Contributions to Random Forests Methods for several Data Analysis Problems

Robin Genuer

▶ **To cite this version:**

Robin Genuer. Contributions to Random Forests Methods for several Data Analysis Problems. Statistics [math.ST]. Université de Bordeaux, 2021. tel-03111020v1

HAL Id: tel-03111020

https://theses.hal.science/tel-03111020v1

Submitted on 15 Jan 2021 (v1), last revised 2 Feb 2021 (v2)

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.

HABILITATION À DIRIGER DES RECHERCHES

Ecole Doctorale Sociétés, Politique, Santé Publique
Option Santé Publique, Biostatistique

Robin GENUER

Maître de conférences

Contributions aux méthodes de forêts aléatoires pour divers problèmes d'analyse de données

Soutenue publiquement le 12 janvier 2021

Membres du jury :

Pierre GEURTS	Professeur	Université de Liège	Rapporteur
Hélène JACQMIN-GADDA	Directrice de Recherche	Inserm, Bordeaux	Examinatrice
Jean-Michel POGGI	Professeur	Université Paris-Saclay	Examinateur
Anne RUIZ-GAZEN	Professeure	Université Toulouse 1	Examinatrice
Adeline SAMSON	Professeure	Université Grenobles-Alpes	Rapportrice
Rodolphe THIEBAUT	Professeur	Université de Bordeaux	Président du jury
Ruoqing ZHU	Assistant professor	Université de l'Illinois	Rapporteur

Remerciements

Mes premiers remerciements vont aux membres du jury qui ont accepté d'évaluer mon travail : merci à Adeline, Pierre et Ruoqing pour leur temps et pour leurs rapports très intéressants ; merci à Anne, Hélène et Jean-Michel d'avoir participé activement à la soutenance ; et merci à Rodolphe d'avoir présidé le jury (ce qui n'a bien sûr pas inhibé sa participation). Malgré des conditions de communication uniquement par visioconférence, j'ai énormément apprécié nos échanges fournis sur de nombreux aspects de mon travail de recherche. Cela permet une mise en perspective et une aide à la réflexion sur de nombreuses directions de recherche, très bénéfiques pour la suite j'en suis convaincu.

Je suis également très reconnaissant envers toutes les personnes qui m'ont envoyé un mot d'encouragement ou de féliciations avant ou après la jour J, et/ou qui se sont connectées pour assister à toute ou partie de la soutenance. C'est d'ailleurs peut-être le seul avantage (alors que plus de points négatifs viennent naturellement à l'esprit) de ces événements organisés en visioconférence : avoir eu un public non exclusivement bordelais m'a fait très plaisir, merci à toutes et à tous.

Cela fait presque dix ans que je travaille à l'institut de santé publique de Bordeaux et je souhaite remercier chaleureusement tous mes collègues des équipes de biostatistique (ancien.e.s et actuel.le.s, titulaires ou non) pour leur accueil et leur inclusion dans les activités de recherche, mais aussi les membres de l'équipe pédagogique pluridisciplinaire du master de santé publique. L'Isped est un endroit où il fait bon travailler, et j'y prends beaucoup de plaisir. Je n'oublie pas mes collègues et collaborateurs des autres sites bordelais avec qui j'ai toujours plaisir à travailler et échanger. Je remercie également les étudiant.e.s du master pour leur intérêt et leur motivation à toute épreuve ; enseigner, dans cette ambiance à la fois sympathique et studieuse, est un des aspects les plus agréables de mon activité. Je les soutiens pleinement dans cette épreuve actuelle, compte tenu des restrictions sanitaires, particulièrement extrêmes pour la population étudiante.

Quelques mentions spéciales : tout d'abord à Jean-Michel pour tout ce que tu m'as apporté, c'est toujours un plaisir de travailler avec toi et l'expérience d'écriture du livre a encore été très enrichissante ; à Rodolphe pour ton soutien sans faille et pour ton dynamisme scientifique hors du commun, travailler dans ton équipe est très stimulant ; à Louis pour t'être lancé dans l'aventure de cette thèse que tu as brillamment soutenue il y a peu, cette première expérience d'encadrement a été pour moi extrêmement positive ; et enfin à El Tonio et Bobo la stat pour la vie de tous les jours au bureau et en dehors (le plus souvent vers l'ouest).

Pour finir je remercie ma famille au sens large, et au sens moins large : mes drôles Aélia, Maïna et Elouen pour les petits et grands plaisirs au quotidien, et Marjolaine pour tout : pour ton soutien, pour ton humour, pour tes attentions, pour tout ce qu'on partage, pour être toi, pour ton amour : oh'ways.

Curriculum Vitae

Parcours professionnel

2011 -	**Maître de Conférences** à l'Université de Bordeaux, ISPED. Membre de l'équipe SISTM des centres INSERM U1219 et INRIA Bordeaux Sud-Ouest.
2010 - 2011	**ATER** à plein temps à l'Université Paris-Descartes.
2007 - 2010	**Thèse de mathématiques** soutenue le 24 novembre 2010 à l'Université Paris-Sud 11 *Forêts aléatoires : aspects théoriques, sélection de variables et applications* Directeur de thèse : Jean-Michel Poggi

Articles publiés ou acceptés

- L. Capitaine, R. Genuer, R. Thiébaut, **Random forests for high-dimensional longitudinal data**, *Statistical Methods in Medical Research*, accepté (2020) DOI
- M. Chavent, R. Genuer, J. Saracco, **Combining clustering of variables and feature selection using random forests**, *Communications in Statistics – Simulation and Computation*, accepté (2018) DOI
- M. Tabue-Teguo, L. Grasset, J.A. Avila-Funes, R. Genuer, C. Proust-Lima, *et al.*, **Prevalence and Co-Occurrence of Geriatric Syndromes in People Aged 75 Years and Older in France: Results From the Bordeaux Three-city Study**, *The journals of gerontology. Series A* 73(1):109-116 (2018) DOI
- L. Zago, P.-Y. Hervé, R. Genuer, A. Laurent, B. Mazoyer, N. Tzourio-Mazoyer, M. Joliot, **Predicting hemispheric dominance for language production in healthy individuals using support vector machine**, *Human Brain Mapping* 38:5871–5889 (2018) DOI
- R. Genuer, J.-M. Poggi, C. Tuleau-Malot, N. Villa-Vialaneix, **Random Forests for Big Data**. *Big Data Research*, 9:28-46 (2017) DOI
- S. Arlot, R. Genuer. **Comments on: « A Random Forest Guided Tour » by G. Biau and E. Scornet**, *TEST* 25(2):228-238 (2016) DOI

- R. Genuer, J.-M. Poggi, C. Tuleau-Malot, **VSURF: An R Package for Variable Selection Using Random Forests**, *The R Journal* 7(2):19-33 (2015) `https://journal.r-project.org/archive/2015/RJ-2015-018`

- R. Genuer, **Variance reduction in purely random forests**. *Journal of Nonparametric Statistics* 24(3):543-562 (2012) DOI

- R. Genuer, J.-M. Poggi, C. Tuleau-Malot, **Variable selection using Random Forests**, *Pattern Recognition Letters* 31(14):2225-2236 (2010) DOI

- R. Genuer, V. Michel, E. Eger, B. Thirion, **Random Forests based feature selection for decoding fMRI data**, *Proceedings of the 19th COMPSTAT* p. 1071-1078 (2010)

Ouvrages ou chapitres d'ouvrages publiés

- R. Genuer, J.-M. Poggi, **Random Forests with R**, coll. Use'R!, Springer, 98 pages (2020)

- R. Genuer, J.-M. Poggi, **Les forêts aléatoires avec R**, coll. Pratique de la statistique, Presses Universitaires de Rennes, 122 pages (2019)

- R. Genuer, J.-M. Poggi, **Arbres CART et Forêts aléatoires, Importance et sélection de variables**, Dans *Apprentissage Statistique et Données Massives*, Maumy-Bertrand M., Saporta G. et Thomas Agnan C. (eds), Technip, p. 295-342 (2018)

Articles soumis et pré-publications

- L. Capitaine, J. Bigot, R. Thiébaut, R. Genuer, **Fréchet random forests for metric space valued regression with non euclidean predictors**, (soumis) arXiv:1906.01741, pré-publication (26 pages)

- S. Arlot, R. Genuer, **Analysis of purely random forests bias** (soumis) arXiv:1407.3939, pré-publication (57 pages)

- R. Genuer, I. Morlais, W. Toussile, **Gametocytes infectiousness to mosquitoes: variable selection using random forests, and zero inflated models** (2010) arXiv:1101.0344, pré-publication (23 pages)
- R. Genuer, J.-M. Poggi, C. Tuleau, **Random Forests: some methodological insigths** (2008) arXiv:0811.3619, pré-publication (32 pages)

Développement logiciel

- Co-créateur et mainteneur du package : R.Genuer, M. Chavent, J. Saracco, **CoVVSURF: Combining clustering of variables with feature selection using random forests**. Package R en développement (depuis 2016) `https://github.com/robingenuer/CoVVSURF`
- Créateur et mainteneur du package : R. Genuer, J.-M. Poggi, C. Tuleau-Malot, **VSURF: Variable Selection Using Random Forests**. R package version 1.1.0 (depuis 2013) `https://cran.r-project.org/package=VSURF`
- Contributeur au package : Adrien Todeschini, **rchallenge: A Simple Datascience Challenge System**. R package version 1.3.2 (2016) `https://cran.r-project.org/package=rchallenge`
- Encadrement de Louis Capitaine dans son développement de package : Louis Capitaine, **LongituRF: Random Forests for Longitudinal Data**. R package version 0.9 (2020) `https://cran.r-project.org/package=LongituRF`
- Encadrement de Louis Capitaine dans son développement de package : Louis Capitaine, **FrechForest: Frechet Random Forests**. Package R en développement (2020) `https://github.com/Lcapitaine/FrechForest`

Activités d'encadrement

- Co-encadrement avec Cécile Proust-Lima de la thèse d'**Anthony Devaux**, *Modélisation et prédiction dynamique individuelle d'évènement de santé à partir de données longitudinales multivariées* (depuis le 01/09/2019)
- Co-direction avec Rodolphe Thiébaut de la thèse de **Louis Capitaine**, *Forêts aléatoires pour données longitudinales de grande dimension* (depuis le 01/09/2017, soutenance prévue le 17/12/2020)

- Co-encadrement avec Cécile Proust-Lima du stage de Master 1 de Santé Publique spécialité Biostatistique, de **Kateline Le Bourdonnec**, *Etude des syndromes gériatriques et des comorbidités dans la cohorte 3C, site Bordeaux, par une analyse factorielle* (2019, 2 mois)
- Encadrement du stage de Master 2 de Modélisation Stochastique et Statistique, de **Louis Capitaine**, *Etude et application des forêts aléatoires pour données longitudinales de grande dimension* (2017, 6 mois)
- Encadrement du stage de Master 2 de Santé Publique spécialité Biostatistique, de **Thomas Esnaud**, *Etude de la méthode de clustering par forêts aléatoires, application à la reconnaissance automatique de populations cellulaires* (2016, 6 mois)
- Encadrement du stage de Master 1 de Santé Publique spécialité Biostatistique, d'**Émilie Chanfreau**, *Etude de l'élagage dans la méthode des forêts aléatoires* (2015, 2 mois)

Responsabilités pédagogiques

- **Responsable du Master 2 Biostatistique** de l'ISPED (depuis 2020)
- **Responsable du Diplôme d'Université** Méthodes Statistiques en Santé de l'ISPED, enseignement à distance (depuis 2017)
- **Référent biostatistique du M1 de Santé Publique** de l'ISPED (2015 - 2020)
- Responsable de trois UEs dans le Master 2 Biostatistique de l'ISPED : *Analyse de données de grande dimension* (depuis 2012), *Analyse de données multidimensionnelles* (2014 - 2016), *Plans d'expérience et ANOVA* (2015)
- Responsable de 4 UEs dans le Master 1 de Santé Publique de l'ISPED : *Outils de simulation en biostatistique* (depuis 2020) *Approche quantitative en santé publique* (2016 - 2020), *Bases mathématiques et programmation* (2012 - 2019), *Tests statistiques* (2011 - 2015)
- Membre du projet **Begin'R** (développement d'un support en ligne pour l'auto-formation au logiciel R). Projet dans le cadre des "Défis numériques", avec le soutient de la MAPI et de l'Idex Bordeaux.
  `http://beginr.u-bordeaux.fr/`

Expertise

- **Rapporteur de la thèse** de Antonio Sutera, *Importance measures derived from random forests*, dirigée par Pierre Geurts et Louis Wehenkel et soutenue le 13/06/2019 à l'Université de Liège, Belgique.
- **Membre du jury de thèse** de Wei Feng, *Investigation des problèmes des données d'apprentissage en classification ensembliste basée sur le concept de marge. Application à la cartographie d'occupation du sol*, dirigée par Samia Boukir et soutenue le 19/07/2017 à l'ENSEGID, Université Bordeaux Montaigne.
- **Rapporteur de la thèse** de Soren Welling, *Characterization of absorption enhancers for orally administered therapeutic peptides in tablet formulations. Applying statistical learning*, dirigée par Per Bruun Brockhoff et Line H. Clemmensen et soutenue le 30/09/2016 à la Technical University of Denmark, Kongens Lyngby, Danemark.

Communications sur invitation

- Séminaire de Statistique et Optimisation de l'institut de mathématiques de Toulouse, *Fréchet random forests for metric space valued regression with non euclidean predictors*, Toulouse (2020)
- Workshop SMEE, *Fréchet random forests*, Anglet (2019)
- Séminaire de Probabilités et Statistiques du laboratoire Paul Painlevé, *Random Forests for Big Data*, Lille (2018)
- ERCIM 2017, *Random Forests for high-dimensional longitudinal data*, dans la session "Recent advances in tree-based methods" organisée par Ruoqing Zhu, Londres (2017)
- Séminaire de Statistique appliquée du CNAM, *Random Forests for Big Data*, Paris (2017)
- jSTAR2015 : journées de STAtistique de Rennes, sur le thème des Méthodes statistiques d'agrégation, *Analyse du biais de forêts purement aléatoires*, Rennes (2015)
- Séminaire de Probabilités et Statistique du Laboratoire de Mathématiques et Applications, *Variable Selection using Random Forests*, Pau (2014)

- ERCIM 2013, *Analysis of purely random forests bias*, dans la session "Random forests and related methods: theory and practice", organisée par Jean-Michel Poggi, Londres (2013)
- Séminaire LERFoB, *Variable Selection using Random Forests*, Nancy (2013)
- Workshop STATLEARN, *Combining clustering of variables and feature selection using random forests*, Bordeaux (2013)
- Journées Modélisation des Biomolécules et de leurs Interactions, *Variable Selection using Random Forests*, Nancy (2012)

  (Cette liste s'arrête à l'année 2012)

Contents

Chapter 1

Introduction

Random forests (RF henceforth) are a non-parametric statistical learning method introduced by Leo Breiman in 2001, adapted to both supervised classification problems and regression problems. They allow to consider qualitative and quantitative explanatory variables together without pre-processing. They can be used to process standard data for which the number of observations is higher than the number of variables, while also performing very well in the high dimensional case, where the number of variables is quite large in comparison to the number of observations. Thus, due for sure to their excellent predictive performance, but also to their flexibility and ease of application, they are extensively used in many fields of application, such as genomics (Boulesteix, Janitza, Kruppa, & König, 2012; Dìaz-Uriarte & Alvarez De Andres, 2006; Goldstein, Hubbard, Cutler, & Barcellos, 2010), ecology (Prasad, Iverson, & Liaw, 2006), pollution prediction (Ghattas, 1999), and for a broader review, see Verikas, Gelzinis, & Bacauskiene (2011). They are now one of the state-of-the-art methods in machine learning and data analysis.

RF are obtained by aggregating a collection of randomized tree predictors. As introduced in Breiman (2001) they use a variant of CART (Classification And Regression Trees) (Breiman, Friedman, Olshen, & Stone, 1984) as individual predictors. More generally, RF are part of the ensemble methods family, which share the same idea of aggregating several individual predictors in order to get the final one: Bagging (Breiman, 1996), Arcing (Breiman, 1998), Randomization (Dietterich, 2000), Random Subspace (Ho, 1998) and Adaboost (Freund & Schapire, 1996), among a large variety of proposals. Ensemble methods ideas have also been used for other methods: Bolasso (Bach, 2008) and Randomized Lasso (Meinshausen & Bühlmann, 2010) are respectively a Bagging-like and and RF-like algorithms developed to stabilize results of the Lasso method (Tibshirani, 1996).

In addition, beyond the performance and the easy to tune feature of the method with very few parameters to adjust, one of the most important aspects in terms of application is the quantification of the input variables relative importance. This

concept, which is not so much examined by statisticians (see for example, Grömping (2015), in regression), finds a convenient definition in the context of RF that is easy to evaluate and which naturally extends to the case of groups of variables (Gregorutti, Michel, & Saint-Pierre, 2015).

