J. R. (2006). Random forests for land cover classification. *Pattern recognition letters*, 27(4):294–300.
- [Goovaerts and Goovaerts, 1997] Goovaerts, P. and Goovaerts, D. (1997). *Geostatistics for Natural Resources Evaluation*. Applied geostatistics series. Oxford University Press.
- [Goovaerts, 1997] Goovaerts, P. (1997). *Geostatistics for Natural Resources Evaluation*. Applied geostatistics series. Oxford University Press.
- [Grandini et al., 2020] Grandini, M., Bagli, E., and Visani, G. (2020). Metrics for multi-class classification: an overview. *ArXiv preprint*, abs/2008.05756.
- [Grekousis et al., 2015] Grekousis, G., Mountrakis, G., and Kavouras, M. (2015). An overview of 21 global and 43 regional land-cover mapping products. *International Journal of Remote Sensing*, 36(21):5309–5335.
- [Griffiths et al., 2013] Griffiths, P., van der Linden, S., Kuemmerle, T., and Hostert, P. (2013). A pixel-based landsat compositing algorithm for large area land cover mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6(5):2088–2101.
- [Grizonnet et al., 2017] Grizonnet, M., Michel, J., Poughon, V., Inglada, J., Savinaud, M., and Cresson, R. (2017). Orfeo toolbox: Open source processing of remote sensing images. *Open Geospatial Data, Software and Standards*, 2(1):15.
- [Guo et al., 2021] Guo, J., Luan, Y., Li, Z., Liu, X., Li, C., and Chang, X. (2021). Mozambique Flood (2019) Caused by Tropical Cyclone Idai Monitored From Sentinel-1 and Sentinel-2 Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:8761–8772.
- [Gupta and Gupta, 2019] Gupta, S. and Gupta, A. (2019). Dealing with noise problem in machine learning data-sets: A systematic review. *Procedia Computer Science*, 161:466–474. The Fifth Information Systems International Conference, 23-24 July 2019, Surabaya, Indonesia.
- [Halder et al., 2011] Halder, A., Ghosh, A., and Ghosh, S. (2011). Supervised and unsupervised landuse map generation from remotely sensed images using ant based systems. *Applied Soft Computing*, 11(8):5770–5781.
- [Haralick, 1979] Haralick, R. M. (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804.
- [Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.
- [He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

- [Hensman et al., 2013] Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. In Nicholson, A. and Smyth, P., editors, *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, UAI 2013, Bellevue, WA, USA, August 11-15, 2013*. AUAI Press.
- [Hensman et al., 2015] Hensman, J., de G. Matthews, A. G., and Ghahramani, Z. (2015). Scalable variational gaussian process classification. In Lebanon, G. and Vishwanathan, S. V. N., editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015*, volume 38 of *JMLR Workshop and Conference Proceedings*. JMLR.org.
- [Hermosilla et al., 2018] Hermosilla, T., Wulder, M. A., White, J. C., Coops, N. C., and Hobart, G. W. (2018). Disturbance-informed annual land cover classification maps of canada’s forested ecosystems for a 29-year landsat time series. *Canadian Journal of Remote Sensing*, 44(1):67–87.
- [Hermosilla et al., 2022] Hermosilla, T., Wulder, M. A., White, J. C., and Coops, N. C. (2022). Land cover classification in an era of big and open data: Optimizing localized implementation and training data selection to improve mapping outcomes. *Remote Sensing of Environment*, 268:112780.
- [Higdon et al., 1998] Higdon, D., Swall, J., and Kern, J. (1998). Non-stationary spatial modeling.
- [Higdon, 2002] Higdon, D. (2002). Space and Space-Time Modeling using Process Convolutions. In Anderson, C. W., Barnett, V., Chatwin, P. C., and El-Shaarawi, A. H., editors, *Quantitative Methods for Current Environmental Issues*, pages 37–56. Springer London, London.
- [Hilbert and Ostendorf, 2001] Hilbert, D. W. and Ostendorf, B. (2001). The utility of artificial neural networks for modelling the distribution of vegetation in past, present and future climates. *Ecological modelling*, 146(1-3):311–327.
- [Hill et al., 2020] Hill, J., Lopes, M., Frison, P.-L., Crowson, M., Warren-Thomas, E., Hariyadi, B., Kartika, W., Agus, F., Hamer, K., Stringer, L., and Pettorelli, N. (2020). Improving the accuracy of land cover classification in cloud persistent areas using optical and radar satellite image time series. *Methods in ecology and evolution*. © 2020 British Ecological Society. This is an author-produced version of the published paper. Uploaded in accordance with the publisher’s self-archiving policy. Further copying may not be permitted; contact the publisher for details.
- [Hirayama et al., 2019] Hirayama, H., Sharma, R. C., Tomita, M., and Hara, K. (2019). Evaluating multiple classifier system for the reduction of salt-and-pepper noise in the classification of very-high-resolution satellite images. *International Journal of Remote Sensing*, 40(7):2542–2557.
- [Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

- [Holtgrave et al., 2023] Holtgrave, A.-K., Lobert, F., Erasmi, S., Röder, N., and Kleinschmit, B. (2023). Grassland mowing event detection using combined optical, sar, and weather time series. *Remote Sensing of Environment*, 295:113680.
- [Hsu and Lin, 2002] Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425.
- [Hughes, 1968] Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *IEEE transactions on information theory*, 14(1):55–63.
- [Ienco et al., 2017] Ienco, D., Gaetano, R., Dupaquier, C., and Maurel, P. (2017). Land cover classification via multitemporal spatial data by deep recurrent neural networks. *IEEE Geoscience and Remote Sensing Letters*, 14(10):1685–1689.
- [IGN, 2022] IGN (2022). OCS GE Version 1.1 - Descriptif de contenu. Technical report, French IGN.
- [Inglada et al., 2015] Inglada, J., Arias, M., Tardy, B., Hagolle, O., Valero, S., Morin, D., Dedieu, G., Sepulcre, G., Bontemps, S., Defourny, P., et al. (2015). Assessment of an operational system for crop type map production using high temporal and spatial resolution satellite optical imagery. *Remote Sensing*, 7(9):12356–12379.
- [Inglada et al., 2016] Inglada, J., Vincent, A., Arias, M., and Tardy, B. (2016). *iota2-a25386*. <https://doi.org/10.5281/zenodo.58150>.
- [Inglada et al., 2017] Inglada, J., Vincent, A., Arias, M., Tardy, B., Morin, D., and Rodes, I. (2017). Operational High Resolution Land Cover Map Production at the Country Scale Using Satellite Image Time Series. *Remote Sensing*, 9(1).
- [Inglada et al., 2018] Inglada, J., Vincent, A., and Thierion, V. (2018). Theia oso land cover map 2018.
- [Interdonato et al., 2019] Interdonato, R., Ienco, D., Gaetano, R., and Ose, K. (2019). Duplo: A dual view point deep learning architecture for time series classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 149:91–104.
- [IPCC, 2023] IPCC (2023). SYNTHESIS REPORT OF THE IPCC SIXTH ASSESSMENT REPORT (AR6) longer report. https://report.ipcc.ch/ar6syr/pdf/IPCC_AR6_SYR_LongerReport.pdf.
- [Jensen, 1906] Jensen, J. L. W. V. (1906). Sur les fonctions convexes et les inégalités entre les valeurs Moyennes.
- [Jin et al., 2018] Jin, Y., Liu, X., Chen, Y., and Liang, X. (2018). Land-cover mapping using random forest classification and incorporating ndvi time-series and texture: a case study of central shandong. *International Journal of Remote Sensing*, 39(23):8703–8723.
- [Jing and Tian, 2020] Jing, L. and Tian, Y. (2020). Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):4037–4058.