Since their introduction, RF have been generalized to various statistical problems. For example, for survival data analysis, Ishwaran, Kogalur, Blackstone, & Lauer (2008) introduced Random Survival Forests, transposing the main ideas of RF to the case where the quantity to be predicted is the time to an event, while Hothorn, Bühlmann, Dudoit, Molinaro, & Van Der Laan (2006) proposed to apply regression RF to a transformation of survival data. Random forests have also been generalized to the multivariate output variable case (see the review by Segal & Xiao (2011), which also provides references from the 1990s), and been adapted to address the problem of ranking (Clémençon, Depecker, & Vayatis, 2013).

1.1 Notations

In this chapter and the next one, we stay as general as possible, without assuming any too restrictive structure on the data. In subsequent chapter, we will assume some more precise structure on the input and output spaces, depending on the types of data and problems at hand.

We assume that a learning sample is available: $\mathcal{L}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ made of n independent and identically distributed (i.i.d.) random couples, coming from the same common distribution as a couple $(X, Y) \in \mathcal{X} \times \mathcal{Y}$. This distribution is, of course, unknown in practice and the purpose is precisely to estimate it, or more specifically to estimate the link that exists between X and Y.

- $\mathcal{X}$ is called the input space and the $X_i, i = 1, \ldots, n$ are the inputs. We assume that $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$ is a product of p spaces. The j-th coordinate X^j of X hence belongs to $\mathcal{X}_j$, and could, e.g., be a real number if $\mathcal{X}_j = \mathbb{R}$ or a function if $\mathcal{X}_j$ is a functional space, or another statistical quantity of interest regarding the type of data available.

- $\mathcal{Y}$ is the output space and the $Y_i, i = 1, \ldots, n$ are the outputs. The nature of prediction problem depends on the nature of the space $\mathcal{Y}$. In this document, we address standard regression and (supervised) classification problems, as well as more general regression problems with metric space valued regression or even time-to-event prediction problems.

The only thing that we need on spaces $\mathcal{X}_1, \ldots, \mathcal{X}_p$ and $\mathcal{Y}$ is that they are all measurable, so that notions of probability, expectation and random variables are well defined. Note that the most classical case in statistical literature is to consider

that for all $j = 1, \ldots, p$ $\mathcal{X}_j = \mathbb{R}$, hence $\mathcal{X} = \mathbb{R}^p$ and we get p continuous input variables. In addition, we will be many times facing high-dimensional data, for which the number of variables p can be very large compared to the number of observations n, usually denoted by $n << p$.

1.2 Statistical Objectives

1.2.1 Prediction

The main objective in statistical learning is prediction: the aim is, using the learning sample $\mathcal{L}_n$, to build a predictor:

$$\widehat{h} : \mathcal{X} \to \mathcal{Y}$$

which associates a prediction $\widehat{y}$ of the output corresponding to any given input $x \in \mathcal{X}$. The "hat" on $\widehat{h}$ means that this predictor is constructed using $\mathcal{L}_n$. We omit the dependence over n for the predictor to simplify the notations, but it does exist.

More precisely, we want to build a powerful predictor with the lowest prediction error (also called generalization error) as possible. The definition of the prediction error of a predictor $\widehat{h}$ depends on the problem at stake:

- In standard regression, where $\mathcal{Y} = \mathbb{R}$, we consider the expectation of the quadratic error: $\mathrm{E}\left[(Y - \widehat{h}(X))^2\right]$.
- In classification with C classes, where $Y \in \mathcal{Y} = \{1, \ldots, C\}$ is a categorical variable, we consider the probability of misclassification: $\mathrm{P}\left(Y \neq \widehat{h}(X)\right)$.
- In metric space valued regression, where $\mathcal{Y} = (\mathcal{Y}, d_{\mathcal{Y}})$ is a metric space with distance $d_{\mathcal{Y}}$, we consider: $\mathrm{E}\left[d^2_{\mathcal{Y}}(Y, \widehat{h}(X))\right]$.
- If $Y \in \mathbb{R}_+$ is a time to event, we consider the Brier score, at a given time point $t \in \mathbb{R}_+$: $\mathrm{E}\left[(\mathbf{1}_{Y \leq t} - \widehat{h}_t(X))^2\right]$ where $\widehat{h}_t(X)$ is the predicted probability that the event occurs at time t.

In all frameworks, the prediction error depends on the unknown distribution of the random couple (X, Y), so it must be estimated. One classical way to proceed is, using a test sample $\mathcal{T}_m = \{(X'_1, Y'_1), \ldots, (X'_m, Y'_m)\}$, also made of i.i.d. random couples drawn from the same distribution as (X, Y), to compute a test error. For example, in standard regression, it leads to calculate the mean square error: $\frac{1}{m}\sum_{i=1}^{m}\left(Y'_i - \widehat{h}(X'_i)\right)^2$.

In the case where a test sample is not available, the prediction error can still be estimated either by cross-validation, or by a specific estimate included in RF, called Out-Of-Bag error (see Section 2.4).

1.2.2 Variable selection and importance of variables

A second classical objective, especially useful when analyzing high-dimensional data, is variable selection. This involves determining a subset of the input variables that are actually useful and active in explaining the input-output relationship. The quality of a subset of selected variables is often assessed by the performance obtained with a predictor using only these variables instead of all initial set.

In addition, we can focus on constructing a hierarchy of input variables based on a quantification of the importance of the effects on the output variable. Such an index of importance therefore provides a ranking of variables, from the most important to the least important. As we will see, RF offer a very interesting framework to the definition of a variable importance score, which can be defined in general and easily adapted to the different problems addressed here (standard regression, classification, time-to-event analysis, etc.).

1.3 Applications

Most of applications presented in this document comes from problems arising from health domain. For some of them, they are actually the motivation for the development of new RF methods.

1.3.1 Image data in brain functioning study

The knowledge of how the brain functions, e.g., which region of the brain are activated when an individual performs a certain task (visual recognition of objects, language, mental calculation, etc.) is an active problem in nowadays neurology domain. Functional Magnetic Resonance Imagery (fMRI) allows to very precisely measure the brain activity, leading to dataset with several hundreds of thousands of voxels (a pixel in 3-D) as continuous variables. Since the number of experiments is usually of order a few tenth or at most hundreds, we face very high-dimensional problems when analyzing this type of data.

In this context, we apply a variable selection procedure based on RF on fMRI data(Robin Genuer et al., 2010) coming from $n = 64$ experiments and a total of $p = 1000$ continuous input variables (obtained by a clustering method apply on the voxels measuring activity in the all brain). The task done by individuals was an object recognition task with 4 different shapes of the same object. Hence, in

this application, we have a high-dimensional classification problem with 4 classes. The main objective is then to select the brain regions which permit to discriminate between the stimuli (the different object shapes), given the brain activities measurements. The prediction objective is kind of secondary here, and is more considered as a tool to serve the variable selection goal.

Another application in this field was done in Zago et al. (2017). In this work, we apply a variable selection technique based on support vector machines (Guyon, Weston, Barnhill, & Vapnik, 2002; Schölkopf, Smola, Williamson, & Bartlett, 2000) to assess hemispheric pattern of language dominance.

1.3.2 Proteomic data in clinical trials

The use of "omics" data to understand the behavior of individuals in terms of response of a treatment or development of pathologies, is a challenging problem, more and more tackled in the health domain since more and more characteristics are now collected on individuals.

In Chavent, Genuer, & Saracco (2019) we apply a combination of clustering of variables and RF-based variable selection procedure to clinical trial data. It involved $n = 44$ patients with a rectum cancer who undertook a treatment of chemotherapy and radiotherapy, before a surgery intervention. The main goal of this study was to predict if a patient will respond favorably to the treatment, using $p = 4786$ continuous input variables measuring protein abundances at baseline. We tackled here an high-dimensional binary classification problem and the method manage to select interesting groups of informative variables, whereas the group structure was a priori unknown.

1.3.3 Genomic data in vaccine trials

The measure of genomic data becomes the norm in vaccine trial. In Capitaine et al. (2020b) and Capitaine et al. (2020a) we focused on two vaccine trials for HIV positive patients. The first, called DALIA-1 (Lévy et al., 2014), is a therapeutic vaccine trial including $n = 19$ HIV-infected patients who received an HIV vaccine candidate before stopping their antiretroviral treatment. Expression of $p = 5399$ gene transcripts and the HIV viral load were measured at 15 time points during the trial (6 time points before antiretroviral treatment interruption, and 9 time points after). The objective is to predict the HIV viral load dynamics after antiretroviral treatment interruption for a patient given the evolution of his/her gene expression during the vaccination phase (Thiébaut et al., 2019). We deal here with high-dimensional longitudinal data, and we have to take into account that both continuous input variables (gene expression) and the continuous output variable (HIV viral load) are repeatedly measured. Furthermore, more than the predictions

themselves, knowing which genes are the most involved in the prediction problem is of great interest to better understand the vaccination mechanism in this context.

The second trial that we analyzed in Capitaine et al. (2020a), called LIGHT, is a therapeutic vaccine trial including $n = 97$ HIV-infected patients, and $p = 1150$ continuous input variables of genes expression were measured between 1 and 4 times, leading to a total number of 234 observations. Hence, compared to DALIA trial, we have a lot more individuals, but less repeated measurements. The objective of this analysis was to assess the capacity of predicting the abundance of CD4 T cells using gene expression data as measured by RNA sequencing in whole blood. Hence we were again dealing with high-dimensional longitudinal data but with sparse and unbalanced trajectories.

1.3.4 Dynamic predictions of health events

In an ongoing work with Anthony Devaux and Cécile Proust-Lima, we develop a dynamic prediction method based on machine learning algorithms including RF. We adopt a landmark approach, which consists in using past data until a landmark time t_{LM} to predict the probability that an event occurs at some horizon time t_{Hor}. We apply the proposed methodology in two different contexts in health that share the following characteristics: for some individuals we measure about 10 input variables (some are continuous, some are categorical) repeatedly over time and the objective is to predict the time an health event occurs. We also have additional characteristics of individuals, not repeated over time.

For the first application, the event to predict is death for primary biliary cirrhosis patients. For $t_{\text{LM}} = 4$ years after the follow-up start, we get $n = 225$ patients and there are 11 longitudinal input variables and 3 time-independent input variables. For the second application, the objective is to predict all cause death of elderly people. For $t_{\text{LM}} = 80$ years, we have $n = 1561$ people, with 9 longitudinal input variables and 18 time-independent input variables.

The proposed strategy consists in summarizing longitudinal variables trajectories until t_{LM} and using those summaries into time-to-event prediction methods, such as random survival forests (Ishwaran et al., 2008). Therefore, we face here a problem of time-to-event prediction when input variables are either repeated over time, or time-independent. Again, one interesting question in both contexts is to select the most informative variables.

Chapter 2

Random Forests: General Principle

The purpose of this chapter is to give the main ideas that drive RF methods. Here, we state some definitions as general as possible and make some remarks that apply in all particular cases developed in the next chapters.

The general principle of RF is to aggregate a collection of randomized trees. The main idea is, instead of seeking to optimize a predictor "at once" as it is usually done by most statistical learning methods (e.g., a single tree), to put together a set of predictors. As we will see more in details in the following, individual predictors are not built in optimal manner. On the contrary, they are usually obtained by randomly perturbing optimal procedures. But by doing so, the forest benefits from an extensive exploration of the space of all possible predictors, which, in practice, results in better predictive performance.

We give the following general definition of random forests, inspired by that of Breiman (2001):

Definition 2.1 (General Random Forests)**.** Let $\left\{\widehat{h}(.,\Theta_1),\ldots,\widehat{h}(.,\Theta_q)\right\}$ be a collection of tree predictors, with $\Theta_1,\ldots,\Theta_q$ q i.i.d. random variables independent of $\mathcal{L}_n$. The RF predictor $\widehat{h}_{\mathrm{RF}}(.)$ is obtained by aggregating this collection of randomized trees.

Remark. We stress that in this definition the variables Θ_ℓ are independent of $\mathcal{L}_n$. Even though it was not specify in Breiman (2001), is more consistent with the intuition that the additional randomness provided by the Θ_ℓ is disconnected from the learning sample. In addition, it encompasses the most commonly used RF variants and e.g., all RF methods introduced in this document.

We also note that the RF predictor $\widehat{h}_{\mathrm{RF}}(.)$ actually depends on all $\Theta_1,\ldots,\Theta_q$, but we do not make this dependence explicit to lighten notation.

This definition is illustrated by the diagram in Figure 2.1. This scheme will be adapted several times later on, depending on the different RF methods presented. Hence, to precisely define a RF method adapted to a particular context, we need

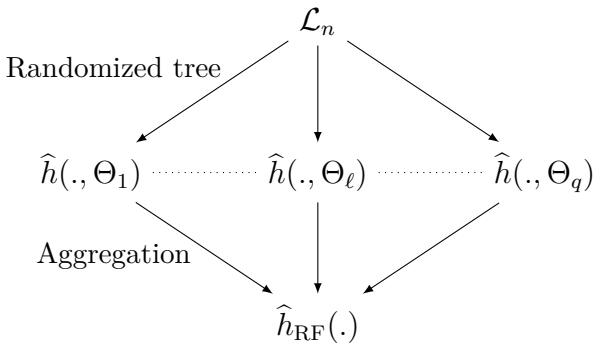

Figure 2.1: General scheme of random forests.

the three mandatory following components:

1. A way of building a tree adapted to the data at stake.
2. A way of randomizing individual trees.
3. A way of aggregating several trees.

In the following sections, we discuss more in details those three aspects of RF, but before that it is useful to keep in mind the following remark.

In order to obtain good predictive performance, a RF method must, in general, build a collection of trees that is:

- As diverse as possible, because aggregating a set of predictors that are all very similar would give nothing more than again a similar predictor.
- Made up of individual predictors with acceptable predictive capacity, because if for a new observation x all trees provide a bad prediction, the aggregation of these predictions has no chance of being correct.

2.1 General Statements about Trees

Initially, ensemble methods, like RF, were introduced to improve prediction performance of trees, and the main idea from one of the first of them, Bagging (Breiman, 1996), was to stabilize CART (Breiman et al., 1984). Indeed, CART's principal drawback is its instability, in the sense that one can observe a huge change in the

resulting predictor when applying CART on a slightly modified dataset. Bagging, by aggregating several CART trees built on different bootstrap samples from $\mathcal{L}_n$, do stabilize the result. Moreover, having the "diversity among trees" notion in mind, the instability of CART is actually an advantage for Bagging. In other words, if the method used to get individual predictors were too stable, then the aggregation of those individual predictors would not be much different from them.

However, as a matter of fact, ideas driving tree methods are quite general, and can thus be transposed to many frameworks. First, trees are piece-wise constant predictors, and the general principle to obtain them is to recursively partition the input space $\mathcal{X}$. The way this partitioning is lead is the key aspect of the methodology. Thus, trees belong to the family of partitioning predictors and more particularly to the family of data-dependent partitioning predictors (see, e.g., Chap. 4 and 13 of Györfi, Kohler, Krzyżak, & Walk (2002)). Partitioning predictors are quite well-known. For example, some general results about consistency and rate of convergence were obtain by Stone (1977) and Stone (1982), under assumptions on the partition. For data-dependent partitioning predictors, Nobel (1996) and Lugosi & Nobel (1996) also stated general consistency results.

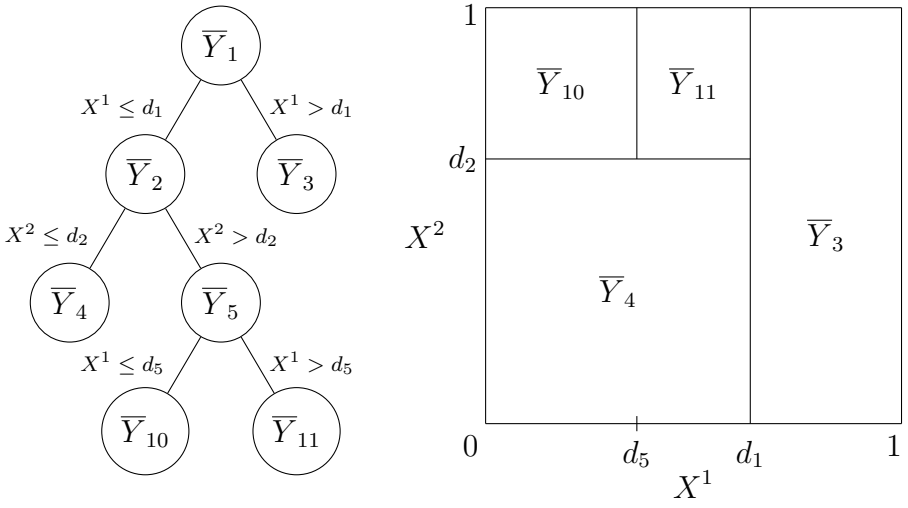

Figure 2.2: Left: a regression tree with $\mathcal{X} = [0,1] \times [0,1]$ and $\mathcal{Y} = \mathbb{R}$. Right: the associated partition in the input space ($\overline{Y}_1$, $\overline{Y}_2$ and $\overline{Y}_5$ do not appear because they are not associated with leaves).