- [Jing and Chao, 2020] Jing, S. and Chao, T. (2020). Time series land cover classification based on semi-supervised convolutional long short-term memory neural networks. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2020:1521–1528.
- [Joly et al., 2010] Joly, D., Brossard, T., Cardot, H., Cavailhes, J., Hilal, M., and Wavresky, P. (2010). Les types de climats en France, une construction spatiale. *Cybergeo: European Journal of Geography*.
- [Jonsson and Eklundh, 2002] Jonsson, P. and Eklundh, L. (2002). Seasonality extraction by function fitting to time-series of satellite sensor data. *IEEE transactions on Geoscience and Remote Sensing*, 40(8):1824–1832.
- [Jospin et al., 2022] Jospin, L. V., Laga, H., Boussaid, F., Buntine, W., and Bennamoun, M. (2022). Hands-on bayesian neural networks—a tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48.
- [Journel and Huijbregts, 1976] Journel, A. G. and Huijbregts, C. J. (1976). Mining geostatistics. *Academic Press*.
- [Julien and Sobrino, 2010] Julien, Y. and Sobrino, J. A. (2010). Comparison of cloud-reconstruction methods for time series of composite ndvi data. *Remote Sensing of Environment*, 114(3):618–625.
- [Karasiak et al., 2017] Karasiak, N., Sheeren, D., Fauvel, M., Willm, J., Dejoux, J.-F., and Monteil, C. (2017). Mapping tree species of forests in southwest France using Sentinel-2 image time series. In *2017 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp)*, pages 1–4.
- [Karra et al., 2021] Karra, K., Kontgis, C., Statman-Weil, Z., Mazzariello, J. C., Mathis, M., and Brumby, S. P. (2021). Global land use/land cover with sentinel 2 and deep learning. In *2021 IEEE international geoscience and remote sensing symposium IGARSS*, pages 4704–4707. IEEE.
- [Kavzoglu, 2009] Kavzoglu, T. (2009). Increasing the accuracy of neural network classification using refined training data. *Environmental Modelling & Software*, 24(7):850–858.
- [Khan et al., 2005] Khan, N. M., Rastoskuev, V. V., Sato, Y., and Shiozawa, S. (2005). Assessment of hydrosaline land degradation by using a simple approach of remote sensing indicators. *Agricultural Water Management*, 77(1):96–109. Special Issue on Land and Water Use: Environmental Management Tools and Practices.
- [Kingma and Welling, 2014] Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.
- [Kingma and Welling, 2019] Kingma, D. P. and Welling, M. (2019). An Introduction to Variational Autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392.

- [Klein et al., 2007] Klein, A.-M., Vaissière, B. E., Cane, J. H., Steffan-Dewenter, I., Cunningham, S. A., Kremen, C., and Tscharrntke, T. (2007). Importance of pollinators in changing landscapes for world crops. *Proceedings of the Royal Society B: Biological Sciences*, 274(1608):303–313.
- [Knapp, 2020] Knapp, S. (2020). Ecosystem - definition, examples and types.
- [Kondmann et al., 2021] Kondmann, L., Toker, A., Rußwurm, M., Camero, A., Peressuti, D., Milcinski, G., Mathieu, P.-P., Longépé, N., Davis, T., Marchisio, G. B., Leal-Taixé, L., and Zhu, X. (2021). Denethor: The dynamic earthnet dataset for harmonized, inter-operable, analysis-ready, daily crop monitoring from space. In *NeurIPS Datasets and Benchmarks*.
- [Kremen and Ostfeld, 2005] Kremen, C. and Ostfeld, R. S. (2005). A call to ecologists: Measuring, analyzing, and managing ecosystem services. *Frontiers in Ecology and the Environment*, 3(10):540–548.
- [Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.
- [Kuhn and Johnson, 2019] Kuhn, M. and Johnson, K. (2019). *Feature Engineering and Selection: A Practical Approach for Predictive Models*. Chapman and Hall/CRC, 1 edition.
- [Kumar et al., 2015] Kumar, P., Gupta, D. K., Mishra, V. N., and Prasad, R. (2015). Comparison of support vector machine, artificial neural network, and spectral angle mapper algorithms for crop classification using liss iv data. *International Journal of Remote Sensing*, 36(6):1604–1617.
- [Kussul et al., 2017] Kussul, N., Lavreniuk, M., Skakun, S., and Shelestov, A. (2017). Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14:778–782.
- [Landgrebe, 2005] Landgrebe, D. A. (2005). *Signal Theory Methods in Multispectral Remote Sensing*. Wiley, Newark, NJ.
- [Landis and Koch, 1977] Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.
- [Laso Bayas et al., 2020] Laso Bayas, J. C., See, L., Bartl, H., Sturn, T., Karner, M., Fraisl, D., Moorthy, I., Busch, M., van der Velde, M., and Fritz, S. (2020). Crowdsourcing lucas: Citizens generating reference land cover and land use data with a mobile app. *Land*, 9(11).
- [Lavreniuk et al., 2015] Lavreniuk, M., Kussul, N., Skakun, S., Shelestov, A., and Yailymov, B. (2015). Regional retrospective high resolution land cover for ukraine: Methodology and results. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3965–3968.

- [Le et al., 2022] Le, T. D. H., Pham, L. H., Dinh, Q. T., Hang, N. T. T., and Tran, T. A. T. (2022). Rapid method for yearly lulc classification using random forest and incorporating time-series ndvi and topography: a case study of thanh hoa province, vietnam. *Geocarto International*, 37(27):17200–17215.
- [LeCun et al., 1989] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551.
- [Lecun et al., 1998] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- [Leibfried et al., 2020] Leibfried, F., Dutordoir, V., John, S. T., and Durrande, N. (2020). A tutorial on sparse gaussian processes and variational inference. *ArXiv*, abs/2012.13962.
- [Leinenkugel et al., 2019] Leinenkugel, P., Deck, R., Huth, J., Ottinger, M., and Mack, B. (2019). The potential of open geodata for automated large-scale land use and land cover classification. *Remote Sensing*, 11(19):2249.
- [Li et al., 2021] Li, C., Ma, Z., Wang, L., Yu, W., Tan, D., Gao, B., Feng, Q., Guo, H., and Zhao, Y. (2021). Improving the accuracy of land cover mapping by distributing training samples. *Remote Sensing*, 13(22).
- [Li and Racine, 2023] Li, Q. and Racine, J. S. (2023). *Nonparametric econometrics: theory and practice*. Princeton University Press.
- [Li and Marlin, 2016] Li, S. C. and Marlin, B. M. (2016). A scalable end-to-end gaussian process adapter for irregularly sampled time series classification. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1804–1812.
- [Li et al., 2013] Li, W., Prasad, S., and Fowler, J. E. (2013). Classification and reconstruction from random projections for hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 51(2):833–843.
- [Li et al., 2022] Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2022). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019.
- [Lillesand et al., 2015] Lillesand, T., Kiefer, R. W., and Chipman, J. (2015). *Remote sensing and image interpretation*. John Wiley & Sons.
- [Lin et al., 2022] Lin, C., Zhong, L., Song, X.-P., Dong, J., Lobell, D. B., and Jin, Z. (2022). Early- and in-season crop type mapping without current-year ground truth: Generating labels from historical information via a topology-based approach. *Remote Sensing of Environment*, 274:112994.
- [Liu et al., 2018a] Liu, B., Yu, X., Yu, A., Zhang, P., and Wan, G. (2018a). Spectral-spatial classification of hyperspectral imagery based on recurrent neural networks. *Remote Sensing Letters*, 9(12):1118–1127.