In this document, we focus on trees and we are calling them like this thanks to the recursive characteristics of their partitioning, to which we naturally associate a tree (in terms of graph). In addition, we restrict ourselves to the case of binary

trees, which means that sets of the current partition are always split into two sets during the partitioning process. Figure 2.2 give a representation of a tree and its associated partition of the input space, in the standard regression case. It can be seen that each node of the tree has an attached output value: in this case, this is the mean of output values associated to observations belonging to the node. We also call a leaf, a node that do not have children nodes. The resulting piece-wise constant predictor is the partitioning predictor which, for a new observation x, predicts the output value corresponding to the set of the partition x belongs to. An equivalent formulation of the prediction process for the tree is: drop down observation x at the top of the tree (i.e., the root node) and determine the leaf it falls into, after going through the different splits, given x input variables values, and then predict the output value attached to that leaf.

All references and works cited here were developed in a classical framework with inputs in $\mathbb{R}^p$ and outputs in $\mathbb{R}$. However, the core ideas are far more general:

- The notion of partition exists for very general input spaces.
- The recursive binary partitioning process is simple enough to be generalized for many kinds of data.

We do not give full details of the pruning algorithm of CART (Breiman et al., 1984; Gey & Nedelec, 2005) here, but we stress that once a tree is obtained, it is most of the time possible to apply the pruning step, in order to find the final tree with the best prediction performance, i.e., the one that well balances empirical error and total number of leaves.

Finally, since we focus on binary trees in this document, finding a way of building individual trees of RF reduces to find a way to partition a set of observations into two subsets. This separation into two subsets is called a split and this division procedure is called the splitting process.

2.2 Additional Randomness

Once the tree building strategy is set, the way of randomizing individual trees has to be chosen. The main kinds of additional randomness in the literature are actually quite transverse, in the sense that they can be used for several RF methods:

- To randomly resample the learning set before applying a tree method. Bootstrap samples (obtained by n uniform draws among $\mathcal{L}_n$ observations, with replacement) are usually used, but "m out of n" samples (obtained by m

uniform draws among $\mathcal{L}_n$ observations, without replacement) were also considered, in practice or to allow theoretical analyses (Banerjee & McKeague, 2007; Bühlmann & Yu, 2002; Scornet, Biau, & Vert, 2015).

- To randomly choose a subset of input variables (obtained by r uniform draws among the p input variables, without replacement), either at each node of a tree (Breiman, 2001; Geurts, Ernst, & Wehenkel, 2006) or for the entire tree (Ho, 1998), and restrict the search for the best split only among the selected variables.

- To randomly choose a split, either among a set of optimized splits (Dietterich, 2000) or uniformly among all possible splits only along r input variables previously chosen (Geurts et al., 2006).

A lot of combinations of those types of randomness have been tried in the literature. The addition of randomness before or during the tree building has to be well dosed: too little randomness could lead to a collection of trees not diverse enough; too much randomness could generate individual trees with too weak prediction performance.

2.3 Aggregation

The last ingredient of RF, and more generally of any ensemble method, is the way individual predictors are aggregated. The aggregation has to be adapted to the type of data in the output space $\mathcal{Y}$. Indeed, the aggregation is done at the predictions level: for a new observation $x \in \mathcal{X}$, each tree provides a prediction $\widehat{y} \in \mathcal{Y}$ of its associated output, and RF aggregate those prediction values.

Even if the aggregation is a very important step of RF, a majority of RF methods in the literature put together individual predictions in a quite simple and usually the most natural way. For example, in a standard regression framework with $\mathcal{Y} = \mathbb{R}$, the aggregation reduces to computing the mean of individual predictions. More advanced ways of aggregating trees predictions, such as aggregation procedures (see e.g., Lecué (2007) among others) that seek the best way of putting together predictions to get the best aggregated predictor, have not been, up to our knowledge, fully addressed in the literature. This is actually not so surprising, and could come from the fact that in RF, trees are i) built independently from each other and ii) sub-optimal by nature, because they are usually obtained by a random perturbation of an optimal algorithm. Therefore, one tree of a RF is not so interesting by itself, but it is all the collection once aggregated that matters. Moreover, it is not clear what would be the advantage of treating the trees differently from each other. By doing so, and by optimizing the aggregation procedure, we could also loose the benefit of the exploration of the space of all possible

predictors, given by the previously mentioned additional randomness. From this perspective, RF are fundamentally different from other ensemble methods, like, e.g., Boosting (Freund & Schapire, 1996).

Extract the best tree of a RF, instead of aggregating all trees, could be at first quite tempting. Indeed, getting a tree allows to retrieve a great interpretability of the resulting predictor. However, extracting a tree with comparable predictive performance is almost hopeless because a tree and a RF are in nature too different.

To conclude, we stress that even if the aggregation is not optimized for most RF methods, it remains a crucial step of the methodology. It is indeed, when the collection of trees is aggregated that the gain in prediction performance is obtained.

2.4 Out-Of-Bag Error and Variable Importance

In this section, we define two very useful quantities naturally computed in RF methods. The mandatory condition to actually get these quantities is that a resampling step has been performed before the building of a tree. So, this section is a little bit less general than the previous ones, but as we will see in subsequent chapters, this resampling step is very common in RF, and the following mechanisms are general enough to be used in many different frameworks.

So, assume that the ℓ-th tree of the RF is built on $\mathcal{L}_n^{\Theta_\ell}$, a bootstrap sample of $\mathcal{L}_n$ (other resampling techniques would also work as soon as some observations of $\mathcal{L}_n$ are left out the resulting sample). On average (for n sufficiently large), the bootstrap procedure leaves $0.368n$ observations outside the resulting sample. Thus, to each bootstrap sample $\mathcal{L}_n^{\Theta_\ell}$, we can associate $\mathcal{L}_n^{\mathrm{OOB}_\ell} = \mathcal{L}_n \setminus \mathcal{L}_n^{\Theta_\ell}$ (ignoring repetitions in $\mathcal{L}_n^{\Theta_\ell}$), the set of $\mathcal{L}_n$ observations not belonging to $\mathcal{L}_n^{\Theta_\ell}$. $\mathcal{L}_n^{\mathrm{OOB}_\ell}$ is called the Out-Of-Bag (OOB) sample associated to the ℓ-th tree.

The main idea of OOB error and variable importance (VI) is to use those OOB samples as "local test sets". Indeed, since a tree is built on a bootstrap sample, then the corresponding OOB observations have not already been considered by this tree and thus can be used to fairly assess predictive performance. The OOB error is defined as follows:

Definition 2.2 (Out-Of-Bag error)**.** Consider the i-th observation X_i of the learning set $\mathcal{L}_n$. To predict the output associated to this input, only individual predictors built on bootstrap samples not containing the couple (X_i, Y_i) are aggregated. This provides a prediction $\widehat{Y}_i$ of Y_i. Doing this for every observations of $\mathcal{L}_n$ allows to compute an estimation of the prediction error.

The computation of the prediction error estimation depends on the type of the problem considered (see Section 1.2.1). This OOB error estimate permits to

get unbiased prediction error estimation, without the need to perform, e.g., an additional cross-validation procedure.

One of the main drawback of switching from trees to RF methods is the loss of the ease of interpretation. Indeed, while a result of tree is quite easy to understand and to comment, an RF obtained by aggregating hundreds of trees can not be visualized or interpreted easily. In order to fill the gap between trees and forests interpretability, variable importance scores were introduced. There exist two main VI scores: a permutation-based VI score (also called MDA for Mean Decrease Accuracy), and a heterogeneity-based VI score (also called MDI for Mean Decrease Impurity).

In this document we focus on the permutation-based VI score, defined as follows, for the j-th variable X^j:

Definition 2.3 (Permutation-based variable importance)**.** Let us fix $j \in \{1, \ldots, p\}$ and calculate $\mathrm{VI}(X^j)$ the permutation-based importance score of variable X^j:

- Consider the ℓ-th bootstrap sample $\mathcal{L}_n^{\Theta_\ell}$ and the associated $\mathcal{L}_n^{\mathrm{OOB}_\ell}$ sample.
- Calculate errOOB_ℓ, the error made on $\mathcal{L}_n^{\mathrm{OOB}_\ell}$ by the tree built on $\mathcal{L}_n^{\Theta_\ell}$.
- Then randomly permute the values of variable X^j in $\mathcal{L}_n^{\mathrm{OOB}_\ell}$. This gives a perturbed sample, noted $\mathcal{L}_n^{\widetilde{\mathrm{OOB}_\ell}^j}$.
- Finally, calculate $\widetilde{\mathrm{errOOB}_\ell}^j$, the error made on $\mathcal{L}_n^{\widetilde{\mathrm{OOB}_\ell}^j}$ by the tree built on $\mathcal{L}_n^{\Theta_\ell}$.
- Repeat these operations for all bootstrap samples. The variable importance of variable X^j, is then defined by the difference between the average error of a tree on the perturbed OOB sample and that on the OOB sample:

$$\mathrm{VI}(X^j) = \frac{1}{q} \sum_{\ell=1}^{q} \left(\widetilde{\mathrm{errOOB}_\ell}^j - \mathrm{errOOB}_\ell \right) \quad .$$

With this VI definition, the more the increase of the averaged error of a tree on its associated OOB sample after permutation is, the more important the variable is.

Chapter 3

Standard Random Forests

In this chapter, I present a series of works that use standard RF, that is the algorithm RF-RI (Random Forests Random Inputs) from Breiman (2001). After precisely defining this method by setting tree building, additional randomness and aggregation, I present a variable selection procedure based on standard RF. Finally I address the particular case of using RF to analyze Big Data.

3.1 Definitions

Standard RF were introduced to deal with standard regression and classification problems. Hence in this chapter, the input space $\mathcal{X} = \mathbb{R}^p$ (input variables also can be categorical), and the output space $\mathcal{Y}$ is either $\mathbb{R}$ for regression, or $\{1, \ldots, C\}$ for classification.

In this context, trees are usually CART-like trees (Breiman et al., 1984), in which the way of splitting a node t into two child nodes is performed by maximizing the following criterion:

$$\Delta(t) = \Phi(t) - \left[\frac{n_{t_L}}{n_t}\Phi(t_L) + \frac{n_{t_R}}{n_t}\Phi(t_R)\right]$$

where $\Phi(t)$ is an heterogeneity measure of outputs in node t, t_L and t_R the left and right child nodes respectively, and n_t the number of observations belonging to a node t. The heterogeneity measure $\Phi(t)$ must be adapted to the outputs nature:

- In regression, $\Phi(t) = \dfrac{1}{n_t} \displaystyle\sum_{i:X_i \in t} (Y_i - \overline{Y}_t)^2$, is the variance of outputs in node t.
- In classification, $\Phi(t) = \displaystyle\sum_{c=1}^{C} \hat{p}_t^c(1 - \hat{p}_t^c)$, with $\hat{p}_t^c$ the proportion of observations of class c in node t, is the Gini index of outputs in node t.

In CART, the criterion $\Delta(t)$ is maximized (we seek the largest heterogeneity decrease at each split) w.r.t. all admissible splits, i.e., all splits involving any of the p input variables are considered. For a continuous input variable X^j, admissible splits are of the form $\{X^j \leq s\} \cup \{X^j > s\}$ with $s \in \mathbb{R}$, while for a categorical variable $X^{j'}$ they are of the form $\{X^{j'} \in A\} \cup \{X^{j'} \notin A\}$ with A a subset of $X^{j'}$ categories. In both cases, we take the convention that observations that verify the left event ($\{X^j \leq s\}$ or $\{X^{j'} \in A\}$) go to the left child node, and other observations go to the right child node.

We now give the definition of the RF used in this chapter.

Definition 3.1 (Random Forests Random Inputs)**.** The RF-RI predictor is an RF predictor with the following characteristics:

- Before building a tree, a bootstrap sample is drawn and the associated randomness is referred as Θ^1.
- To split a node t, a set $\mathcal{D}_t$ of d input variables is randomly selected (uniformly and without replacement among the p input variables), and the criterion $\Delta(t)$ is maximized w.r.t. admissible splits only involving variables in $\mathcal{D}_t$. We denote by Θ^2 all random draws of input variables sets at each node of a tree. Furthermore, the trees are fully developed and no pruning step is performed.
- Individual trees are aggregated as follows:
  - $\widehat{h}_{\mathrm{RF-RI}}(x) = \dfrac{1}{q}\displaystyle\sum_{\ell=1}^{q} \widehat{h}(x, \Theta_\ell^1, \Theta_\ell^2)$ (standard mean in regression).
  - $\widehat{h}_{\mathrm{RF-RI}}(x) = \underset{1\leq c\leq C}{\arg\max} \displaystyle\sum_{\ell=1}^{q} \mathbf{1}_{\widehat{h}(x,\Theta_\ell^1,\Theta_\ell^2)=c}$ (majority vote in classification).

We note that the ℓ-th tree is denoted by $\widehat{h}(., \Theta_\ell^1, \Theta_\ell^2)$ and can be seen as a doubly randomized tree: first thanks to the bootstrap sample draw, and secondly thanks to the random draws of variable sets at each node. In addition, we stress that the size d of the variable sets is the same for every nodes of every trees in the forest. A diagram of the RF-RI algorithm can be found in Figure 3.1.

Remark. The language abuse consisting in naming the RF-RI method by RF is widely used in the literature on random forests. Since, we give a more general presentation of RF in this document, we try to avoid this language abuse.

This language abuse is also frequent in implementations of RF methodology: for example, the very popular R-package `randomForest` (Liaw & Wiener, 2002) actually implements the RF-RI algorithm.

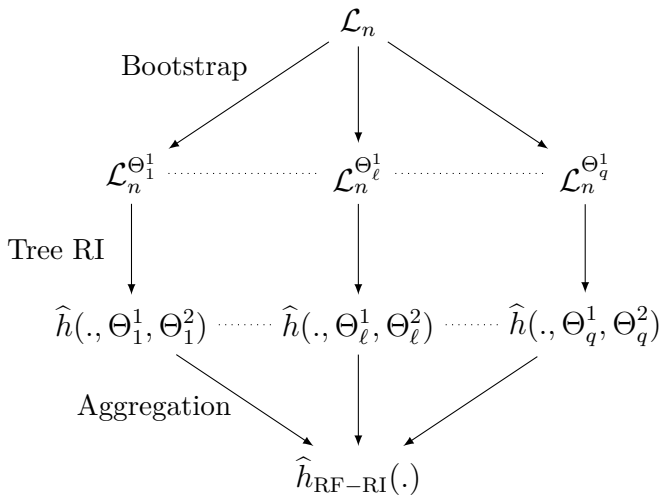

Figure 3.1: RF-RI scheme.

3.2 Variable Selection using Random Forests

As part of my first research activities, I develop a variable selection (Guyon & Elisseeff, 2003) procedure based on RF-RI, called VSURF (for Variable Selection Using Random Forests). This led to a series of papers from the methodological presentation of the procedure (R. Genuer et al., 2010), its application to analyze fMRI (functional Magnetic Resonance Imagery) data (Genuer et al., 2010), to the development of the R package `VSURF` (Genuer, Poggi, & Tuleau-Malot, 2015, 2019). Since its introduction, VSURF has been quite used in different application domains or compared with other variable selection procedures (Cadenas, Garrido, & Martínez, 2013; Sanchez-Pinto, Venable, Fahrenbach, & Churpek, 2018; Speiser, Miller, Tooze, & Ip, 2019).

VSURF is an automatic variable selection procedure, heavily based on RF-RI, that combines thresholding and stepwise strategies, specifically designed to analyze high-dimensional data:

- Its most appealing characteristic is that it is fully automatic, in the sense that during the procedure all quantities of interest are computed on the available data only, and hence no a priori (e.g., on the number of variables to select) is needed. Together with the fact that it is based on RF-RI which are a non-parametric predictor, it makes VSURF a very versatile tool, applicable to many different datasets (at least for regression and classification).
- VSURF heavily uses RF-RI, both to compute (permutation-based) variable importance, but also to build predictors involving subset of variables that

are compared using OOB error. Moreover, the order given by variable importance scores is used during all the procedure. Thus, VSURF is highly dependent on the capacity of RF-RI to well measure variable importance and well estimate prediction error of several predictors by OOB error.

- The main limitation of VSURF is surely its overall computation time. Indeed, in its first step, it computed several RF-RI predictors with variable importance calculation, so if one RF-RI is already computationally demanding, this first step runtime will sometimes be prohibitive. However, the implementation allows to easily use parallel computing, and even if the procedure has been thought to be automatic, there exist several tuning parameters that can be adapted to decrease the computational burden.

We now give more details about VSURF procedure. The method involves two main steps: the first, fairly coarse, proceeds by thresholding the importance of the variables to eliminate a large number of useless variables, while the second, finer and ascending, consists of a sequential introduction of variables into RF-RI predictors.

In this procedure, we distinguish two variable selection objectives: interpretation and prediction (although this terminology may lead to confusion):

- For interpretation, we try to select all the variables X^j strongly related to the response variable Y (even if the variables X^j are correlated with each other).

- While for a prediction purpose, we try to select a parsimonious subset of variables sufficient to properly predict the output variable.

Typically, a subset built to satisfy the first objective may contain many variables, which will potentially be highly correlated with each other. On the contrary, a subset of variables satisfying the second one will contain few variables, weakly correlated.

The following situation illustrates the distinction between the two types of variable selection objectives. Consider a high-dimensional classification problem ($n << p$) for which each explanatory variable is associated with a pixel in an image or a voxel in a 3D image as in brain activity (fMRI) classification problems (Genuer et al., 2010). In such situations, it is natural to assume that many variables are useless or uninformative and that there are unknown groups of highly correlated predictors corresponding to regions of the brain involved in the response to a given stimulation. Although both variable selection objectives may be of interest in this case, it is clear that finding all the important variables highly related to the response variable is useful for interpretation, since the selected variables would correspond to regions of the brain. Of course, the search for a small number of

variables, sufficient for a good prediction, makes it possible to obtain the most discriminating variables in the regions previously highlighted but is of less priority in this context.

With those two objectives defined, VSURF works as follows:

- Step 1. Ranking and preliminary elimination:
  - Rank the variables by decreasing importance (in fact by average VI over typically 50 forests).
  - Eliminate the variables of low importance (let us denote by m the number of retained variables).
    More precisely, starting from this order, we consider the corresponding sequence of standard deviations of the VIs that we use to estimate a threshold value on the VIs. Since the variability of the VIs is greater for the variables truly in the model than for the uninformative variables, the threshold value is given by estimating the standard deviation of the VI for the latter variables. This threshold is set at the minimum predicted value given by the CART model fitting the data (X, Y) where the Y are the standard deviations of the VI and the X are their ranks. Then only variables whose average importance VI is greater than this threshold are kept.
- Step 2. Variable selection:
  - For *interpretation*: we build the collection of nested models given by forests built on the data restricted to the first k variables (that is the k most important), for $k = 1$ to m and we select the variables of the model leading to the lowest OOB error. Let us denote by m' the number of selected variables.
    More precisely, we calculate the averages (typically over 25 forests) of the OOB errors of the nested models starting with the one with only the most important variable and ending with the one involving all the important variables previously selected. Ideally, the variables of the model leading to the lowest OOB error are selected. In fact, to deal with instability, we use a classical trick: we select the smallest model with an error less than the lowest OOB error plus an estimate of the standard deviation of this error (based on the same 25 RF).
  - For *prediction*: from the variables selected for interpretation, a sequence of models is constructed by sequentially introducing the variables in increasing order of importance and iteratively testing them. The variables of the last model are finally selected.

More precisely, the sequential introduction of variables is based on the following test: a variable is added only if the OOB error decreases more than a threshold. The idea is that the OOB error must decrease more than the average variation generated by the inclusion of non-informative variables. The threshold is set to the average of the absolute values of the first order differences of the OOB errors between the models including m' variables and the one with m variables:

$$\frac{1}{m-m'}\sum_{k=m'}^{m-1} |\,\mathrm{errOOB}(k+1) - \mathrm{errOOB}(k)\,| \tag{3.1}$$

where $\mathrm{errOOB}(k)$ is the OOB error of the forest built with the k most important variables.

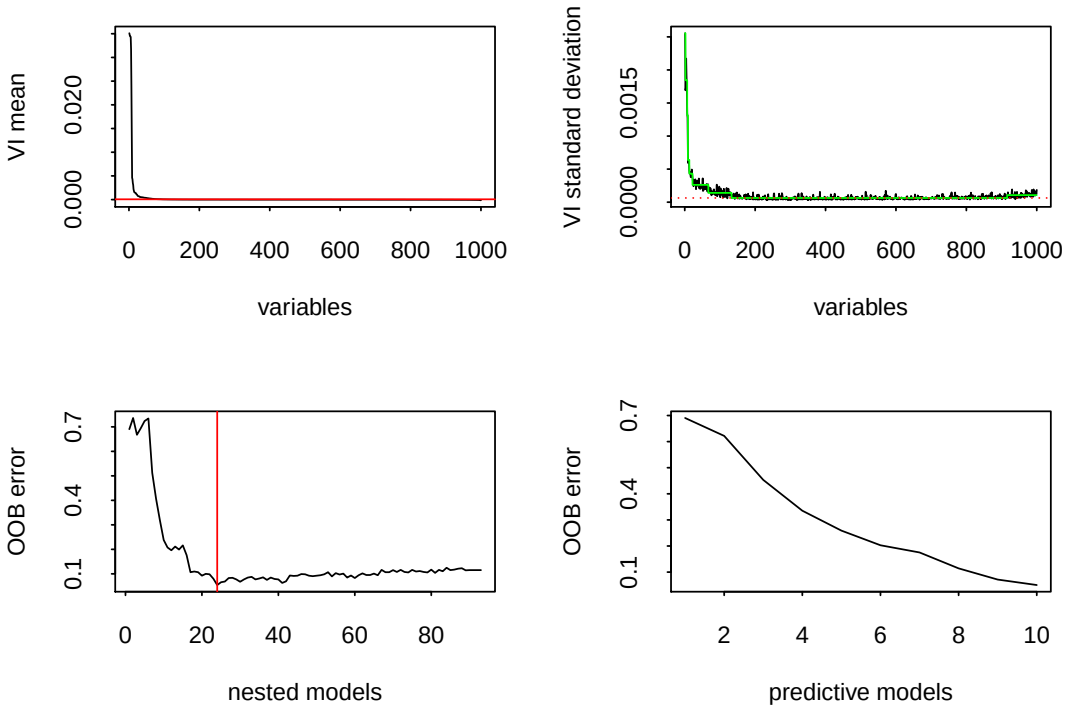

Figure 3.2: Graphs illustrating the results of VSURF on Vac18 data.

A typical output of VSURF is given in Figure 3.2. The procedure has been applied on the `vac18` dataset, coming from an HIV prophylactic vaccine trial (Thiébaut et al., 2012). Expressions of a subset of $p = 1000$ genes were measured for $n = 42$ observations (corresponding to 12 negative HIV participants), corresponding to 4

different stimuli (different vaccines). The prediction objective here is to determine, in view of gene expression, the stimulation that has been used. Thus, this a 4-class high-dimensional classification problem.

In this example, starting with the 1000 available variables, the first thresholding step keeps 93 variables, the interpretation step leads to the selection of 24 variables, while the prediction step selects 10 variables.

3.3 Random Forests for Big Data

Due to the very rapid development of technology, huge amounts of data are nowadays collected daily in many domains. In this section we focus on the context, usually called Big Data, where the number of observations included in a dataset is extremely large: more than hundreds of millions, to give an idea of the order of magnitude. In this context a first objective is to study the applicability of statistical methods to such datasets and adapt them if needed. For example, some datasets are too large to fit in a single computer memory, thus to be analyzed the dataset has to be distributed among several computers. Hence, one question is how can, e.g., RF-RI methods, be applied to this kind of data.

In Genuer, Poggi, Tuleau-Malot, & Villa-Vialaneix (2017), we focus on classification problems and study five RF-RI variants that adapts to Big Data: one relies on subsampling while three others are related to parallel implementations of RF-RI and involve either various adaptations of bootstrap to Big Data or "divide-and-conquer" approaches. The fifth variant relates to online learning of RF.

First of all, since, as stated in Definition 2.1, individual trees are always independent, RF methods can easily be parallelized. So, if the building of one tree can be achieved in reasonable amount of time and if enough processors are available, computation time of a RF run can naturally be reduced. However if the data are so large that they do not fit in the computer memory, it is not enough. In our study, we consider two different parallel implementations of RF-RI, which aim at reducing the size of the data handled by one single process (in order to build individual trees):

- First, we consider the m-out-of-n bootstrap (Bickel, Götze, & Zwet, 1997), which randomly selects only m (with $m << n$) observations without replacement from the learning set $\mathcal{L}_n$, into RF-RI.

- Secondly, we also implement a variant of RF-RI involving the "Bag of Little Bootstraps" (BLB) (Kleiner, Talwalkar, Sarkar, & Jordan, 2014). This method builds K bootstrap samples of size n but each containing only m (again with $m << n$) different observations. On each obtained sample,

forests of q trees are built, and finally aggregated into the final predictor (made of $K \times q$ trees). Figure 3.3 gives a diagram of this variant.

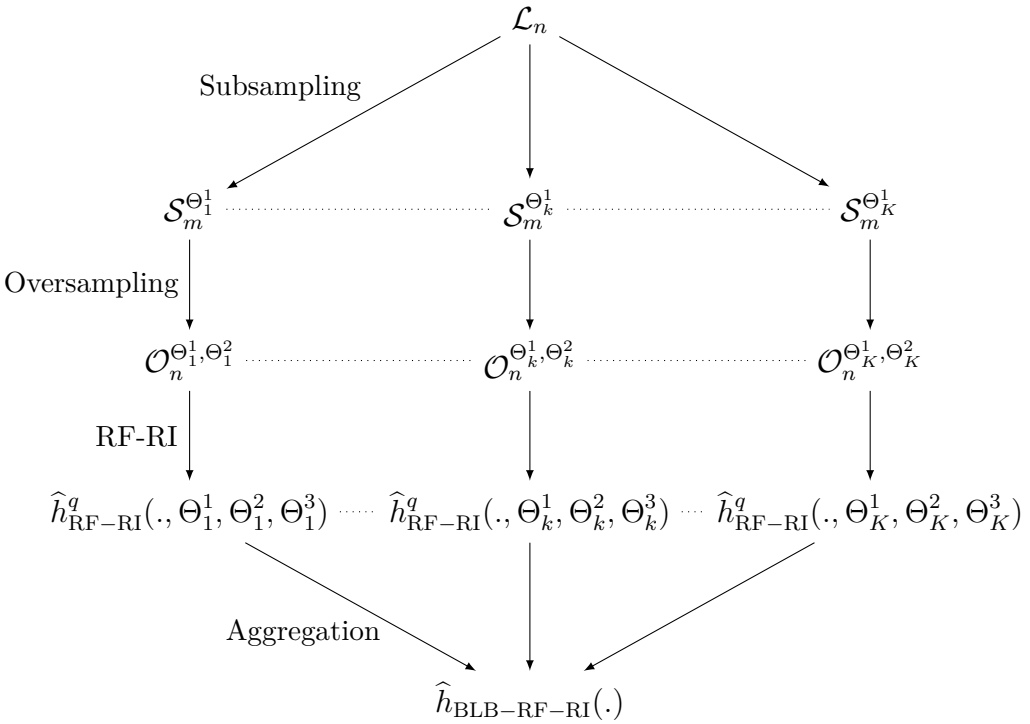

Figure 3.3: Scheme of a RF-RI variant using Bag of Little Bootstraps. $\mathcal{S}_m^{\Theta_k^1}$ denotes a random subsample of $\mathcal{L}_n$ made of m observations, $\mathcal{O}_n^{\Theta_k^1,\Theta_k^2}$ is obtained by randomly oversample $\mathcal{S}_m^{\Theta_k^1}$ until having n observations and $\widehat{h}^q_{\mathrm{RF-RI}}(.,\Theta_k^1,\Theta_k^2,\Theta_k^3)$ is a RF-RI predictor with q trees built on $\mathcal{O}_n^{\Theta_k^1,\Theta_k^2}$, with Θ_k^3 denoting the random selection of subset of variables at each node of the trees.

The remaining RF variants that we study are *i)* a simple RF-RI built on a random subsample of $\mathcal{L}_n$, *ii)* a "divide-and-conquer" approach (Chu et al., 2010) of RF-RI (where forests of q trees are built on the K sets of a partition of $\mathcal{L}_n$) and *iii)* an online RF variant (Denil, Matheson, & Freitas, 2013; Saffari, Leistner, Santner, Godec, & Bischof, 2009). This last variant is quite different from the other ones: the general principle is to update a RF predictor every time a new observation is considered, the learning set being read sequentially. Furthermore, whereas all previous variants are all RF-RI adaptations, Denil et al. (2013) used Extremely Randomized Trees (Geurts et al., 2006) to be able to perform quick update of their predictor.

In our work, we compare those Big Data RF variants first in a simulation study with 15 millions of observations, and then on a real world dataset made of 120 millions observations. Let us focus on the following result: in the simulation study, we compare prediction performance of the 5 RF variants and more precisely we compute on one hand a test error (on a simulated test set, independent from the learning set), denoted errTest, and on the other hand the OOB error. The OOB error was calculated in a way adapted to the Big Data RF variant (meaning that it does not use all available data but only a subset, corresponding to a subsample or a partition set), denoted BDerrForest, and also with in a standard way (see Definition 2.2) using the entire learning set, denoted errForest[1].

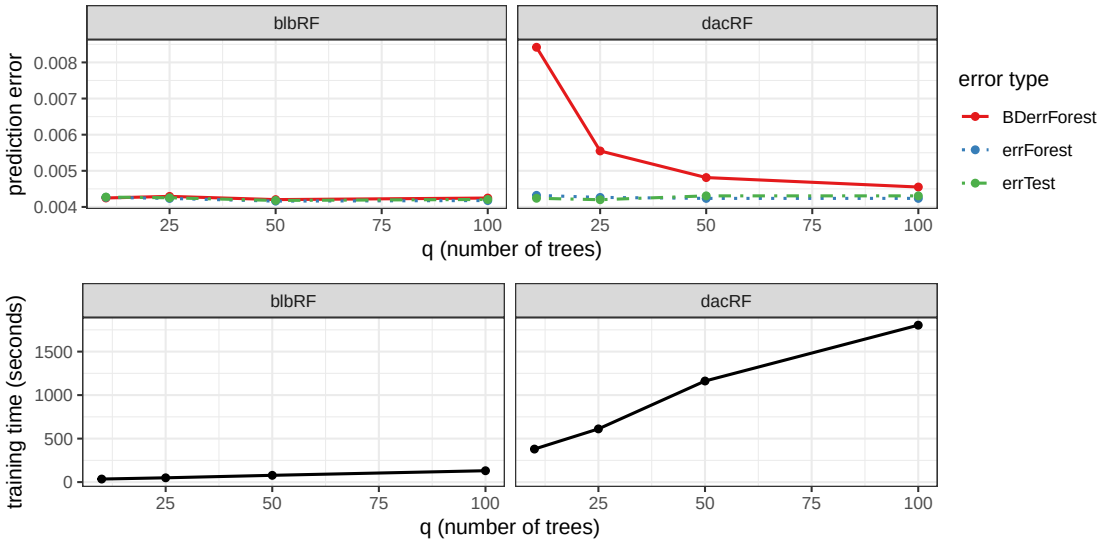

Figure 3.4: Evolution of the prediction error (top) and computational time for training (bottom) versus q. q is the number of trees in each sub-sample for blbRF, the Bag of Little Bootstrap RF-RI (left) or the number of trees in each chunk for dacRF, the "divide an conquer RF" (right). K is set to 10 and is the number of subsample for blbRF or the number of sets of the partition for dacRF.

One striking result illustrated in Figure 3.4 is that the Bag of Little Bootstrap RF are remarkably stable when the number of trees q varies (and this remains true when other parameters are modified) and their computational time remains very low even with high number of trees. On the other hand, the divide and conquer RF, even if they also reach low prediction error (in terms of test error or standard OOB

[1]The code of the different Big Data RF variants and of the simulation study is available at: `https://github.com/tuxette/bigdatarf`

error), are computationally demanding, and their intrinsic estimated OOB error is highly biased, especially when the number of trees is low. Indeed, BDerrForest only begins to fairly estimated the prediction error when q is set to 100, but with a high computational cost.

This focus illustrates the fact that some RF variants adapted to Big Data have to be applied with care, and that their parameters must be tuned properly, even in distributed environments such as Hadoop or Spark.