- [Liu et al., 2018b] Liu, H., Cai, J., and Ong, Y.-S. (2018b). Remarks on multi-output gaussian process regression. *Knowledge-Based Systems*, 144:102–121.
- [Liu et al., 2020] Liu, H., Ong, Y.-S., Shen, X., and Cai, J. (2020). When gaussian process meets big data: a review of scalable gps. *IEEE Transactions on Neural Networks and Learning Systems*, 31(11):4405–4423.
- [Liu et al., 2022] Liu, H., Ding, J., Xie, X., Jiang, X., Zhao, Y., and Wang, X. (2022). Scalable multi-task gaussian processes with neural embedding of coregionalization. *Knowledge-Based Systems*, 247:108775.
- [Liu et al., 2018c] Liu, R., Lehman, J., Molino, P., Such, F. P., Frank, E., Sergeev, A., and Yosinski, J. (2018c). An intriguing failing of convolutional neural networks and the coordconv solution. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9628–9639.
- [Lockhart and Wiseman, 2014] Lockhart, J. and Wiseman, A. (2014). *Introduction to Crop Husbandry: (Including Grassland)*. Pergamon international library of science, technology, engineering, and social studies. Elsevier Science.
- [Lomelí-Huerta et al., 2021] Lomelí-Huerta, R., Avila-George, H., Rivera-Caicedo, J. P., and De-la Torre, M. (2021). Water Pollution Detection in Acapulco Coasts Using Merged Data from the Sentinel-2 and Sentinel-3 Satellites. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 1518–1521.
- [Long et al., 2015] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3431–3440. IEEE Computer Society.
- [Longépé et al., 2011] Longépé, N., Rakwatin, P., Isoguchi, O., Shimada, M., Uryu, Y., and Yulianto, K. (2011). Assessment of alos palsar 50 m orthorectified fbd data for regional land cover classification by support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 49:2135–2150.
- [Lu et al., 2014] Lu, L., Di, L., and Ye, Y. (2014). A decision-tree classifier for extracting transparent plastic-mulched landcover from landsat-5 tm images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7:4548–4558.
- [Lu et al., 2007] Lu, X., Liu, R., Liu, J., and Liang, S. (2007). Removal of noise by wavelet method to generate high quality temporal data of terrestrial modis products. *Photogrammetric Engineering and Remote Sensing*, 73:1129–1139.
- [Lucas et al., 2021] Lucas, B., Pelletier, C., Schmidt, D., Webb, G., and Petitjean, F. (2021). A bayesian-inspired, deep learning-based, semi-supervised domain adaptation technique for land cover mapping. *Machine Learning*.

- [L'Heureux et al., 2017] L'Heureux, A., Grolinger, K., Elyamany, H. F., and Capretz, M. A. M. (2017). Machine learning with big data: Challenges and approaches. *IEEE Access*, 5:7776–7797.
- [M Rustowicz et al., 2019] M Rustowicz, R., Cheong, R., Wang, L., Ermon, S., Burke, M., and Lobell, D. (2019). Semantic segmentation of crop type in africa: A novel dataset and analysis of deep learning methods. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.
- [Ma et al., 2017] Ma, L., Li, M., Ma, X., Cheng, L., Du, P., and Liu, Y. (2017). A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130:277–293.
- [Ma et al., 2019] Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., and Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS journal of photogrammetry and remote sensing*, 152:166–177.
- [MacKay, 1996] MacKay, D. J. C. (1996). *Bayesian Non-Linear Modeling for the Prediction Competition*. Springer Netherlands, Dordrecht.
- [Mallet and Le Bris, 2020] Mallet, C. and Le Bris, A. (2020). Current challenges in operational very high resolution land-cover mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2020:703–710.
- [Mansor et al., 2002] Mansor, S., Hong, W. T., and Shariff, A. R. M. (2002). Object oriented classification for land cover mapping. *Proceedings of map Asia*, pages 7–9.
- [Martínez-Ferrer et al., 2021] Martínez-Ferrer, L., Piles, M., and Camps-Valls, G. (2021). Crop yield estimation and interpretability with gaussian processes. *IEEE Geoscience and Remote Sensing Letters*, 18(12):2043–2047.
- [Mateo-Sanchis et al., 2018] Mateo-Sanchis, A., Munoz-Mari, J., Campos-Taberner, M., Garcia-Haro, J., and Camps-Valls, G. (2018). Gap Filling of Biophysical Parameter Time Series with Multi-Output Gaussian Processes. In *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE.
- [Matthews et al., 2017] Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrà, P., Ghahramani, Z., and Hensman, J. (2017). GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18(40):1–6.
- [Maugeais et al., 2011] Maugeais, E., Lecordix, F., Halbecq, X., and Braun, A. (2011). Dérivation cartographique multi échelles de la BDTopo de l'IGN France: mise en oeuvre du processus de production de la Nouvelle Carte de Base. In *Proc 25th Int Cartogr Conf Paris*, pages 3–8.
- [Maus et al., 2016] Maus, V., Câmara, G., Cartaxo, R., Sanchez, A., Ramos, F. M., and de Queiroz, G. R. (2016). A time-weighted dynamic time warping method for land-use and land-cover mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(8):3729–3739.

- [Maus et al., 2019] Maus, V., Câmara, G., Appel, M., and Pebesma, E. (2019). dtwsat: Time-weighted dynamic time warping for satellite image time series analysis in r. *Journal of Statistical Software*, 88(5):1–31.
- [Max, 1950] Max, A. W. (1950). Inverting modified matrices. In *Memorandum Rept. 42, Statistical Research Group*, page 4. Princeton Univ.
- [Maxwell et al., 2018] Maxwell, A. E., Warner, T. A., and Fang, F. (2018). Implementation of machine-learning classification in remote sensing: An applied review. *International journal of remote sensing*, 39(9):2784–2817.
- [McFEETERS, 1996] McFEETERS, S. K. (1996). The use of the normalized difference water index (ndwi) in the delineation of open water features. *International Journal of Remote Sensing*, 17(7):1425–1432.
- [Medsker and Jain, 2001] Medsker, L. R. and Jain, L. (2001). Recurrent neural networks. *Design and Applications*, 5(64-67):2.
- [Meerdink et al., 2019] Meerdink, S. K., Hook, S. J., Roberts, D. A., and Abbott, E. A. (2019). The ecostress spectral library version 1.0. *Remote Sensing of Environment*, 230:111196.
- [Melgani and Bruzzone, 2004] Melgani, F. and Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1778–1790.
- [Menon et al., 2016] Menon, V., Du, Q., and Fowler, J. E. (2016). Hadamard-walsh random projection for hyperspectral image classification. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5141–5144.
- [Millennium ecosystem assessment, 2005] Millennium ecosystem assessment, M. (2005). *Ecosystems and human well-being: synthesis*. Island Press, Washington, DC, Washington, DC. OCLC: ocm59279709.
- [Miller and Colwell, 1961] Miller, C. I. and Colwell, R. N. (1961). Manual of photographic interpretation. *Journal of Wildlife Management*.
- [Minka, 2001] Minka, T. P. (2001). Expectation Propagation for Approximate Bayesian Inference. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence, UAI'01*, pages 362–369, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- [Moeini Rad et al., 2019] Moeini Rad, A., Ashourloo, D., Salehi Shahrabi, H., and Nematollahi, H. (2019). Developing an Automatic Phenology-Based Algorithm for Rice Detection Using Sentinel-2 Time-Series Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(5):1471–1481.
- [Monserud and Leemans, 1992] Monserud, R. A. and Leemans, R. (1992). Comparing global vegetation maps with the kappa statistic. *Ecological modelling*, 62(4):275–293.
- [Montanaro et al., 2022] Montanaro, A., Valsesia, D., Fracastoro, G., and Magli, E. (2022). Semi-supervised learning for joint sar and multispectral land cover classification. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5.