Chapter 4

Embedding Random Forests

In this chapter, I show how RF, and more precisely RF-RI and VSURF (detailed in Chapter 3), can be used together with other statistical methods to tackle specific problems. In the first section, the problem is to select groups of correlated variables in a classification framework, when the group structure is a priori unknown. RF-RI are thus combined with a clustering of variables method in this work (Chavent et al., 2019). While in the second section, the goal is to analyze high-dimensional longitudinal data, and RF-RI are embedded in an EM algorithm in order to estimate all quantities of interest from a semi-parametric mixed model (Capitaine et al., 2020b).

4.1 Combining Random Forests and Clustering of Variables

In this work, we address the problem of prediction and variable selection in an high-dimensional classification context, but with an additional group structure for input variables. Hence, we assume that some input variables are related with each others (so form groups of variables), and that we can also have some input variables independent from all other variables. This assumption is quite realistic, e.g., in omics data where gene expressions or protein abundances are often measured for every genes or proteins whereas some of them are involved in the same biological pathways, and hence highly correlated with each other. We stress that in our approach the group structure (which input variable belongs to which group) is a priori unknown. Thus, the proposed method differs from group-Lasso (Yuan & Lin, 2006) or group-sPLS (Liquet, Micheaux, Hejblum, & Thiébaut, 2015) techniques (among others), that must know the groups in advance to be applied.

Managing to select groups of informative variables can be interesting both in terms of prediction performance: if the redundant information brought by several

highly related variables is well summarized it can help the learning task; and also in terms of interpretation: in addition to the fact that some variables are selected we also add the information on how those variables are related to each other (and all of this in one integrated method).

To reach those objectives, we propose a combination between a clustering of variables (denoted CoV) method (Chavent, Kuentz-Simonet, Liquet, & Saracco, 2012) and the VSURF procedure (described in Section 3.2). The main principle of CoV is to sort input variables into homogeneous clusters, and to summarize variables belonging to the a cluster by a synthetic variable obtained with the first principal component applied only on variables of that cluster (in general, when variables can be continuous or categorical, the PCAmix algorithm (Chavent, Kuentz-Simonet, Labenne, & Saracco, 2017) is used). In the following, we use a hierarchical clustering of variables algorithm, which builds nested partitions and is naturally associated to a tree (or a dendrogram). We propose the following method that we call CoV/VSURF[1]:

a) Groups of informative variables selection:

- Apply CoV on input data to obtain a hierarchy (a tree) of variables.
- For each $K = 2, \ldots, p$, cut CoV tree in K clusters, train a RF-RI with the K synthetic variables $f^1, \ldots, f^K$ as predictors and $(y_1, \ldots, y_n)$ as outputs and compute its OOB error rate.
- Choose the optimal number K^* of clusters, which leads to the minimum OOB error rate. Cut CoV tree in K^* clusters.
  Perform VSURF with the K^* synthetic variables $f^1, \ldots, f^{K^*}$ as predictors and $(y_1, \ldots, y_n)$ as outputs. Denote by $m \leq K^*$ the number of selected informative synthetic variables (corresponding to the interpretation set of VSURF).

b) Prediction of a new observation:

- Train a random forest, $\hat{f}$, on the dataset consisting of the m selected synthetic variables and outputs.
- Compute the scores of the new observation on the m selected synthetic variables and predict its class label using $\hat{f}$.

To illustrate the procedure, we plot (Fig. 4.1) OOB error rate of RF-RI according to CoV partition cardinal (from the partition into 2 groups to the partition in p groups) for a learning set associated to a simulation study for which $n = 60$ and

[1] An R package implementing our approach is available at: `https://github.com/robingenuer/CoVVSURF`

$p = 120$. Input variables were simulated in order to get 9 groups of correlated variables and additional noise variables. The dashed vertical line indicates the optimal choice of partition in terms of prediction error and in this case it leads to $K^* = 10$, fairly retrieving the group structure of input variables. VSURF applied on the $K^* = 10$ associated synthetic variables selects $m = 4$ clusters, which correspond to the 4 most informative clusters of this simulation.

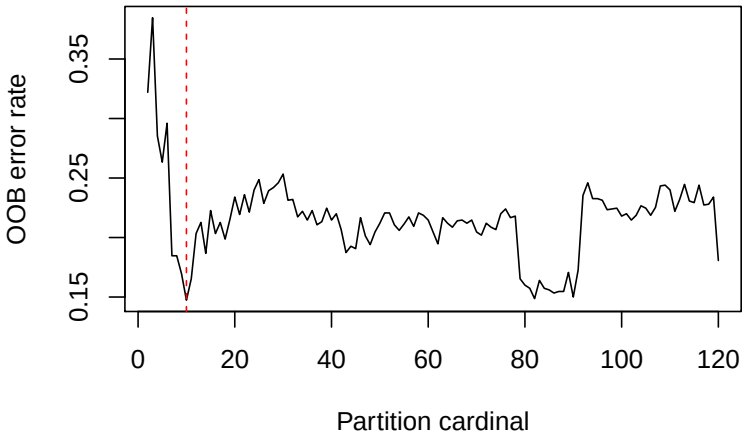

Figure 4.1: Random Forests OOB error rate according to CoV partition cardinal for a simulated learning dataset with $n = 60$ observations and $p = 120$ variables. The dashed red vertical line corresponds to $K^* = 10$ clusters.

The main feature of this algorithm is that it outputs a list of m selected informative synthetic variables. Since each synthetic variable is built on a subset of original variables, the algorithm implicitly leads to select groups of original variables.

Another interesting feature of the procedure is that even if the ascendant hierarchical clustering of variables algorithm is *unsupervised* (in the sense that it does not use outputs), the final variables partition is *supervised* since the number of clusters is optimized in terms of RF-RI classifier prediction error. This choice is justified by the fact that our main goal is prediction, so we are primarily interested in groups of informative variables, rather than groups of all variables. We stress that each group of informative variables is summarized by its synthetic variable (the first principal component of the group). These synthetic variables are then used to build the predictive model. Since, the clustering is optimized with a prediction criterion, informative variables should be well represented by their associated synthetic variables.

In Chavent et al. (2019), after leading a simulation study (with varied num-

bers of individuals, variables, groups, sizes of those groups and correlation between variables) which illustrates the good behavior of the proposed method in terms of informative groups of variables retrieval, we applied the method to a proteomic dataset coming from a clinical trial including $n = 44$ patients. The patients had a rectum cancer and undertook a treatment of chemotherapy and radiotherapy, before a surgery intervention. The main goal of this study was to predict if a patient will respond favorably to the treatment. Our approach managed to highlight 4 informative groups of variables (peptides in this application) gathering a total of 143 among the 4786 initial peptides abundances. For comparison, a standalone application of VSURF lead to a selection of 35 peptides. Thus, in this example, VSURF gave a sparse variable selection, but without group structure, and potentially could miss variables that are too redundant with already selected variables.

4.2 Embedding Random Forests in an EM Algorithm

In this section, we consider the problem of analyzing longitudinal data. Those data are very common, especially in health domain where variables are very often measured several times, e.g., during the follow-up of subjects or patients of a study. In this work (Capitaine et al., 2020b), we tackle the problem of analyzing high-dimensional longitudinal data, where we get repeated measurements of a large number of variables. The main principle is to use ideas from RF which behave well in high-dimension coupled with mixed effects models which handle repeated measurements. Following previous contributions from Sela & Simonoff (2012), Hajjem, Bellavance, & Larocque (2011), and Hajjem, Bellavance, & Larocque (2014) we use a semi-parametric mixed effects model and estimate all quantities of interest with an EM algorithm where the non-parametric part is estimated by RF-RI. Moreover, we extend previous models by adding a stochastic process, which is particularly useful in an high-dimensional context where it is not possible to put random effects on all variables nor easy to choose which variables get one. Thus, we introduce the following semi-parametric stochastic mixed effects model:

$$Y_{ij} = f(X_{ij}) + Z_{ij}b_i + \omega_i(t_{ij}) + \varepsilon_{ij} \tag{4.1}$$

where Y_{ij} (for all $i = 1, \ldots, n$ and $j = 1, \ldots, n_i$) is the response of the ith individual at time t_{ij}, X_{ij} is the $p \times 1$ vector of covariates, $f : \mathbb{R}^p \longrightarrow \mathbb{R}$ is the unknown mean behavior function, b_i is a $q \times 1$ vector of random effects associated with a $1 \times q$ vector of covariates Z_{ij}, $\omega_i(t)$ is a stochastic process used to model serial correlation and ε_{ij} denotes a measurement error.

In addition, we suppose that for all $i = 1, \ldots, n$ the b_i are independent, as well as the $\omega_i(t)$. And the ε_{ij} are also independent for all $i = 1, \ldots, n;\ j = 1, \ldots, n_i$. We assume that $b_i, \omega_i(t)$ and ε_{ij} are mutually independent. We also suppose that the ε_{ij} are normally distributed as $\mathcal{N}(0, \sigma^2)$, the b_i are normally distributed as $\mathcal{N}(0, B)$ where B is a $q \times q$ positive definite matrix and $\omega_i(t)$ is a centered Gaussian process.

We note that in classical linear mixed models (Laird & Ware, 1982), the fixed part $f(X_{ij})$ of our model is replaced by a linear combination of input variables. However those models are not adapted to the high-dimensional context we consider, where the number of input variables p is larger than n the number of individuals and even larger than $N = \sum_{i=1}^{n} n_i$ the total number of observations.

To estimate all parameters of the model and also the mean behavior function f (the non-parametric part of the model), the idea is to use an EM (Expectation-Maximization)-like algorithm (McLachlan & Krishnan, 1997) as follows:

- Initialization: Let $r = 0$, $\widehat{b}_{i,(0)} = 0_q$, $\widehat{\omega}_{i,(0)} = 0_{n_i}$, $\widehat{B}_{(0)} = I_q$, $\widehat{\gamma}^2_{(0)} = 1$ and $\widehat{\sigma}^2_{(0)} = 1$.
- Repeat, until convergence:
  1. Set $r = r+1$, compute $\widetilde{Y}_{ij,(r-1)} = Y_{ij} - Z_{ij}\widehat{b}_{i,(r-1)} - \widehat{\omega}_{ij,(r-1)}$ and estimate f in the standard regression framework (with all N observations):

     $$\widetilde{Y}_{ij,(r-1)} = f\left(X_{ij}\right) + \varepsilon_{ij}$$

     to get $\widehat{f}_{i,(r)}$. Then predict $\widehat{b}_{i,(r)}$ and $\widehat{\omega}_{i,(r)}$ using $\widehat{B}_{(r-1)}, \widehat{\gamma}^2_{(r-1)}, \widehat{\sigma}^2_{(r-1)}$ and $\widehat{f}_{i,(r)}$.
  2. Update $\widehat{B}_{(r)}$, $\widehat{\gamma}^2_{(r)}$ and $\widehat{\sigma}^2_{(r)}$ using $\widehat{f}_{i,(r)}$, $\widehat{b}_{i,(r)}$ and $\widehat{\omega}_{i,(r)}$.

The main principle of this procedure is to iterate between estimation/prediction of all quantities of interest with fixed variance parameters, and update of variance parameters given current estimations/predictions of the mean behavior function, random effects and stochastic process. We note that at step 1 of the loop, the estimation of f could be performed with any statistical method, but we focused on tree-based methods in our work. The main idea of step 1, is that if random effects and stochastic process are well predicted, then the dependence structure between observations from the same individual will be well modeled, hence removing those predicted random parts of the model from outputs Y_{ij} will lead to quite independent observations. And those observations $\widetilde{Y}_{ij}$ can be handled, e.g., by RF-RI.

In this work, we introduce a method generalizing previous methods, called SREEMforest for Stochastic Random Effects-EM forest. The name was inspired

from REEMtree (Sela & Simonoff, 2012), and the fact that we generalize the method with an aggregation of randomized REEMtrees and the addition of a stochastic process. The main characteristic of REEMtree is that in step 1 of the procedure, it starts by building a CART tree, but then update the leaves values by taking into account intra-individual covariance (through a linear mixed effects model where the indicator matrix giving which observations belong to each leaf plays the role of fix effects matrix). The way we randomize individual tree is then the same as in RF-RI with the bootstrap resampling preceding trees building and the random selection of d variables at each node of each tree.

Of course the two steps of the loop are strongly related and in our experiments we observed that RF-RI must be tuned carefully, specifically the number d of variables randomly selected at each node, to ensure convergence of the algorithm. To illustrate this phenomenon, we plot in Figure 4.2 the log-likelihood according to the number of iterations of SREEMforest applied to a vaccine trial (called DALIA) dataset including 19 HIV-infected patients, where input variables correspond to 32979 gene transcripts, and the variable to predict is the HIV viral load. As it can be seen, if d (named mtry as in the `randomForest` R package here) is set to a relatively low value ($\sqrt{p}$) the log-likelihood is even decreasing. Thus, the best results and stability are obtained for quite large values (at least $2p/3$ in this example).

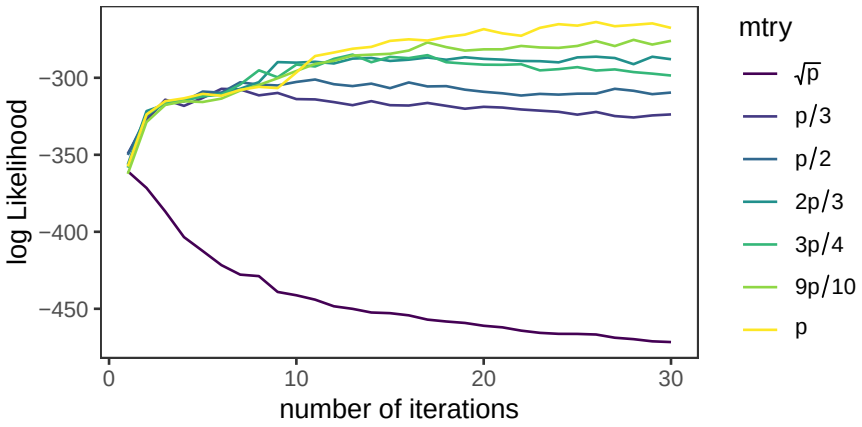

Figure 4.2: Log-likelihood according to the number of iterations in SREEMforest, DALIA trial.

Furthermore, we compare the previously mentioned methods in a simulation study and emphasize the gain of taking into account repeated measurements by also comparing prediction error with standard CART and RF-RI methods. Note that we implement our approach in the R package `LongituRF` (Capitaine, 2020), which also integrates the previously mentioned methods.

Finally, we propose to apply VSURF once the REEMforest procedure has converged to also select input variables in the longitudinal context. Hence, we developed a RF method able to handle high-dimensional longitudinal data and which performs in a final step a variable selection. In the case of the DALIA trial, the overall procedure outputs 21 gene transcripts which were biologically relevant in this application. However, we emphasize that to apply REEMforest (or other competitors) to the DALIA dataset, we had to consider both gene expression data and viral load data on the same time points: those after the antiretroviral treatment interruption of patients. Therefore, the objective is not the same as the one mentioned in Section 1.3.3: with SREEMforest we study the link between gene expression and viral load both measured after the treatment interruption. As we will see in Section 6.1, the primary objective of being able to predict viral load after treatment interruption using gene expression before the interruption can be addressed by Fréchet Random Forests.

Chapter 5

Purely random forests

In this chapter, I present theoretical works, which aim at explaining the good behavior of RF methods, in the particular case of Purely Random Forests (PRF). In PRF, the partition of the input space is obtained independently of the learning set, which allows to ease the theoretical analysis of RF. In this framework, we show that it is possible to derive asymptotic results illustrating that RF estimators are better than individual trees estimators. In the first section, I discuss several choices from the literature for obtaining the input space partition randomly. In the second section, I give some details about a first result of variance reduction brought by RF (Genuer, 2012) and other results on a bias focused analysis (Arlot & Genuer, 2014).

5.1 Different Purely Random Partitioning Schemes

In this chapter we focus on the standard regression problem, where $\mathcal{X} = \mathbb{R}^p$ and $\mathcal{Y} = \mathbb{R}$. Purely Random Forests (PRF) were first introduced in Breiman (2000), already in order to simplify theoretical of RF methods. In this document, we choose the following definition for PRF:

Definition 5.1 (Purely Random Forests)**.** Let $\mathbb{U}_1, \ldots, \mathbb{U}_q$ be q i.i.d. random finite measurable partitions of $\mathcal{X}$ that are independent with $\mathcal{L}_n$, and let $\widehat{h}_{\mathrm{PRT}}(., \mathbb{U}_1), \ldots, \widehat{h}_{\mathrm{PRT}}(., \mathbb{U}_q)$ be tree predictors associated with partitions $\mathbb{U}_1, \ldots, \mathbb{U}_q$ respectively. The aggregated predictor of this collection of trees, $\widehat{h}_{\mathrm{PRF}}(.)$, is called a purely random forest.

Remark. This definition is indeed a particular case of Definition 2.1 since individual trees depend on i.i.d. random variables, which are for PRF the entire input space partitions.