- [Montero et al., 2014] Montero, E., Van Wolvelaer, J., and Garzón, A. (2014). *The European Urban Atlas*. Springer Netherlands.
- [Moraes et al., 2021] Moraes, D., Benevides, P., Costa, H., Moreira, F. D., and Caetano, M. (2021). Assessment of the introduction of spatial stratification and manual training in automatic supervised image classification. In *Earth Resources and Environmental Remote Sensing/GIS Applications XII*, volume 11863, pages 291–298. SPIE.
- [Morales-Alvarez et al., 2018] Morales-Alvarez, P., Perez-Suay, A., Molina, R., and Camps-Valls, G. (2018). Remote Sensing Image Classification with Large Scale Gaussian Processes. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2):1103–1114.
- [Moreno et al., 2014] Moreno, Á., García-Haro, F. J., Martínez, B., and Gilabert, M. A. (2014). Noise reduction and gap filling of fapar time series using an adapted local regression filter. *Remote Sensing*, 6(9):8238–8260.
- [Moreno-Muñoz et al., 2018] Moreno-Muñoz, P., Artés-Rodríguez, A., and Álvarez, M. A. (2018). Heterogeneous multi-output gaussian process prediction. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6712–6721.
- [Mountrakis et al., 2011] Mountrakis, G., Im, J., and Ogole, C. (2011). Support vector machines in remote sensing: A review. *ISPRS journal of photogrammetry and remote sensing*, 66(3):247–259.
- [Mouret et al., 2022] Mouret, F., Albughdadi, M., Duthoit, S., Kouamé, D., Rieu, G., and Tourneret, J.-Y. (2022). Reconstruction of sentinel-2 derived time series using robust gaussian mixture models – application to the detection of anomalous crop development. *Computers and Electronics in Agriculture*, 198:106983.
- [Munoz-Mari et al., 2009] Munoz-Mari, J., Gómez-Chova, L., Martínez-Ramón, M., Rojo-Alvarez, J. L., Calpe-Maravilla, J., and Camps-Valls, G. (2009). Multi-temporal image classification with kernels. *Kernel Methods for Remote Sensing Data Analysis*, 125.
- [Neal, 1997] Neal, R. M. (1997). Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification. *arXiv: Data Analysis, Statistics and Probability*.
- [Neil et al., 2016] Neil, D., Pfeiffer, M., and Liu, S. (2016). Phased LSTM: accelerating recurrent network training for long or event-based sequences. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3882–3890.
- [Nelson, 2005] Nelson, G. C. (2005). Drivers of ecosystem change: summary chapter. *Ecosystems*.
- [Nguyen et al., 2020] Nguyen, L. H., Joshi, D. R., Clay, D. E., and Henebry, G. M. (2020). Characterizing land cover/land use from multiple years of landsat and modis time series: A

- novel approach using land surface phenology modeling and random forest classifier. *Remote Sensing of Environment*, 238:111017. Time Series Analysis with High Spatial Resolution Imagery.
- [Nickisch and Rasmussen, 2008] Nickisch, H. and Rasmussen, C. (2008). Approximations for binary gaussian process classification. *Journal of Machine Learning Research*, 9:2035–2078.
- [Niroumand-Jadidi and Bovolo, 2023] Niroumand-Jadidi, M. and Bovolo, F. (2023). Deep-learning-based retrieval of an orange band sensitive to cyanobacteria for landsat-8/9 and sentinel-2. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:3929–3937.
- [Nitze et al., 2015] Nitze, I., Barrett, B., and Cawkwell, F. (2015). Temporal optimisation of image acquisition for land cover classification with random forest and modis time-series. *International Journal of Applied Earth Observation and Geoinformation*, 34:136–146.
- [Nyborg et al., 2022] Nyborg, J., Pelletier, C., and Assent, I. (2022). Generalized classification of satellite image time series with thermal positional encoding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022*, pages 1391–1401. IEEE.
- [Opgenoorth and Faith, 2013] Opgenoorth, L. and Faith, D. (2013). The intergovernmental science-policy platform on biodiversity and ecosystem services (ipbes), up and walking. *Frontiers of Biogeography*, 5:207–211.
- [Oruc et al., 2004] Oruc, M., Marangoz, A., and Buyuksalih, G. (2004). Comparison of pixel-based and object-oriented classification approaches using landsat-7 etm spectral bands.
- [Otukey and Blaschke, 2010] Otukey, J. and Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation*, 12:S27–S31. Supplement Issue on "Remote Sensing for Africa – A Special Collection from the African Association for Remote Sensing of the Environment (AARSE)".
- [Paciorek and Schervish, 2006] Paciorek, C. J. and Schervish, M. J. (2006). Spatial modelling using a new class of nonstationary covariance functions. *Environmetrics*, 17(5):483–506.
- [Pal and Mather, 2003] Pal, M. and Mather, P. M. (2003). An assessment of the effectiveness of decision tree methods for land cover classification. *Remote Sensing of Environment*, 86(4):554–565.
- [Pal, 2009] Pal, M. (2009). Kernel methods in remote sensing: a review. *ISH Journal of Hydraulic Engineering*, 15(sup1):194–215.
- [Parker, 2012] Parker, C. (2012). Unexpected challenges in large scale machine learning. In *Proceedings of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications*, BigMine '12, page 1–6, New York, NY, USA. Association for Computing Machinery.

- [Pasolli et al., 2010] Pasolli, L., Melgani, F., and Blanzieri, E. (2010). Gaussian Process Regression for Estimating Chlorophyll Concentration in Subsurface Waters From Remote Sensing Data. *IEEE Geoscience and Remote Sensing Letters*, 7(3):464–468.
- [Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E. B., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.
- [Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- [Pelletier et al., 2016] Pelletier, C., Valero, S., Inglada, J., Champion, N., and Dedieu, G. (2016). Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas. *Remote Sensing of Environment*, 187:156–168.
- [Pelletier et al., 2017] Pelletier, C., Valero, S., Inglada, J., Champion, N., Marais Sicre, C., and Dedieu, G. (2017). Effect of training class label noise on classification performances for land cover mapping with satellite image time series. *Remote Sensing*, 9(2):173.
- [Pelletier et al., 2019] Pelletier, C., Webb, G., and Petitjean, F. (2019). Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series. *Remote Sensing*, 11(5):523.
- [Petitjean et al., 2012] Petitjean, F., Inglada, J., and Gancarski, P. (2012). Satellite image time series analysis under time warping. *IEEE Transactions on Geoscience and Remote Sensing*, 50(8):3081–3095.
- [Pfeffer et al., 2014] Pfeffer, W. T., Arendt, A. A., Bliss, A., Bolch, T., Cogley, J. G., Gardner, A. S., Hagen, J.-O., Hock, R., Kaser, G., Kienholz, C., Miles, E. S., Moholdt, G., Mölg, N., Paul, F., Radić, V., Rastner, P., Raup, B. H., Rich, J., Sharp, M. J., and The Randolph Consortium (2014). The Randolph Glacier Inventory: a globally complete inventory of glaciers. *Journal of Glaciology*, 60(221):537–552.
- [Phan et al., 2020] Phan, T.-N., Kuch, V., and Lehnert, L. (2020). Land cover classification using google earth engine and random forest classifier - the role of image composition. *Remote Sensing*.
- [Pipia et al., 2019] Pipia, L., Muñoz-Marí, J., Amin, E., Belda, S., Camps-Valls, G., and Verrelst, J. (2019). Fusing optical and sar time series for lai gap filling with multioutput gaussian processes. *Remote Sensing of Environment*, 235:111452.

- [Poggio et al., 2012] Poggio, L., Gimona, A., and Brown, I. (2012). Spatio-temporal modis evi gap filling under cloud cover: An example in scotland. *ISPRS Journal of Photogrammetry and Remote Sensing*, 72:56–72.
- [Pontius Jr and Millones, 2011] Pontius Jr, R. G. and Millones, M. (2011). Death to kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment. *International Journal of Remote Sensing*, 32(15):4407–4429.
- [Probeck et al., 2021] Probeck, M., Ruiz, I., Ramminger, G., Fourie, C., Maier, P., Ickerott, M., Storch, C., Homolka, A., Muller, S. J., Tiwari, H., Stumpf, A., Chun, S., Mattos, C., Lindmayer, A., Jahangir, F., Endara, P., Berndt, F., Dohr, M., Kapferer, W., Schleicher, C., Ralsler, S., Innerbichler, F., Riffler, M., Siklar, M., Aifantopoulou, D., Paralykidis, S., Pinet, C., Jaffrain, G., di Federico, A., Corsi, M., Langanke, T., and Dufourmont, H. (2021). Clc+ backbone: Set the scene in copernicus for the coming decade. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 2076–2079.
- [Probst et al., 2019] Probst, P., Wright, M. N., and Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3):e1301.
- [Pérez-Suay et al., 2017] Pérez-Suay, A., Amorós-López, J., Gómez-Chova, L., Laparra, V., Muñoz-Marí, J., and Camps-Valls, G. (2017). Randomized kernels for large scale earth observation applications. *Remote Sensing of Environment*, 202:54–63. Big Remotely Sensed Data: tools, applications and experiences.
- [Quiñonero-Candela and Rasmussen, 2005] Quiñonero-Candela, J. and Rasmussen, C. E. (2005). A Unifying View of Sparse Approximate Gaussian Process Regression. *J. Mach. Learn. Res.*, 6:1939–1959.
- [Ranganathan et al., 2017] Ranganathan, P., Pramesh, C., and Aggarwal, R. (2017). Common pitfalls in statistical analysis: Measures of agreement. *Perspectives in clinical research*, 8(4):187.
- [Rao, 1948] Rao, C. R. (1948). The utilization of multiple measurements in problems of biological classification. *Journal of the Royal Statistical Society. Series B (Methodological)*, 10(2):159–203.
- [Rapinel et al., 2018] Rapinel, S., Clément, B., Dufour, S., and Hubert-Moy, L. (2018). Fine-scale monitoring of long-term wetland loss using lidar data and historical aerial photographs: The example of the couesnon floodplain, france. *Wetlands*, 38:423–435.
- [Rasmussen and Williams, 2005] Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.
- [Ratajczak et al., 2018] Ratajczak, Z., Carpenter, S. R., Ives, A. R., Kucharik, C. J., Ramiadantsoa, T., Stegner, M. A., Williams, J. W., Zhang, J., and Turner, M. G. (2018). Abrupt change in ecological systems: Inference and diagnosis. *Trends in Ecology & Evolution*, 33(7):513–526.

- [Ren et al., 2021] Ren, B., Zhao, Y., Hou, B., Chanussot, J., and Jiao, L. (2021). A mutual information-based self-supervised learning model for polar land cover classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11):9224–9237.
- [Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.
- [Rodriguez-Galiano et al., 2012a] Rodriguez-Galiano, V., Chica-Olmo, M., Abarca-Hernandez, F., Atkinson, P. M., and Jeganathan, C. (2012a). Random forest classification of mediterranean land cover using multi-seasonal imagery and multi-seasonal texture. *Remote Sensing of Environment*, 121:93–107.
- [Rodriguez-Galiano et al., 2012b] Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., and Rigol-Sanchez, J. P. (2012b). An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS journal of photogrammetry and remote sensing*, 67:93–104.
- [Romanov et al., 2000] Romanov, P., Gutman, G., and Csiszar, I. (2000). Automated monitoring of snow cover over north america with multispectral satellite data. *Journal of Applied Meteorology*, 39(11):1866–1880.
- [Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*, pages 234–241. Springer.
- [Roujean et al., 2021] Roujean, J.-L., Bhattacharya, B., Gamet, P., Pandya, M. R., Boulet, G., Olios, A., Singh, S. K., Shukla, M. V., Mishra, M., Babu, S., Raju, P. V., Murthy, C. S., Briottet, X., Rodler, A., Autret, E., Dadou, I., Adlakha, D., Sarkar, M., Picard, G., Kouraev, A., Ferrari, C., Irvine, M., Delogu, E., Vidal, T., Hagolle, O., Maisongrande, P., Sekhar, M., and Mallick, K. (2021). Trishna: An indo-french space mission to study the thermography of the earth at fine spatio-temporal resolution. In *2021 IEEE International India Geoscience and Remote Sensing Symposium (InGARSS)*, pages 49–52.
- [Rouse et al., 1974] Rouse, J. W., Jr., Haas, R. H., Schell, J. A., and Deering, D. W. (1974). Monitoring Vegetation Systems in the Great Plains with ERTS. In *NASA Special Publication*, volume 351, page 309.
- [Rouse Jr et al., 1974] Rouse Jr, J. W., Haas, R. H., Deering, D., Schell, J., and Harlan, J. C. (1974). Monitoring the vernal advancement and retrogradation (green wave effect) of natural vegetation. Technical report.
- [Ruan et al., 2017] Ruan, W., Milstein, A. B., Blackwell, W., and Miller, E. L. (2017). Multiple output gaussian process regression algorithm for multi-frequency scattered data interpolation. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3992–3995.
- [Rulloni et al., 2012] Rulloni, V., Bustos, O., and Flesia, A. G. (2012). Large gap imputation in remote sensed imagery of the environment. *Computational Statistics & Data Analysis*, 56(8):2388–2403.

- [Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Representations by Back-propagating Errors. *Nature*, 323(6088):533–536.
- [Russell, 2010] Russell, S. J. (2010). *Artificial intelligence a modern approach*. Pearson Education, Inc.
- [Rußwurm and Körner, 2017] Rußwurm, M. and Körner, M. (2017). Temporal vegetation modelling using long short-term memory networks for crop identification from medium-resolution multi-spectral satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.
- [Rußwurm and Körner, 2018a] Rußwurm, M. and Körner, M. (2018a). Convolutional LSTMs for Cloud-Robust Segmentation of Remote Sensing Imagery. *ArXiv preprint*, abs/1811.02471.
- [Rußwurm and Körner, 2018b] Rußwurm, M. and Körner, M. (2018b). Multi-temporal land cover classification with sequential recurrent encoders. *ISPRS International Journal of Geo-Information*, 7(4).
- [Rußwurm and Körner, 2020] Rußwurm, M. and Körner, M. (2020). Self-attention for raw optical satellite time series classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:421–435.
- [Rußwurm et al., 2023a] Rußwurm, M., Courty, N., Emonet, R., Lefèvre, S., Tuia, D., and Tave-nard, R. (2023a). End-to-end learned early classification of time series for in-season crop type mapping. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:445–456.
- [Rußwurm et al., 2023b] Rußwurm, M., Klemmer, K., Rolf, E., Zbinden, R., and Tuia, D. (2023b). Geographic location encoding with spherical harmonics and sinusoidal representation networks. *arXiv preprint arXiv:2310.06743*.
- [Saah et al., 2020] Saah, D., Tenneson, K., Poortinga, A., Nguyen, Q., Chishtie, F., San Aung, K., Markert, K. N., Clinton, N., Anderson, E. R., Cutter, P., et al. (2020). Primitives as building blocks for constructing land cover maps. *International Journal of Applied Earth Observation and Geoinformation*, 85:101979.
- [Saatci, 2011] Saatci, Y. (2011). *Scalable Inference for Structured Gaussian Process Models*. Ph.D. dissertation, University of Cambridge.
- [Samrat et al., 2022] Samrat, N., Islam, N., and Haque, A. (2022). Forest land cover changes and its socio-economic consequences on south-eastern part of bangladesh. *Journal of Sustainable Forestry*, 42:1–19.
- [Sasaki et al., 2007] Sasaki, Y. et al. (2007). The truth of the f-measure. *Teach tutor mater*, 1(5):1–5.
- [Scharlemann et al., 2008] Scharlemann, J. P., Benz, D., Hay, S. I., Purse, B. V., Tatem, A. J., Wint, G. W., and Rogers, D. J. (2008). Global data for ecology and epidemiology: a novel algorithm for temporal fourier processing modis data. *PloS one*, 3(1):e1408.