We emphasize that each individual tree $\widehat{h}_{\text{PRT}}(., \mathbb{U}_\ell)$ still depends on $\mathcal{L}_n$, since the mean values of outputs, over observations belonging to a set of partition $\mathbb{U}_\ell$, is allocated to this set (as usual in regression). The fundamental characteristic of PRF is that the patition of the input space is performed independently of the data $\mathcal{L}_n$.

Finally, we note that we choose this general definition for PRF, and consider several PRF variants which differ from each other by the way the random partitions are obtained. However, in the literature, PRF sometimes denote a particular case: e.g., in Breiman (2000) and Biau, Devroye, & Lugosi (2008), PRF correspond to the UPBRF variant defined below.

We now detail several PRF variants that have been introduced and theoretically analyzed in the literature, by defining their random partitioning of $\mathcal{X}$ scheme. From now on, we assume that $\mathcal{X} = [0, 1]^p$.

Definition 5.2 (Unbalanced Purely Random Forests (UBPRF))**.** To build one individual tree of UBPRF:

1. Put $[0, 1]^p$ at the root of the tree.
2. Repeat $k - 1$ times:
   a) Randomly choose a terminal node t to be splitted, uniformly among all terminal nodes.
   b) Randomly choose a split variable X^j, uniformly among the p input variables.
   c) Randomly choose a split point s uniformly over the j-th direction of t and perform the split $\{X^j \leq s\} \cup \{X^j > s\}$ to obtain the two children nodes of t.

This variant was introduced by Breiman (2000) and further analyzed in Biau et al. (2008). We call it *unbalanced* PRF, because the choice of the next terminal node to split is done uniformly among all terminal nodes of a tree and thus induces unbalanced trees. This contrasts with the following *balanced* PRF variant.

Definition 5.3 (Balanced Purely Random Forests (UBPRF))**.** To build one individual tree of BPRF:

1. Put $[0, 1]^p$ at the root of the tree.
2. Repeat $\log_2(k)$ times:
   - For every terminal node t:

a) Randomly choose a split variable X^j, uniformly among the p input variables.

b) Randomly choose a split point s uniformly over the j-th direction of t and perform the split $\{X^j \leq s\} \cup \{X^j > s\}$ to obtain the two children nodes of t.

For BPRF, the depth is the same for all branches of the tree, since at each iteration every terminal nodes of the same generation are split. The resulting trees are then balanced in this sense. The BPRF was introduced in Arlot & Genuer (2014).

Definition 5.4 (Purely Uniformly Random Forests (PURF))**.** To build one individual tree of PURF:

1. Put $[0,1]^p$ at the root of the tree.

2. Repeat $k-1$ times:

   a) Randomly choose a terminal node t to be splitted, each with a probability equal to its volume.

   b) Randomly choose a split variable X^j uniformly among the p input variables.

   c) Randomly choose a split point s uniformly over the j-th direction of t and perform the split $\{X^j \leq s\} \cup \{X^j > s\}$ to obtain the two children nodes of t.

The PURF variant was introduced in Genuer (2012) in the particular case of $p = 1$, with the simpler, and yet equivalent, following formulation: draw $k-1$ random variables $\xi_1, \ldots, \xi_{k-1}$ with uniform distribution on $[0,1]$ and consider the obtained partition in k sets:

$$\mathbb{U} = \{[0, \xi_{(1)}), \ldots, [\xi_{(k-1)}, 1)\}$$

where $\xi_{(1)} < \cdots < \xi_{(k-1)}$ denotes the corresponding order statistics.

We illustrate the different partitioning schemes in Figure 5.1, in a particular setting with $p = 2$ and $k = 64$. We plot one realization of random partition for each previously introduced PRF variant and we also add a plot for a TOY model. The partition of the TOY model is simply obtained by randomly translate (to the bottom and to the left) a regular partition of the unit square made of $\sqrt{k} \times \sqrt{k}$ squares.

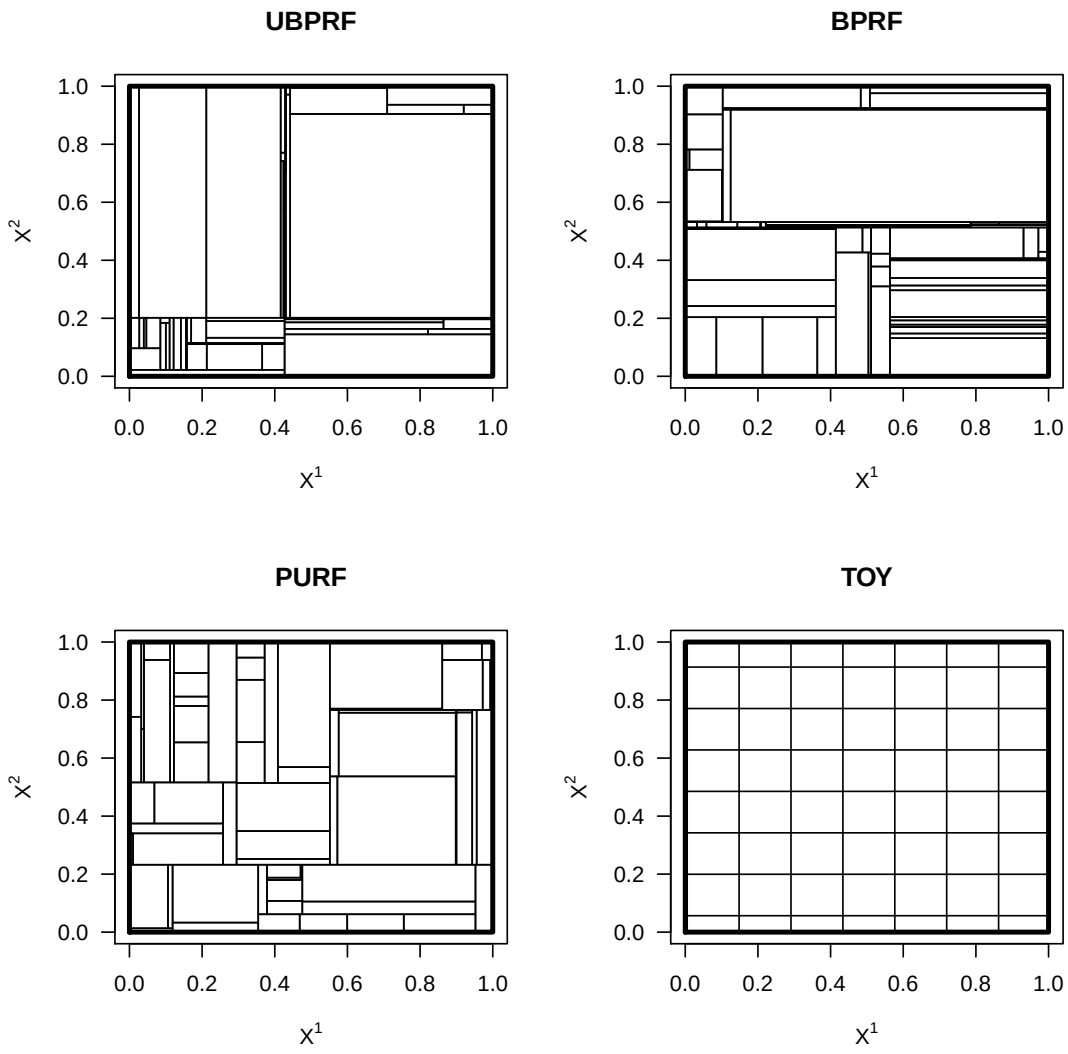

Figure 5.1: Partition of the unit square obtained by four different random partitioning scheme: three PRF variants and one TOY random translation of the regular grid. All partitions are made of 64 sets.

As it can be seen, the choice of the next terminal node to split has a great impact of the resulting partition. Actually the four partitions plotted are sorted (from left to right and then from top to bottom), from the least to the best distributed partition. In other words, the TOY partition has sets in all regions of space (since it is very close to a regular partition), and as we move from PURF to UBPRF, we see that more and more numerous and large regions has not been split. This remark will help understand the rates of convergence obtained in the

next section.

Before presenting our asymptotic analysis, we mention other partitioning scheme. First, Mondrian Forests introduced in Lakshminarayanan, Roy, & Teh (2014) are another example of PRF, with a random partitioning based on Mondrian processes (Roy & Teh, 2008). On the practical side, some works have also been made to compare performance of PRF-like methods (or close to it) with e.g. RF-RI, the state-of-the-art method. Cutler & Zhao (2001) introduced Perfect Random Tree ensemble which are close to UBPRF, while Geurts et al. (2006) studied Extremely Randomized Trees (ERT) which lay between UBPRF and RF-RI methods: in ERT, to split a node, d input variables are randomly selected, then one split point is chosen uniformly in the node in each of the d selected directions, and the final split the one that maximize heterogeneity decrease among the d obtained splits. Hence, in the particular case of $d = 1$, ERT is equivalent to UBPRF.

5.2 Theoretical Results

The first asymptotic analysis was sketched by Breiman (2000) and precisely conducted and completed by Biau et al. (2008). Those results concern UBPRF, and it is shown that both trees and forests are consistent, when they are building with this partitioning process. Thus, if the number of observations in $\mathcal{L}_n$ grows to infinity, trees and forests estimators converges to the true regression function.

Next, in Genuer (2012), we studied PURF and show, at least when $p = 1$ that for this variant, trees and forests reach the minimax rate of convergence for the Lipschitz functions class. Furthermore, we emphasized a gain brought by forests when compared to trees in the variance of the corresponding estimators. More precisely, we showed that the variance of forests are less than $3/4$ times the variance of trees. This was, up to our knowledge, the first theoretical result explicitly showing that forests perform better than trees (which is almost always observed in practice).

A focus on approximation error was then conducted in Arlot & Genuer (2014), in which UBPRF, BPRF and PURF variants, and a TOY variant of PRF associated to the TOY partitioning scheme introduced in the previous section, were analyzed. In this work, we prove that if we consider more regular regression function, forests are even better than trees, because they reach faster rates of convergence. More precisely, if the regression function is assumed to be C^2 (twice differentiable with a second derivative continuous), then the forests approximation error rate of convergence towards zero is twice the one of trees. And this results holds for

the four PRF variants analyzed. Moreover, an interesting result of our analysis is that we found different rates of convergence for the different PRF variants. More precisely, we showed that PURF and TOY reach minimax rates of convergence for the class of C^2 functions whereas BPRF converge slower and UPBRF even slower. The plots of Figure 5.1 and previous remarks help to give an intuition about this result. To help even more, we add the same plots but with $k = 1024$ (Fig. 5.2).

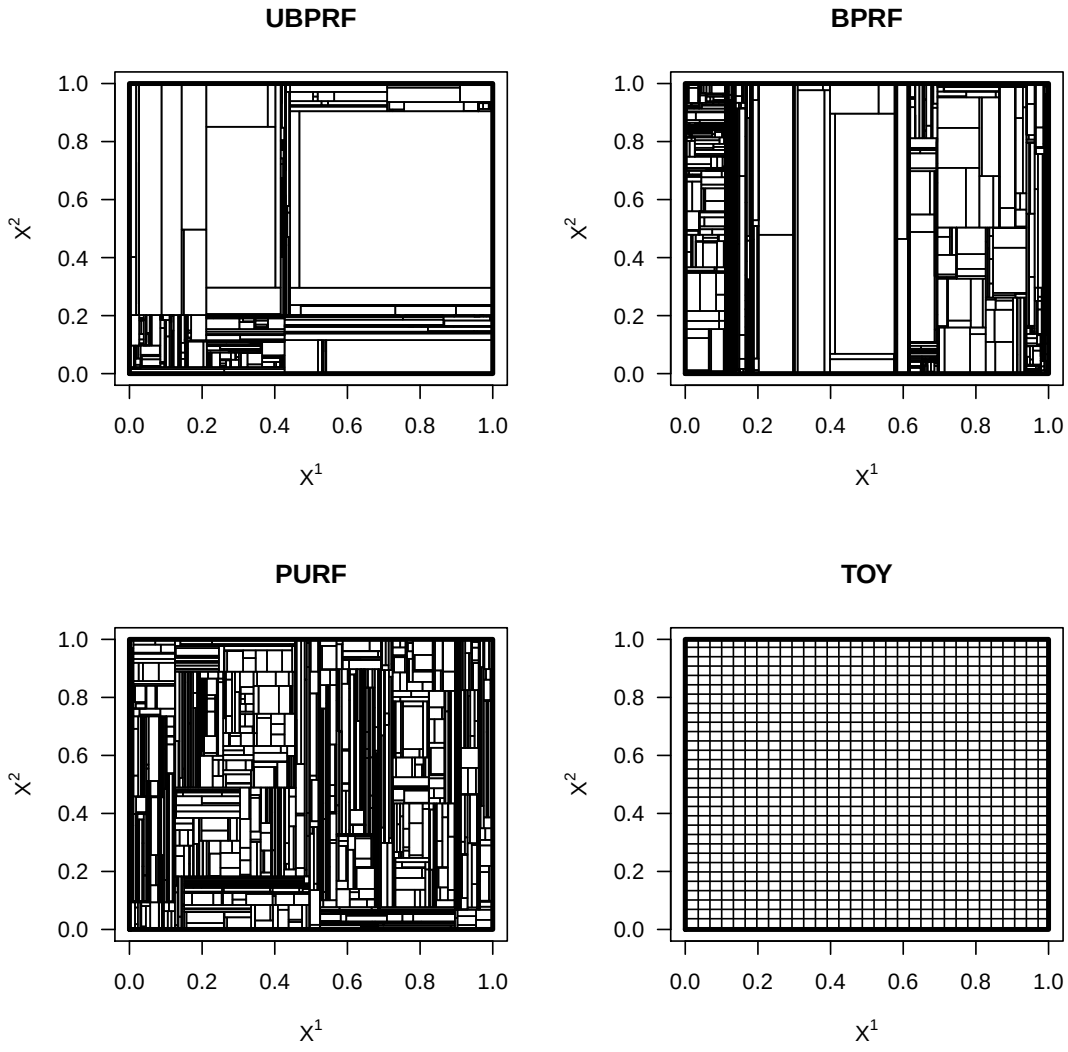

Figure 5.2: Partition of the unit square obtained by four different random partitioning scheme: three PRF variants and one TOY random translation of the regular grid. All partitions are made of 64 sets.

We thus can see that even when k increases, there remain quite large sets in the partition obtained by UBPRF and to lesser extend for BPRF. However, we see that the PURF partition is more satisfactory in terms of space exploration.

Of course, those remarks only illustrate individual partitions and hence apply more directly to trees, but as our result showed, the relative comparison in terms of rates of convergence between PRF variants are of the same order for trees and forests.

In Arlot & Genuer (2016), we investigate the impact of subsampling and randomization of the partitions, precisely for the TOY model and with numerical experiments for what we call "Hold-Out" RF (HORF). The principle of HORF is to first randomly split the learning set $\mathcal{L}_n$ into two subsamples $\mathcal{L}_n^1$ and $\mathcal{L}_n^2$; then a RF-RI is applied on $\mathcal{L}_n^1$ to get the partition of the input space $\mathcal{X}$, but then for each terminal node of each tree of the forest, the associated value is replaced by the mean of output values from observations of $\mathcal{L}_n^2$ belonging to the node. Hence, $\mathcal{L}_n^1$ is only used to build the partitions of $\mathcal{X}$ and $\mathcal{L}_n^2$ is only used to calculate the labels associated to the cells of the partitions. Thus, HORF are close to a PRF variant, since the partitions are independent of data used to labeled them. However the partitions themselves are not independent from each other, since they are built using the same data $\mathcal{L}_n^1$. Nevertheless, they are a good middle point between PRF and RF-RI.

For both TOY and HORF we have illustrated that of course at least one source of randomization is mandatory to reduce quadratic risk of RF compared to trees. Moreover, the best performance arise when both subsampling and randomization of the partitions are done. And finally, if we have to choose only one source of randomization, it seems that randomizing the partitions would be the best choice. We note that this is in line with ERT (Geurts et al., 2006) where the authors chose to not perform any subsampling or resampling step.

We finish this section by pointing out recent results of the same nature that were obtained by Mourtada, Gaïffas, & Scornet (2020) for Mondrian forests. The author proved that Mondrian trees reach minimax rates of convergence for s-Hölder functions with $s \in (0, 1]$ whereas it holds for Mondrian forests for $s \in (0, 2]$. Thus, they illustrate better rates of convergence for forests compared to trees for more regular regression functions.

Chapter 6

Generalized Random Forests

This chapter is dedicated to two works that either develop or use generalized RF methods, in the sense that the need is to apply RF methods adapted to problems that are neither standard regression nor classification. In the first approach (Capitaine et al., 2020a), we develop a quite general RF method adapted to metric spaces valued input and output data, while in the second one (in progress in collaboration with Anthony Devaux and Cécile Proust-Lima), we apply existing random survival forests in a dynamic prediction of health events context.

6.1 Fréchet Random Forests

In this section, we assume that the input space $\mathcal{X} = (\mathcal{X}_1, d_1) \times \cdots \times (\mathcal{X}_p, d_p)$ is a product of p metric spaces and the output space $(\mathcal{Y}, d_{\mathcal{Y}})$ is also a metric space. Hence, the proposed generalization of RF detailed here can handle input data that are potentially all of different kinds, e.g., some coordinates can be functional variables, others image-structured variables, and finally others can be standard continuous or categorical variables. In addition, the output space is also a general metric space and thus can be potentially of different nature than inputs. In the following, we call heterogeneous data such data including input and output variables of different kinds.

The challenge is then to find a way to keep as many general characteristics of RF as possible in such a context. As shown in Chapter 2 we need to determine an adapted way of building individual trees, to randomize them and finally to aggregate them.

6.1.1 Fréchet mean and variance

The first idea is to use notions of mean and variance adapted to metric spaces, because those two notions are central in standard RF, both in the tree building process and the aggregation procedure. This motivates the use of Fréchet mean and Fréchet variance (Fréchet, 1948) which are indeed natural generalizations of mean and variance in metric spaces.

Definition 6.1 (Fréchet mean and variance)**.** Let $z_1, \dots, z_n \in (\mathcal{Z}, d)$ a metric space.

- The empirical Fréchet mean of $z_1, \dots, z_n$ is defined as:

$$\overline{Z}_n \in \operatorname*{argmin}_{z \in (\mathcal{Z}, d)} \frac{1}{n} \sum_{i=1}^{n} d^2(z_i, z)$$

- The empirical Fréchet variance of $z_1, \dots, z_n$ is thus defined as:

$$\mathcal{V}_n = \frac{1}{n} \sum_{i=1}^{n} d^2(z_i, \overline{Z}_n)$$

Remark. Note that even if in general the empirical Fréchet mean may not exist nor be unique, we assume from now on, that it does exist and is unique. On the other hand, the Fréchet variance is always unique. We also stress that the names Fréchet mean and Fréchet variance will always refer to Fréchet empirical mean and Fréchet empirical variance in this section.

6.1.2 Splitting rule

One key ingredient to define a RF method is to indicate the tree building process, and as emphasized in Chapter 2 it suffices to determine a splitting rule that permits to split a node of a tree into two child nodes.

We start by defining what we call a split for metric spaces input data. The main idea is that since, in general, input metric spaces are unordered, we can compare different points with each other only via the distances of those spaces.

Definition 6.2 (Split for metric spaces input data)**.** Let t be a node of a tree to split and $j = 1, \dots, p$ a variable index. To every couple $(c_{j,L}, c_{j,R}) \in (\mathcal{X}_j, d_j)^2$, we define a split of t along variable X^j as the following partition into $t_{j,L}$ and $t_{j,R}$, the left and right child nodes respectively:

$$\underbrace{\left\{x \in t : d_j(x^j, c_{j,L}) \leq d_j(x^j, c_{j,R})\right\}}_{t_{j,L}} \cup \underbrace{\left\{x \in t : d_j(x^j, c_{j,L}) > d_j(x^j, c_{j,R})\right\}}_{t_{j,R}}$$

Then, we fix the heterogeneity measure $\Phi(t)$ of a node t to be the Fréchet variance of outputs in node t:

$$\Phi(t) = \mathcal{V}_{n_t}(t) \ .$$

Finally, if we assume to have a split $(c_{j,L}, c_{j,R})$ for every input variable $X^j, j = 1, \ldots, p$, the optimized split is defined as:

$$\Delta(t) = \underset{1 \leq j \leq p}{\operatorname{argmax}} \left\{ \Phi(t) - \left(\frac{n_{t_{j,L}}}{n_t} \Phi(t_{j,L}) + \frac{n_{t_{j,R}}}{n_t} \Phi(t_{j,R}) \right) \right\}$$

where we recall that n_t denotes the number of observations contained in node t.

In other words, given splits for every input variables, the optimization of the splitting process is of the same kind of that using in CART, detailed in Section 3.1, replacing standard variance by Fréchet variance.

The last thing we need to define the splitting process is what we call a splitting function that associates to any data in a node a couple $(c_{j,L}, c_{j,R})$. More precisely, we need one splitting function for each input variable X^j (because input variables can be of different nature). For example:

- If a k-means algorithm is available in space $(\mathcal{X}_j, d_j)$, then the 2-means methods (k-means with $k = 2$) can be used as a splitting function for the corresponding input variable X^j.

- The application of the splitting rule close to the one introduced in Geurts et al. (2006), for the Extremely Randomized Trees method, can be used as a split function in general. Indeed, for any $j = 1, \ldots, p$, one can always randomly select S couples $(c^1_{j,L}, c^1_{j,R}), \ldots, (c^S_{j,L}, c^S_{j,R})$ at random among observations in a node t and then maximize over the S heterogeneity decreases, to get one split of node t for input variable X^j.

6.1.3 Fréchet tree

Once the splitting function is chosen, the splitting rule can be recursively applied to develop what we call a Fréchet tree. The development of the tree is carried on until Fréchet variance of outputs in a node is null. We get a maximal tree structure, and the final tree predictor is the associated partitioning predictor with values associated to each node obtained by computing Fréchet mean of outputs in that node.

In addition, we emphasize that even if we do not focus on pruning in this document, all steps of the pruning algorithm performed in CART has been generalized in our implementation of Fréchet trees.

Let us finally describe how predictions are obtained: any observation $x \in \mathcal{X}$ can be drop down a Fréchet tree by computing, at each node, the distance between x^j and the two components of the split couple $(c_{j,L}, c_{j,R})$ for the input variable X^j actually used to split the node. Observation x then goes to the left child node is $d_j(x^j, c_{j,L}) < d_j(x^j, c_{j,R})$ and the right child node otherwise, and so one until x reaches a leaf. The prediction of the output of x is then the value associated to this leaf.

An example of a Fréchet tree built on heterogeneous data including curves, images and scalars (i.e., standard continuous variables) as input variables, is given in Figure 6.1. In this example, the Fréchet distance (Fréchet, 1906) is used to compare curves (it compares their shapes), and the euclidean distance is used to compare images (thus the comparison is made pixel per pixel).

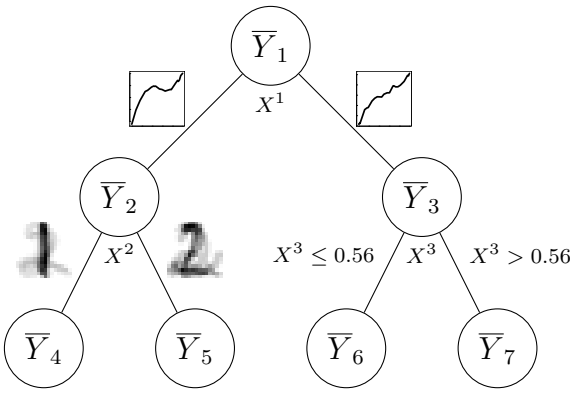

Figure 6.1: Example of a Fréchet tree built on heterogeneous data including curves (X^1), images (X^2) and scalars (X^3) for input variables. The $\overline{Y}_t$ associated to each node are the Fréchet mean of outputs in the node. Split variables are indicated under internal nodes, while split couples are represented on the left and right branches under the nodes.

6.1.4 Additional randomness and aggregation

Once the splitting function fixed, a collection of randomized Fréchet trees can be built and aggregated to get a Fréchet RF predictor. Most classical choices for additional randomness are still applicable in this framework. For example, bootstrap samples can be drawn to resample observations of $\mathcal{L}_n$ before Fréchet trees building, subsets of variables can be randomly selected at each node before optimizing the splits.

Since $(\mathcal{Y}, d_{\mathcal{Y}})$ is a metric space, once a collection of individual Fréchet trees is built, they are aggregated using Fréchet mean of individual predictions. For any

$x \in \mathcal{X}$, the predicted output is obtained as follows:

$$\widehat{y} = \underset{z \in \mathcal{Y}}{\operatorname{argmin}} \sum_{\ell=1}^{q} d_{\mathcal{Y}}^2(z, \widehat{h}(x, \Theta_l) \ .$$

Figure 6.2 illustrates the Fréchet RF method with both bootstrap and input variables subsets selection (as in RF-RI, see Fig. 3.1), hence a Fréchet tree RI is a Fréchet tree where at each node a subset of input variables is randomly selected before optimizing the split. We stress that, in addition to the choice of additional randomness, Fréchet RF depend on the choice of the splitting functions[1].

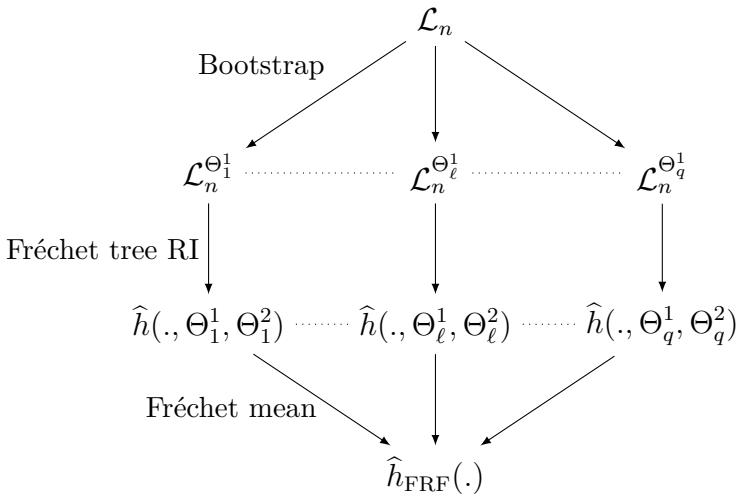

Figure 6.2: Scheme of Fréchet random forests.

Going back to the DALIA vaccine trial, with Fréchet RF we have been able to address the objective detailed in Section 1.3.3, that is predict the HIV viral load of patients after the interruption of their antiretroviral treatment, using the gene expression measured before this interruption. This is of primary interest, because it allows to rely the gene expression variations during the vaccination phase (before interruption) with the later response in terms of viral load. To do this, we apply Fréchet RF on those high-dimensional longitudinal dataset, by choosing the Fréchet distance for both gene expression input spaces and the viral load output space. Hence, we analyze the data with a functional approach: each evolution of a gene expression is considered as a curve, and the viral load evolution is also viewed as a curve. Furthermore, we choose the 2-means longitudinal method

[1]The R package `FrechForest` is available at: `https://github.com/Lcapitaine/FrechForest`

(Genolini et al., 2016) as the splitting function for all functional input variables. The results showed quite good predictions (the predicted curves were quite close in shape to the true viral load trajectories) and also permit to highlight very relevant genes associated to inflammation and T cell groups of genes.

Finally, we point out that a first consistency result has been obtained for Fréchet regressograms on data-driven partitions (input space was assumed to be $\mathbb{R}^p$, but $\mathcal{Y}$ is a general metric space), which lays the foundations for further theoretical analysis of this general method.

6.2 Using Random Survival Forests to Make Dynamic Prediction

In this ongoing work, in collaboration with Anthony Devaux and Cécile Proust-Lima, we aim at using Random Survival Forests (Ishwaran et al., 2008), denoted RSF, (among other prediction methods) to provide individual dynamic predictions of health events. The idea is to use all information available for an individual, including repeated measurements of potentially a lot of variables and other time-independent variables, to correctly predict the occurrence of an event (beginning or progression of a disease, death, etc.). The objective is, for example, to be able to adapt as soon as possible a treatment strategy of a given patient. Hence, we again have longitudinal data, as in Section 4.2, but the output is now a time-to-event.

Our proposed approach works in two steps, in line with landmark models (Van Houwelingen, 2007): the first step consists in modeling the longitudinal processes using only the information collected until a given time, called landmark time t_{LM}, and computing quantities summarizing trajectories of variable with repeated measurements; while in the second step, methods predicting the occurrence of the event are applied, for individuals still at risk at t_{LM}, using the previously calculated summaries and time-independent variables as input variables.

In this document, I only describe the methodology using RSF, but we also include other methods in our study, such as penalized Cox models (Goeman, 2010; Simon, Friedman, Hastie, & Tibshirani, 2011) and Sparse-Partial Least Square Cox models (Bastien, Bertrand, Meyer, & Maumy-Bertrand, 2015). The proposed methodology is thus the following:

1. Longitudinal modeling:

    a) Apply a generalized mixed model to each variable with repeated measurements using the information collected up to t_{LM}.

b) Compute summaries of the evolutions of those variables (current level, slope, etc.).

2. Event prediction:

a) Apply RSF with time-independent variables and summaries computed in 1.b) as input variables.

b) Compute the individual predictions using the RSF trained in 2.a).

Since RSF, as all RF methods, naturally handle high-dimensional data, the number of variables modeled in the first step and the number of summaries calculated for each of them, are not limited. This characteristic is the strength of our approach, the hope being that including more information can lead to more precise individual predictions.

6.2.1 Random Survival Forests

For every $i = 1, \ldots, n$, let $T_i^* = \min(T_i, C_i)$ be the observed event time, T_i being the event time and C_i the independent censoring time. We fix an horizon time, denoted t_{Hor}, and define the probability that a new subject has the event between t_{LM} and t_{Hor} as follows:

$$P(T \leq t_{\text{LM}} + t_{\text{Hor}} \,|\, T > t_{\text{LM}}, \Gamma(t_{\text{LM}}), X)$$

where T is the event time, $\Gamma(t_{\text{LM}})$ are the summaries of the trajectories of variables with repeated measurements calculated at the landmark time and X are the time-independent variables, all those for the new subject. The objective is now to estimate this probability using RSF.

In RSF, the splitting rule is adapted to survival data. Admissible splits are the same as in standard RF (see section 3.1): $\{X^j \leq s\}$ where s is a threshold for continuous variables or $\{X^{j'} \in A\}$ where A is a subset of categories for categorical variables. Those splits are compared using a criterion adapted to time-to-event data. We use here the classical log-rank test to split a node: the optimized split is the one that has the lowest log-rank test p-value, meaning that this split maximizes the difference in terms of survival between the two groups of individuals corresponding to the left and right child nodes. Thus, as the depth of a survival tree increases, the survival profiles of individuals belonging to a node becomes more and more homogeneous.

In each terminal node, a Nelson-Aalen estimator of the cumulative hazard function (CHF) is computed only using observations belonging to that node. Thus, given a new individual, the average of the q individual trees estimates of the CHF

is computed:

$$\widehat{\Lambda}_{\mathrm{RSF}}(t_{\mathrm{LM}} + t_{\mathrm{Hor}} \,|\, \Gamma(t_{\mathrm{LM}}), X) = \frac{1}{q} \sum_{\ell=1}^{q} \widehat{\Lambda}(t_{\mathrm{LM}} + t_{\mathrm{Hor}}, \Theta_\ell^1, \Theta_\ell^2 \,|\, \Gamma(t_{\mathrm{LM}}), X)$$

where $\widehat{\Lambda}$ denotes the Nelson-Aalen estimator of the CHF. The additional randomness is composed of bootstrap sampling before tree building and the random choice of d variables before optimizing the split of a node (as in RF-RI). Finally, the probability of occurrence of the event for the new individual is estimated by:

$$1 - \exp\left(-\widehat{\Lambda}_{\mathrm{RSF}}(t_{\mathrm{LM}} + t_{\mathrm{Hor}} \,|\, \Gamma(t_{\mathrm{LM}}), X)\right)$$

6.2.2 Preliminary results

In this ongoing work, we lead simulation experiments to study the behavior of the proposed approach, and also we compare the use of RSF in the second step with other prediction methods (CoxLasso and sPLScox)[2]. Our approach seems promising since it effectively handles data with many longitudinal variables, and the use of RSF is naturally interesting when the relationship between the time to event and input variables (summaries of trajectories of time-dependent variables and time-independent variables) are complex (non-linear relationships or interactions involved).

In addition to get individual predictions, we can also use the permutation-based variable importance score which can, as all RF methods, be computed for RSF. Hence, we can interpret the prediction results with the extra information of which variables are the most related to occurrence of the event.

[2] An R package implementing all those methods is currently in development.

Perspectives

A general perspective would be, using all knowledge that we already have on RF methods, to continue to derive new RF methods or adapt existing ones, in order to tackle more and more complex data analysis problems, associated to more and more massive, complex and heterogeneous collected data. As presented in Chapter 2, the main ideas of RF are general enough to be adapted in many different frameworks.

When analyzing high-dimensional and/or heterogeneous data, we have seen that the problem of variable selection is always of interest and can sometimes be the primary objective of the data analysis. Hence, as RF methods generalize and are applied to several frameworks, we also need variable selection techniques adapted to those contexts. VSURF could serve as an interesting basis for such developments since the procedure is mainly based on OOB error and Variable Importance, that are also easily generalized together with RF.