- [Scheibenreif et al., 2022] Scheibenreif, L., Hanna, J., Mommert, M., and Borth, D. (2022). Self-supervised vision transformers for land-cover segmentation and classification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022*, pages 1421–1430. IEEE.
- [Scholkopf and Smola, 2001] Scholkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.
- [Seeger, 2003] Seeger, M. (2003). *Bayesian Gaussian Process Models: PAC-Bayesian Generalization Error Bounds and Sparse Approximation*. PhD thesis, University of Edinburgh.
- [Seeger et al., 2003] Seeger, M. W., Williams, C. K. I., and Lawrence, N. D. (2003). Fast forward selection to speed up sparse gaussian process regression. In Bishop, C. M. and Frey, B. J., editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, volume R4 of *Proceedings of Machine Learning Research*, pages 254–261. PMLR. Reissued by PMLR on 01 April 2021.
- [Sharma et al., 2018] Sharma, A., Liu, X., and Yang, X. (2018). Land cover classification from multi-temporal, multi-spectral remotely sensed imagery using patch-based recurrent neural networks. *Neural Networks*, 105:346–355.
- [Shen et al., 2015] Shen, H., Li, X., Cheng, Q., Zeng, C., Yang, G., Li, H., and Zhang, L. (2015). Missing information reconstruction of remote sensing data: A technical review. *IEEE Geoscience and Remote Sensing Magazine*, 3(3):61–85.
- [Shih et al., 2019] Shih, H.-c., Stow, D. A., and Tsai, Y. H. (2019). Guidance on and comparison of machine learning classifiers for landsat-based land cover and land use mapping. *International Journal of Remote Sensing*, 40(4):1248–1274.
- [Shukla and Marlin, 2021] Shukla, S. N. and Marlin, B. M. (2021). Multi-time attention networks for irregularly sampled time series. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- [Smith et al., 2000] Smith, R., Shiel, R., Millward, D., and Corkhill, P. (2000). The interactive effects of management on the productivity and plant community structure of an upland meadow: an 8-year field trial. *Journal of Applied Ecology*, 37(6):1029–1043.
- [Snelson and Ghahramani, 2005] Snelson, E. and Ghahramani, Z. (2005). Sparse gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems 18 [Neural Information Processing Systems, NIPS 2005, December 5-8, 2005, Vancouver, British Columbia, Canada]*, pages 1257–1264.
- [Snelson and Ghahramani, 2007] Snelson, E. and Ghahramani, Z. (2007). Local and global sparse gaussian process approximations. *Journal of Machine Learning Research - Proceedings Track*, 2:524–531.
- [Soille et al., 2018] Soille, P., Burger, A., De Marchi, D., Kempeneers, P., Rodriguez, D., Syrris, V., and Vasilev, V. (2018). A versatile data-intensive computing platform for information retrieval from big geospatial data. *Future Generation Computer Systems*, 81:30–40.

- [Solano-Correa et al., 2022] Solano-Correa, Y. T., Meshkini, K., Bovolo, F., and Bruzzone, L. (2022). Automatic large-scale precise mapping and monitoring of agricultural fields at country level with sentinel-2 sats. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3131–3145.
- [Song, 2019] Song, W. (2019). Mapping cropland abandonment in mountainous areas using an annual land-use trajectory approach. *Sustainability*, 11(21).
- [Stoian et al., 2019a] Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019a). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17).
- [Stoian et al., 2019b] Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019b). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1986.
- [Sun and Schulz, 2015] Sun, L. and Schulz, K. (2015). The improvement of land cover classification by thermal remote sensing. *Remote Sensing*, 7(7):8368–8390.
- [Sun et al., 2015] Sun, S., Zhong, P., Xiao, H., and Wang, R. (2015). Active Learning With Gaussian Process Classifier for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4):1746–1760.
- [Svendsen et al., 2020] Svendsen, D. H., Morales-Álvarez, P., Ruescas, A. B., Molina, R., and Camps-Valls, G. (2020). Deep gaussian processes for biogeophysical parameter retrieval and model inversion. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166:68–81.
- [Tarantino et al., 2018] Tarantino, C., Adamo, M., Lucas, R., and Blonda, P. (2018). Change Detection in (Semi-) Natural Grassland Ecosystems for Biodiversity Monitoring Using Open Data. In *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 8981–8984.
- [Tarasiou et al., 2023] Tarasiou, M., Chavez, E., and Zafeiriou, S. (2023). Vits for sats: Vision transformers for satellite image time series. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10418–10428.
- [Tatem et al., 2008] Tatem, A. J., Goetz, S. J., and Hay, S. I. (2008). Fifty years of earth observation satellites: Views from above have lead to countless advances on the ground in both scientific knowledge and daily life. *American scientist*, 96(5):390.
- [Teh et al., 2005] Teh, Y. W., Seeger, M., and Jordan, M. I. (2005). Semiparametric latent factor models. In Cowell, R. G. and Ghahramani, Z., editors, *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, volume R5 of *Proceedings of Machine Learning Research*, pages 333–340. PMLR. Reissued by PMLR on 30 March 2021.
- [Terrell and Scott, 1992] Terrell, G. R. and Scott, D. W. (1992). Variable Kernel Density Estimation. *The Annals of Statistics*, 20(3):1236 – 1265.

- [The-GPyOpt-authors, 2016] The-GPyOpt-authors (2016). Gpyopt: A bayesian optimization framework in python. <http://github.com/SheffieldML/GPyOpt>.
- [Thonfeld et al., 2020] Thonfeld, F., Steinbach, S., Muro, J., and Kirimi, F. (2020). Long-term land use/land cover change assessment of the kilombo catchment in tanzania using random forest classification and robust change vector analysis. *Remote Sensing*, 12(7).
- [Titsias, 2009] Titsias, M. (2009). Variational learning of inducing variables in sparse gaussian processes. In van Dyk, D. and Welling, M., editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA. PMLR.
- [Tran et al., 2021] Tran, G.-L., Milios, D., Michiardi, P., and Filippone, M. (2021). Sparse within sparse gaussian processes using neighbor information. In *International Conference on Machine Learning*, pages 10369–10378. PMLR.
- [Tsiafouli et al., 2015] Tsiafouli, M. A., Thébault, E., Sgardelis, S. P., De Ruiter, P. C., Van Der Putten, W. H., Birkhofer, K., Hemerik, L., De Vries, F. T., Bardgett, R. D., Brady, M. V., et al. (2015). Intensive agriculture reduces soil biodiversity across europe. *Global change biology*, 21(2):973–985.
- [Valero et al., 2016] Valero, S., Morin, D., Inglada, J., Sepulcre, G., Arias, M., Hagolle, O., Dedieu, G., Bontemps, S., Defourny, P., and Koetz, B. (2016). Production of a dynamic cropland mask by processing remote sensing image series at high temporal and spatial resolutions. *Remote Sensing*, 8(1).
- [Vali et al., 2020] Vali, A., Comai, S., and Matteucci, M. (2020). Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review. *Remote Sensing*, 12(15).
- [van der Wilk et al., 2017] van der Wilk, M., Rasmussen, C. E., and Hensman, J. (2017). Convolutional gaussian processes. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2849–2858.
- [van der Wilk et al., 2020] van der Wilk, M., Dutordoir, V., John, S., Artemev, A., Adam, V., and Hensman, J. (2020). A Framework for Interdomain and Multioutput Gaussian Processes.
- [Van Engelen and Hoos, 2020] Van Engelen, J. E. and Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine learning*, 109(2):373–440.
- [Vanbergen, 2013] Vanbergen, A. J. (2013). Threats to an ecosystem service: pressures on pollinators. *Frontiers in Ecology and the Environment*, 11(5):251–259.
- [Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R.,

- editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.
- [Verrelst et al., 2011] Verrelst, J., Alonso, L., Camps-Valls, G., Delegido, J., and Moreno, J. (2011). Retrieval of vegetation biophysical parameters using gaussian process techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 50(5):1832–1843.
- [Verrelst et al., 2012a] Verrelst, J., Muñoz, J., Alonso, L., Delegido, J., Rivera, J. P., Camps-Valls, G., and Moreno, J. (2012a). Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for Sentinel-2 and -3. *Remote Sensing of Environment*, 118:127–139.
- [Verrelst et al., 2012b] Verrelst, J., Alonso, L., Camps-Valls, G., Delegido, J., and Moreno, J. (2012b). Retrieval of Vegetation Biophysical Parameters Using Gaussian Process Techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 50(5):1832–1843.
- [Verrelst et al., 2013] Verrelst, J., Rivera, J. P., Moreno, J., and Camps-Valls, G. (2013). Gaussian processes uncertainty estimates in experimental Sentinel-2 LAI and leaf chlorophyll content retrieval. *ISPRS Journal of Photogrammetry and Remote Sensing*, 86:157–167.
- [Villacampa-Calvo et al., 2021] Villacampa-Calvo, C., Zaldivar, B., Garrido-Merchán, E. C., and Hernández-Lobato, D. (2021). Multi-class gaussian process classification with noisy inputs. *J. Mach. Learn. Res.*, 22:36:1–36:52.
- [Vintsyuk, 1968] Vintsyuk, T. K. (1968). Speech discrimination by dynamic programming. *Cybernetics*, 4:52–57.
- [Viovy et al., 1992] Viovy, N., Arino, O., and Belward, A. (1992). The best index slope extraction (bise): A method for reducing noise in ndvi time-series. *International Journal of remote sensing*, 13(8):1585–1590.
- [Vuolo et al., 2018] Vuolo, F., Neuwirth, M., Immitzer, M., Atzberger, C., and Ng, W.-T. (2018). How much does multi-temporal sentinel-2 data improve crop type classification? *International Journal of Applied Earth Observation and Geoinformation*, 72:122–130.
- [Wang et al., 2020] Wang, M., Fu, W., He, X., Hao, S., and Wu, X. (2020). A survey on large-scale machine learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(6):2574–2594.
- [Wang et al., 2022] Wang, Y., Albrecht, C. M., Braham, N. A. A., Mou, L., and Zhu, X. X. (2022). Self-supervised learning in remote sensing: A review. *IEEE Geoscience and Remote Sensing Magazine*, 10(4):213–247.
- [Wen et al., 2023] Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., and Sun, L. (2023). Transformers in Time Series: A Survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 6778–6786, Macau, SAR China. International Joint Conferences on Artificial Intelligence Organization.

- [Whiteside et al., 2011] Whiteside, T. G., Boggs, G. S., and Maier, S. W. (2011). Comparing object-based and pixel-based classifications for mapping savannas. *International Journal of Applied Earth Observation and Geoinformation*, 13(6):884–893.
- [Wilcoxon, 1945] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.
- [Willhauck et al., 2000] Willhauck, G., Schneider, T., De Kok, R., and Ammer, U. (2000). Comparison of object oriented classification techniques and standard image analysis for the use of change detection between spot multispectral satellite images and aerial photos. In *Proceedings of XIX ISPRS congress*, volume 33, pages 35–42. Amsterdam: IAPRS.
- [Williams and Barber, 1998] Williams, C. and Barber, D. (1998). Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351.
- [Williams and Seeger, 2000] Williams, C. K. I. and Seeger, M. W. (2000). Using the nyström method to speed up kernel machines. In Leen, T. K., Dietterich, T. G., and Tresp, V., editors, *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pages 682–688. MIT Press.
- [Wilson et al., 2014] Wilson, A. G., Gilboa, E., Cunningham, J. P., and Nehorai, A. (2014). Fast kernel learning for multidimensional pattern extrapolation. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3626–3634.
- [Wilson et al., 2016] Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. (2016). Stochastic variational deep kernel learning. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2586–2594.
- [Woodcock et al., 1988] Woodcock, C. E., Strahler, A. H., and Jupp, D. L. (1988). The use of variograms in remote sensing: I. scene models and simulated images. *Remote Sensing of Environment*, 25(3):323–348.
- [World Meteorological Organization (WMO); United Nations Educational and (ISC), 2022] World Meteorological Organization (WMO); United Nations Educational, S. and (ISC), C. O. U. I. O. C. I. U. N. E. P. U. . I. S. C. (2022). The 2022 GCOS ECVs Requirements (GCOS 245). Technical report, Geneva.
- [Wu et al., 2021] Wu, L., Miller, A., Anderson, L., Pleiss, G., Blei, D., and Cunningham, J. (2021). Hierarchical inducing point gaussian process for inter-domain observations. In *International Conference on Artificial Intelligence and Statistics*, pages 2926–2934. PMLR.
- [Wu et al., 2022] Wu, L., Pleiss, G., and Cunningham, J. P. (2022). Variational nearest neighbor gaussian process. In *International Conference on Machine Learning*, pages 24114–24130. PMLR.

- [Xie et al., 2019] Xie, S., Liu, L., Zhang, X., Yang, J., Chen, X., and Gao, Y. (2019). Automatic land-cover mapping using landsat time-series data based on google earth engine. *Remote Sensing*, 11(24).
- [Xu et al., 2019] Xu, M., Liu, H., Beck, R., Lekki, J., Yang, B., Shu, S., Liu, Y., Benko, T., Anderson, R., Tokars, R., Johansen, R., Emery, E., and Reif, M. (2019). Regionally and locally adaptive models for retrieving chlorophyll-a concentration in inland waters from remotely sensed multispectral and hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7):4758–4774.
- [Xue et al., 2017] Xue, J., Su, B., et al. (2017). Significant remote sensing vegetation indices: A review of developments and applications. *Journal of sensors*, 2017.
- [Yang et al., 2015a] Yang, G., Shen, H., Zhang, L., He, Z., and Li, X. (2015a). A moving weighted harmonic analysis method for reconstructing high-quality spot vegetation ndvi time-series data. *IEEE Transactions on Geoscience and Remote Sensing*, 53:1–14.
- [Yang and Huang, 2021] Yang, J. and Huang, X. (2021). The 30 m annual land cover dataset and its dynamics in china from 1990 to 2019. *Earth System Science Data*, 13(8):3907–3925.
- [Yang et al., 2015b] Yang, M. Y., Liao, W., Rosenhahn, B., and Zhang, Z. (2015b). Hyperspectral image classification using Gaussian process models. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1717–1720, Milan, Italy. IEEE.
- [Yao et al., 2019] Yao, X., Yang, H., Wu, Y., Wu, P., Wang, B., Zhou, X., and Wang, S. (2019). Land use classification of the deep convolutional neural network method reducing the loss of spatial features. *Sensors*, 19(12).
- [Yuan et al., 2009] Yuan, H., Van Der Wiele, C. F., and Khorram, S. (2009). An automated artificial neural network system for land use/land cover classification from landsat tm imagery. *Remote Sensing*, 1(3):243–265.
- [Yuan et al., 2022] Yuan, Y., Lin, L., Liu, Q., Hang, R., and Zhou, Z.-G. (2022). Sits-former: A pre-trained spatio-spectral-temporal representation model for sentinel-2 time series classification. *International Journal of Applied Earth Observation and Geoinformation*, 106:102651.
- [Zanaga et al., 2022] Zanaga, D., Van De Kerchove, R., Daems, D., De Keersmaecker, W., Brockmann, C., Kirches, G., Wevers, J., Cartus, O., Santoro, M., Fritz, S., Lesiv, M., Herold, M., Tsendbazar, N.-E., Xu, P., Ramoino, F., and Arino, O. (2022). Esa worldcover 10 m 2021 v200.
- [Zemp et al., 2022] Zemp, M., Chao, Q., Han Dolman, A. J., Herold, M., Krug, T., Speich, S., Suda, K., Thorne, P., and Yu, W. (2022). GCOS 2022 Implementation Plan. *Global Climate Observing System GCOS*, (244):85.
- [Zhang et al., 2019] Zhang, C., Wei, S., Ji, S., and Lu, M. (2019). Detecting large-scale urban land cover changes from very high resolution remote sensing images using cnn-based classification. *ISPRS International Journal of Geo-Information*, 8(4).