Together with this kind of methodological developments, it would also be very important to progress in tools to ease the interpretation of RF results. Indeed their non-parametric nature, which can be viewed as an advantage for their prediction capacity and generality, could limit their practical use. Some tools already exist, such as variable importance (and also partial plots for RF-RI, see the `partialPlot` function of the R package `randomForest`), but more work could be done in that direction.

Dynamic Predictions with a Tailored Random Forests

The last ongoing work presented in Section 6.2 is an interesting first work to provide individual predictions that make use of many longitudinal variables. However, a more integrated approach that directly use the raw repeated measurements (instead of their summaries) into a RF method could be more adapted and lead to a better use of all available data. We actually plan to address this question in the sequel of Anthony Devaux's thesis. This would surely be based on the introduction of a new way to split nodes adapted to the problem of the prediction of an event and the fact that lots of longitudinal variables are available.

Applications of Fréchet Random Forests

Fréchet random forests offer a very general framework to apply RF methods in many complex situations involving heterogeneous data. Therefore, it would be interesting to apply the method for different data analysis problems. Several informal discussions with colleagues, particularly from the Bordeaux Population Health research center suggest that interesting applications could be possible in the health domain. Indeed, in health studies, many data are collected and those are often of different nature: images from radiology or MRI, high-dimensional longitudinal data in omics... Those applications would surely require some works about the choice of the distance in different input subspaces and I obviously expect implementations and computational issues, but I think that there is room for challenging but interesting applied works with this methodology.

Random Forests Theory

On the theoretical side, the community is quite dynamic from a few years. Indeed, Scornet et al. (2015) obtained the first consistency results for a RF methods very close to RF-RI (the bootstrap step was replaced by a subsampling step), at the price of several assumptions on the regression function. Wager, Hastie, & Efron (2014) and Mentch & Hooker (2016) derived asymptotic normality results and proposed confidence intervals for RF predictions. Other RF methods were also analyzed: Denil et al. (2013) focused to an online RF method and prove its consistency, while Zhu, Zeng, & Kosorok (2015) obtained consistency and an upper bound on the convergence rate of their method Reinforcement Learning Trees. We refer to the very interesting review of Biau & Scornet (2016) for further reading on that matter, and we also point out more recent references: Athey, Tibshirani, & Wager (2019) introduced Generalized Random Forests and derived asymptotic results, Klusowski (2019) goes back to analyze CART and prove its consistency, among others.

There were also theoretical analysis concerning the variable importance scores provided by RF. Louppe, Wehenkel, Sutera, & Geurts (2013) characterized the Mean Decrease Impurity (MDI) score, which is based on the capacity for an input variable to decrease node heterogeneity when used to split nodes (averaged for all trees of a RF), while more recently Ramosaj & Pauly (2019) analyzed the permutation-based variable importance score. In both works, authors managed to prove, in some specific contexts, that those variable importance indices behave very well by associating scores strictly larger to input variables related to the output variable compared to noise variables (independent with the output).

All together (and with results presented in Chapter 5), a bunch of theoretical guaranties are now available and help to better understand RF methods and their

behavior in different situations. However, a lot remains to be done: e.g., study of the effect of the number of variables randomly picked at each node in RF-RI, derive consistency results in terms of variable selection when using variable selection procedure based on RF variable importance, further analyze Fréchet RF or even obtain rates of convergence for data-dependent RF (starting by considering HORF for example).

References

Arlot, S., & Genuer, R. (2014). Analysis of purely random forests bias. *arXiv Preprint arXiv:1407.3939*.

Arlot, S., & Genuer, R. (2016). Comments on: A random forest guided tour. *TEST*, *25*(2), 228–238.

Athey, S., Tibshirani, J., & Wager, S. (2019). Generalized random forests. *Annals of Statistics*, *47*(2), 1148–1178.

Bach, F. R. (2008). Bolasso: Model consistent Lasso estimation through the bootstrap. In *Proceedings of the 25th International Conference on Machine Learning* (pp. 33–40).

Banerjee, M., & McKeague, I. W. (2007). Confidence sets for split points in decision trees. *Annals of Statistics*, *35*(2), 543–574.

Bastien, P., Bertrand, F., Meyer, N., & Maumy-Bertrand, M. (2015). Deviance residuals-based sparse pls and sparse kernel pls regression for censored data. *Bioinformatics*, *31*(3), 397–404.

Biau, G., Devroye, L., & Lugosi, G. (2008). Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research*, *9*, 2015–2033.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST*, *25*(2), 197–227.

Bickel, P., Götze, F., & Zwet, W. van. (1997). Resampling fewer than n observations: Gains, losses and remedies for losses. *Statistica Sinica*, *7*(1), 1–31.

Boulesteix, A.-L., Janitza, S., Kruppa, J., & König, I. R. (2012). Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(6), 493–507.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140.

Breiman, L. (1998). Arcing classifier. *Annals of Statistics*, *26*(3), 801–849.

Breiman, L. (2000). *Some infinity theory for predictor ensembles*. Technical Report 579, Statistics Dept. UCB.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees*. Chapman & Hall, New York.

Bühlmann, P., & Yu, B. (2002). Analyzing bagging. *Annals of Statistics*, *30*(4), 927–961.

Cadenas, J. M., Garrido, M. C., & Martínez, R. (2013). Feature subset selection filter–wrapper based on low quality data. *Expert Systems with Applications*, *40*(16), 6241–6252.

Capitaine, L. (2020). *LongituRF: Random forests for longitudinal data*. Retrieved from `https://CRAN.R-project.org/package=LongituRF`

Capitaine, L., Bigot, J., Thiébaut, R., & Genuer, R. (2020a). Fréchet random forests for metric space valued regression with non euclidean predictors. *arXiv Preprint arXiv:1906.01741*.

Capitaine, L., Genuer, R., & Thiébaut, R. (2020b). Random forests for high-dimensional longitudinal data. *Statistical Methods in Medical Research*, *0*(0), 1–19.

Chavent, M., Genuer, R., & Saracco, J. (2019). Combining clustering of variables and feature selection using random forests. *Communications in Statistics - Simulation and Computation*, *0*(0), 1–20.

Chavent, M., Kuentz-Simonet, V., Labenne, A., & Saracco, J. (2017). Multivariate analysis of mixed data: The R package PCAmixdata. *arXiv Preprint arXiv:1411.4911*.

Chavent, M., Kuentz-Simonet, V., Liquet, B., & Saracco, J. (2012). ClustOfVar: An R package for the clustering of variables. *Journal of Statistical Software*, *50*(13), 1–16.

Chu, C., Kim, S., Lin, Y., Yu, Y., Bradski, G., Ng, A., & Olukotun, K. (2010). Map-Reduce for machine learning on multicore. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, & A. Culotta (Eds.), *Advances in Neural Information Processing Systems* (Vol. 23, pp. 281–288). Hyatt Regency, Vancouver,

Canada.

Clémençon, S., Depecker, M., & Vayatis, N. (2013). Ranking forests. *Journal of Machine Learning Research*, *14*(Jan), 39–73.

Cutler, A., & Zhao, G. (2001). Pert-perfect random tree ensembles. *Computing Science and Statistics*, *33*, 490–497.

Denil, M., Matheson, D., & Freitas, N. de. (2013). Consistency of online random forests. In *Proceedings of the 30th International Conference on Machine Learning* (pp. 1256–1264).

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, *40*(2), 139–157.

Dìaz-Uriarte, R., & Alvarez De Andres, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, *7*(1), 3.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *13th International Conference on Machine Learning* (Vol. 96, pp. 148–156).

Fréchet, M. R. (1906). Sur quelques points du calcul fonctionnel. *Rendiconti Del Circolo Matematico Di Palermo (1884-1940)*, *22*, 1–72.

Fréchet, M. R. (1948). Les éléments aléatoires de nature quelconque dans un espace distancié. *Annales de L'institut Henri Poincaré*, *10*(4), 215–310.

Genolini, C., Ecochard, R., Benghezal, M., Driss, T., Andrieu, S., & Subtil, F. (2016). kmlShape: An efficient method to cluster longitudinal data (time-series) according to their shapes. *Plos One*, *11*(6), e0150738.

Genuer, R. (2012). Variance reduction in purely random forests. *Journal of Nonparametric Statistics*, *24*(3), 543–562.

Genuer, R., Michel, V., Eger, E., & Thirion, B. (2010). Random forests based feature selection for decoding fMRI data. In *Proceedings compstat* (pp. 1–8).

Genuer, R., Poggi, J.-M., & Tuleau-Malot, C. (2015). VSURF: An R package for variable selection using random forests. *The R Journal*, *7*(2), 19–33.

Genuer, R., Poggi, J.-M., & Tuleau-Malot, C. (2019). *VSURF: Variable selection using random forests*. Retrieved from `https://CRAN.R-project.org/package=VSURF`

Genuer, R., Poggi, J.-M., Tuleau-Malot, C., & Villa-Vialaneix, N. (2017). Random forests for big data. *Big Data Research*, *9*, 28–46.

Genuer, R., Poggi, J., & Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern Recognition Letters*, *31*(14), 2225–2236.

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42.

Gey, S., & Nedelec, E. (2005). Model selection for cart regression trees. *IEEE Transactions on Information Theory*, *51*(2), 658–670.

Ghattas, B. (1999). Prévisions des pics d'ozone par arbres de régression, simples et agrégés par bootstrap. *Revue de Statistique Appliquée*, *47*(2), 61–80.

Goeman, J. J. (2010). L1 penalized estimation in the cox proportional hazards model. *Biometrical Journal*, *52*(1), 70–84.

Goldstein, B. A., Hubbard, A. E., Cutler, A., & Barcellos, L. F. (2010). An application of random forests to a genome-wide association dataset: Methodological considerations & new findings. *BMC Genetics*, *11*(1), 1.

Gregorutti, B., Michel, B., & Saint-Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis. *Computational Statistics & Data Analysis*, *90*, 15–35.

Grömping, U. (2015). Variable importance in regression models. *Wiley Interdisciplinary Reviews: Computational Statistics*, *7*(2), 137–152.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, *46*(1-3), 389–422.

Györfi, L., Kohler, M., Krzyżak, A., & Walk, H. (2002). *A Distribution-free Theory of Nonparametric Regression*. New York: Springer-Verlag.

Hajjem, A., Bellavance, F., & Larocque, D. (2011). Mixed effects regression trees for clustered data. *Statistics & Probability Letters*, *81*(4), 451–459.

Hajjem, A., Bellavance, F., & Larocque, D. (2014). Mixed-effects random forest for clustered data. *Journal of Statistical Computation and Simulation*, *84*(6), 1313–1328.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(8), 832–844.

Hothorn, T., Bühlmann, P., Dudoit, S., Molinaro, A., & Van Der Laan, M. J. (2006). Survival ensembles. *Biostatistics*, *7*(3), 355–373.

Ishwaran, H., Kogalur, U. B., Blackstone, E. H., & Lauer, M. S. (2008). Random survival forests. *Annals of Applied Statistics*, 841–860.

Kleiner, A., Talwalkar, A., Sarkar, P., & Jordan, M. (2014). A scalable bootstrap for massive data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *76*(4), 795–816.

Klusowski, J. M. (2019). Analyzing CART. *arXiv Preprint arXiv:1906.10086*.

Laird, N. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, 963–974.

Lakshminarayanan, B., Roy, D. M., & Teh, Y. W. (2014). Mondrian forests: Efficient online random forests. In *Advances in neural information processing systems* (pp. 3140–3148).

Lecué, G. (2007). *Aggregation procedures: optimality and fast rates* (PhD thesis). Université Pierre et Marie Curie - Paris VI.

Lévy, Y., Thiébaut, R., Montes, M., Lacabaratz, C., Sloan, L., King, B., ... others. (2014). Dendritic cell-based therapeutic vaccine elicits polyfunctional hiv-specific t-cell immunity associated with control of viral load. *European Journal of Immunology*, *44*(9), 2802–2810.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News*, *2*(3), 18–22.

Liquet, B., Micheaux, P. L. de, Hejblum, B. P., & Thiébaut, R. (2015). Group and sparse group partial least square approaches applied in genomics context. *Bioinformatics*, *32*(1), 35–42.

Louppe, G., Wehenkel, L., Sutera, A., & Geurts, P. (2013). Understanding variable importances in forests of randomized trees. In *Advances in Neural Information Processing Systems* (pp. 431–439).

Lugosi, G., & Nobel, A. (1996). Consistency of data-driven histogram methods for density estimation and classification. *Annals of Statistics*, *24*(2), 687–706.

McLachlan, G. J., & Krishnan, T. (1997). *The EM algorithm and extensions*. John Wiley & Sons.

Meinshausen, N., & Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *72*(4), 417–473.

Mentch, L., & Hooker, G. (2016). Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *Journal of Machine Learning Research*, *17*(1), 841–881.

Mourtada, J., Gaïffas, S., & Scornet, E. (2020). Minimax optimal rates for mondrian trees and forests. *Annals of Statistics*, *48*(4), 2253–2276.

Nobel, A. (1996). Histogram regression estimation using data-dependent partitions. *Annals of Statistics*, *24*(3), 1084–1105.

Prasad, A. M., Iverson, L. R., & Liaw, A. (2006). Newer classification and regression tree techniques: Bagging and random forests for ecological prediction. *Ecosystems*, *9*(2), 181–199.

Ramosaj, B., & Pauly, M. (2019). Asymptotic unbiasedness of the permutation importance measure in random forest models. *arXiv Preprint arXiv:1912.03306*.

Roy, D. M., & Teh, Y. (2008). The mondrian process. *Advances in Neural Information Processing Systems*, *21*, 1377–1384.

Saffari, A., Leistner, C., Santner, J., Godec, M., & Bischof, H. (2009). On-line random forests. In *Proceedings of IEEE 12th International Conference on Computer Vision Workshops* (pp. 1393–1400). IEEE.

Sanchez-Pinto, L. N., Venable, L. R., Fahrenbach, J., & Churpek, M. M. (2018). Comparison of variable selection methods for clinical predictive modeling. *International Journal of Medical Informatics*, *116*, 10–17.

Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, *12*(5), 1207–1245.

Scornet, E., Biau, G., & Vert, J. (2015). Consistency of random forests. *Annals of Statistics*, *43*(4), 1716–1741.

Segal, M., & Xiao, Y. (2011). Multivariate random forests. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *1*(1), 80–87.

Sela, R. J., & Simonoff, J. S. (2012). RE-EM trees: A data mining approach for longitudinal and clustered data. *Machine Learning*, *86*(2), 169–207.

Simon, N., Friedman, J., Hastie, T., & Tibshirani, R. (2011). Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of Statistical Software*, *39*(5), 1.

Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert*

Systems with Applications, *134*, 93–101.

Stone, C. J. (1977). Consistent nonparametric regression. *Annals of Statistics*, *5*(4), 595–620.

Stone, C. J. (1982). Optimal global rates of convergence for nonparametric regression. *Annals of Statistics*, *10*(4), 1040–1053.

Thiébaut, R., Hejblum, B. P., Hocini, H., Bonnabau, H., Skinner, J., Montes, M., ... Lévy, Y. (2019). Gene expression signatures associated with immune and virological responses to therapeutic vaccination with dendritic cells in hiv-infected individuals. *Frontiers in Immunology*, *10*, 874.

Thiébaut, R., Liquet, B., Hocini, H., Hue, S., Richert, L., Raimbault, M., ... Levy, Y. (2012). A new method for integrated analysis applied to gene expression and cytokines secretion in response to LIPO-5 vaccine in HIV-negative volunteers. *Retrovirology*, *9*(2), 121.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 267–288.

Van Houwelingen, H. C. (2007). Dynamic prediction by landmarking in event history analysis. *Scandinavian Journal of Statistics*, *34*(1), 70–85.

Verikas, A., Gelzinis, A., & Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern Recognition*, *44*(2), 330–349.

Wager, S., Hastie, T., & Efron, B. (2014). Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *Journal of Machine Learning Research*, *15*(1), 1625–1651.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(1), 49–67.

Zago, L., Hervé, P.-Y., Genuer, R., Laurent, A., Mazoyer, B., Tzourio-Mazoyer, N., & Joliot, M. (2017). Predicting hemispheric dominance for language production in healthy individuals using support vector machine. *Human Brain Mapping*, *38*(12), 5871–5889.

Zhu, R., Zeng, D., & Kosorok, M. R. (2015). Reinforcement learning trees. *Journal of the American Statistical Association*, *110*(512), 1770–1784.

École doctorale
Sociétés, politique, santé publique

université de BORDEAUX

Centre de recherche Bordeaux Population Health
Inserm U1219, Université de Bordeaux, 33000 Bordeaux, France.