- [Zhang et al., 2010] Zhang, K., Kimball, J. S., Nemani, R. R., and Running, S. W. (2010). A continuous satellite-derived global record of land surface evapotranspiration from 1983 to 2006. *Water Resources Research*, 46(9).
- [Zhang and Yang, 2020] Zhang, K. and Yang, H. (2020). Semi-supervised multi-spectral land cover classification with multi-attention and adaptive kernel. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1881–1885.
- [Zhang et al., 2018] Zhang, X., Sun, Y., Jiang, K., Li, C., Jiao, L., and Zhou, H. (2018). Spatial sequential recurrent neural network for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(11):4141–4155.
- [Zheng et al., 2015] Zheng, B., Myint, S. W., Thenkabail, P. S., and Aggarwal, R. M. (2015). A support vector machine to identify irrigated crop types using time-series landsat ndvi data. *International Journal of Applied Earth Observation and Geoinformation*, 34:103–112.
- [Zhou et al., 2015] Zhou, J., Jia, L., and Menenti, M. (2015). Reconstruction of global modis vegetation index time series: Performance of harmonic analysis of time series (hants). *Remote Sensing of Environment*, 163:217–228.
- [Zhou et al., 2022] Zhou, Q., Wang, S., and Liu, Y. (2022). Exploring the accuracy and completeness patterns of global land-cover/land-use data in openstreetmap. *Applied Geography*, 145:102742.
- [Zhu et al., 2012] Zhu, W., Pan, Y., He, H., Wang, L., Mou, M., and Liu, J. (2012). A changing-weight filter method for reconstructing a high-quality ndvi time series to preserve the integrity of vegetation phenology. *IEEE Transactions on Geoscience and Remote Sensing*, 50(4):1085–1094.
- [Zhu and Woodcock, 2014] Zhu, Z. and Woodcock, C. E. (2014). Continuous change detection and classification of land cover using all available landsat data. *Remote Sensing of Environment*, 144:152–171.

Titre : Intelligence artificielle appliquée aux séries temporelles d'images satellites pour la surveillance des écosystèmes

Mots clés : télédétection, intelligence artificielle, écosystèmes, classification grande échelle, processus Gaussiens, séries temporelles d'images satellites

Résumé : Dans un contexte de changement climatique, la surveillance des écosystèmes est une mission essentielle. En effet, cela permet de mieux comprendre les changements qui peuvent affecter les écosystèmes mais aussi de prendre des décisions en conséquence afin de préserver les générations actuelles et futures. Les cartes d'occupations du sol sont un outil indispensable en fournissant des informations sur les différents types de couverture physique de la surface de la Terre (e.g. forêts, prairies, terres agricoles). Actuellement, un nombre accru de missions satellites fournissent un volume important de données gratuites et librement accessibles. Les séries temporelles d'images satellites (SITS), dont celles de Sentinel-2, notamment grâce à leurs très hautes résolutions, informent sur la dynamique de la végétation. Des algorithmes d'apprentissage automatique permettent de produire de manière fréquente et régulière des cartes d'occupations du sol à partir de SITS. L'objectif de cette thèse est le développement d'algorithmes de classification supervisée pour la production de cartes d'occupations du sol à grande échelle. Dans un contexte opérationnel, quatre principaux défis se dégagent. Le premier concerne le volume important de données que les algorithmes doivent être capables de gérer. Le second est lié à la prise en compte des corrélations entre les variables spectro-temporelles et leur extraction pour la classification. Le troisième, quant à lui, correspond à la prise en compte de la variabilité spatiale: dans des zones géographiques étendues, la donnée n'est pas stationnaire. Enfin, le quatrième défi concerne l'utilisation de SITS irrégulièrement échantillonnées et non alignées, principalement du aux conditions météorologiques (e.g. nuages) ou à des dates d'acquisitions différentes entre deux orbites. Cette thèse est divisée en deux contributions principales. La première contribution concerne la mise en place de processus gaussiens stochastiques variationnels (SVGP) pour des SITS à grande échelle. Des millions d'échantillons peuvent être utilisés pour l'apprentissage, au lieu de quelques milliers pour les processus gaussiens (GP) traditionnels. Des combinaisons de fonctions de covariances ont été mis en place permettant notamment de prendre en compte l'information spatiale et d'être plus robuste vis à vis de la variabilité spatiale. Cependant, les SITS sont ré-échantillonnées linéairement indépendamment de la tâche de classification. La deuxième contribution concerne donc la mise en place d'un ré-échantillonnage optimisé pour la tâche de classification. Un interpolateur à noyau prenant en compte l'information spatiale permet de produire une représentation latente qui est donnée à notre SVGP. Les expérimentations ont été menées avec les SITS de Sentinel-2 pour l'ensemble de l'année 2018 sur une zone d'environ 200 000 km² (environ 2 milliards de pixels) dans le sud de la France (27 tuiles MGRS). Ce dispositif expérimental est représentatif d'un cadre opérationnel. Les résultats obtenus montrent que les modèles issus des deux contributions sont plus performants que la méthode utilisée pour les systèmes opérationnels actuels (i.e. forêts d'arbres aléatoires avec des SITS linéairement ré-échantillonnées utilisant la stratification spatiale).

Title: Artificial Intelligence for Ecosystem Monitoring using Remote Sensing and Digital Agriculture Data

Key words: remote sensing, artificial intelligence, ecosystem, large-scale classification, Gaussian Processes, satellite image time series

Abstract: In the context of climate change, ecosystem monitoring is a crucial task. It allows to better understand the changes that affect them and also enables decision-making to preserve them for current and future generations. Land use and land cover (LULC) maps are an essential tool in ecosystem monitoring providing information on different types of physical cover of the Earth's surface (e.g. forests, grasslands, croplands). Nowadays, an increasing number of satellite missions generate huge amounts of free and open data. In particular, satellite image time series (SITS), such as the ones produced by Sentinel-2, offer high temporal, spectral and spatial resolutions and provide relevant information about vegetation dynamics. Combined with machine learning algorithms, they allow the production of frequent and accurate LULC maps. This thesis is focused on the development of pixel-based supervised classification algorithms for the production of LULC maps at large scale. Four main challenges arise in an operational context. Firstly, unprecedented amounts of data are available and the algorithms need to be adapted accordingly. Secondly, with the improvement in spatial, spectral and temporal resolutions, the algorithms should be able to take into account correlations between the spectro-temporal features to extract meaningful representations for the purpose of classification. Thirdly, in wide geographical coverage, the problem of non-stationarity of the data arises, therefore the algorithms should be able to take into account this spatial variability. Fourthly, because of the different satellite orbits or meteorological conditions, the acquisition times are irregular and unaligned between pixels, thus, the algorithms should be able to work with irregular and unaligned SITS. This work has been divided into two main parts. The first PhD contribution is the development of stochastic variational Gaussian Processes (SVGP) on massive data sets. The proposed Gaussian Processes (GP) model can be trained with millions of samples, compared to few thousands for traditional GP methods. The spatial and spectro-temporal structure of the data is taken into account thanks to the inclusion of the spatial information in bespoke composite covariance functions. Besides, this development enables to take into account the spatial information and thus to be robust to the spatial variability of the data. However, the time series are linearly resampled independently from the classification. Therefore, the second PhD contribution is the development of an end-to-end learning by combining a time and space informed kernel interpolator with the previous SVGP classifier. The interpolator embeds irregular and unaligned SITS onto a fixed and reduced size latent representation. The obtained latent representation is given to the SVGP classifier and all the parameters are jointly optimized w.r.t. the classification task. Experiments were run with Sentinel-2 SITS of the full year 2018 over an area of 200 000 km² (about 2 billion pixels) in the south of France (27 MGRS tiles), which is representative of an operational setting. Results show that both methods (i.e. SVGP classifier with linearly interpolated time series and the spatially kernel interpolator combined with the SVGP classifier) outperform the method used for current operational systems (i.e. Random Forest with linearly interpolated time series using spatial stratification).