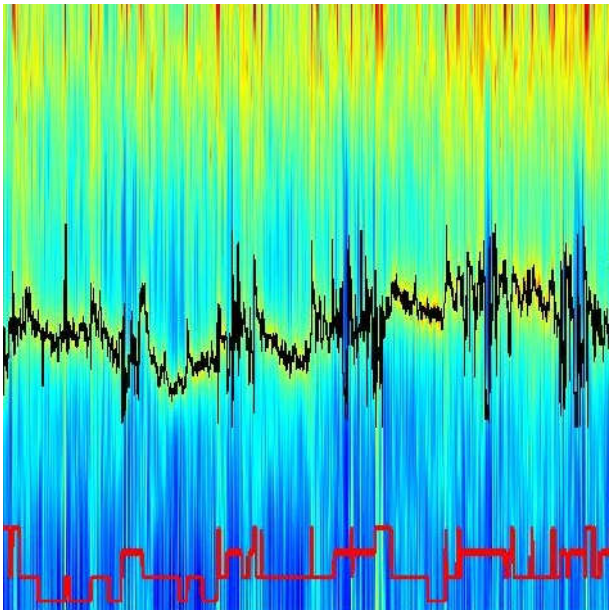
Transfer Learning through Kernel Alignment : Application to Adversary Data Shifts in Automatic Sleep Staging

Bruno Muller

Thèse
de doctorat
de l'UTT

Bruno MULLER

Transfer Learning through Kernel Alignment: Application to Adversary Data Shifts in Automatic Sleep Staging



Champ disciplinaire :
Sciences pour l'Ingénieur

2021TROY0037 **Année 2021**

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

THESE

pour l'obtention du grade de

DOCTEUR
de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

en SCIENCES POUR L'INGENIEUR

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Bruno MULLER

le 24 novembre 2021

Transfer Learning through Kernel Alignment: Application to Adversary Data Shifts in Automatic Sleep Staging

JURY

Mme R. LE BOUQUIN JEANNES	PROFESSEURE DES UNIVERSITES	Présidente
M. D. BRIE	PROFESSEUR DES UNIVERSITES	Rapporteur
M. P. HONEINE	PROFESSEUR DES UNIVERSITES	Rapporteur
M. P. BEAUSEROY	PROFESSEUR DES UNIVERSITES	Examinateur
M. A. VIOLA	DOCTEUR	Examinateur
M. R. LENGELLÉ	PROFESSEUR DES UNIVERSITES	Directeur de thèse

A mes muses :
A. Franquin, G. Remi, M. Audiard, G. Lautner
Ma machine à café
Ma famille et mes compagnons du quotidien

Acknowledgments

Mes premiers remerciements s'adressent à Régis Lengellé, mon directeur de thèse. Tous ces moments passés ensemble sont inestimables, que ce soient les moments d'inspiration, de désespoir occasionnel, ou simplement nos partages d'anecdotes, de curiosités ou de plaisanteries. J'espère que nous aurons encore longtemps le plaisir d'échanger et de collaborer.

J'adresse ensuite toute ma gratitude aux personnes ayant participé à l'évaluation de ces travaux, en particulier les membres du jury de thèse : les rapporteurs, Paul Honeine, professeur à l'université de Rouen-Normandie et David Brie, professeur à l'université de Lorraine - un grand merci pour vos retours, commentaires et suggestions, qui ont permis d'améliorer la qualité du présent manuscrit et ont inspiré certaines perspectives à ces travaux ; la présidente du jury, Régine Le Bouquin Jeannes, professeur à l'université de Rennes, Pierre Beauseroy, professeur à l'université de technologie de Troyes et Antoine Viola, docteur en neurosciences en charge du suivi de ces travaux au sein de l'entreprise PPRS - tous mes remerciements également pour vos questions, suggestions, commentaires et l'intérêt que vous avez porté à ce sujet de recherche. Je remercie également les relecteurs anonymes lors de l'évaluation des produits de cette recherche.

Je remercie l'entreprise d'accueil, PPRS, pour m'avoir offert cette belle opportunité de pouvoir effectuer mes recherches sur une application aussi intéressante que le sommeil, au travers du dispositif Somno-Art. J'adresse une pensée particulière à mes collègues, de qui j'ai pu longuement partager le quotidien - "les bons vieux jours" du futur ; je remercie en particulier mes collaborateurs directs : Antoine Viola, Gil Fuchs et Baptiste Planat, pour leurs conseils, encadrement, gestion des priorités et du projet de thèse ; Débora Kirscher, Laurie Thiesse, Valentin Dehouck et Julien Barascud, pour l'amitié qu'ils m'ont témoignée et leur soutien au quotidien. Je remercie les pôles administratifs pour la gestion et le suivi du projet doctoral : côté université de technologie de Troyes, Isabelle Leclercq, Pascale Denis et Thérèse Kazarian ; côté PPRS, Charlène Rahmoune, Agathe Leyssens et Denise Daller. Un grand merci ! J'exprime également toute ma gratitude à l'ensemble de mes collaborateurs des autres pôles et de la direction de PPRS.

Enfin, j'adresse mes remerciements les plus sincères à ma famille pour leur soutien inconditionnel : mes grands-parents Renée et René, Bernadette et Aloyse ; mes parents Michèle et Gérard ; mon frère Yann. Bien évidemment, je n'oublie pas mes amis, dont la liste est trop longue pour être citée intégralement ici. Merci à tous !

Contents

List of Figures

List of Tables

Introduction

Sleep staging through the gold standard of polysomnography (PSG) is heavy in requirements, expensive and time-consuming. For these reasons, alternatives to the PSG have been established over the years and the scoring of sleep has been automated thanks to machine learning.

PPRS − the company where this project takes place − has developed an alternative to the PSG: Somno-Art Solution, a medical device composed of a wristband (hardware) which records the heart rate through photoplethysmography and the movements through accelerometry, and of a software that analyses these recordings and scores sleep.

As for most physiological data, there is a strong dependence to individual parameters (age, sex, health) and to environmental influences (noise, light, stress, drug intake to cite some); we distinguish inter-individual variabilities, affected by individual parameters, and intra-individual effects, due to the latter influences. The combination of both these variabilities explains the differences in the data distribution of two nights of sleep, and of their association with the sought sleep stages (labels). These constitute adversary effects on the machine learning classification task, which may explain lower performances on part of the recordings.

One of the situations tackled by Transfer learning (TL) is the transfer of knowledge between a training night (source) of known labels, and a new night of unknown labels (target), where the classification task is unique and the differences between source and target lay in their domains (observations distribution and their conditional relation to the sought labels and their prior probabilities). This seems to be our situation, therefore we searched and developed methods of homogeneous transductive transfer learning (the specific sub-field of transfer learning tackling such situations). Furthermore, as there is a good number of recordings available in PPRS, we also extended our work to training *source* selection (TSS): for a given target recording, there should exist an ideal source detector which will result in improved classification performances. Part of our work is focused on the identification of good performances predictor that will help us select this ideal source detector.

As kernel methods yield practical properties, especially when dealing with great amounts of data thanks to the kernel trick, we chose to orient our researches in this direction. We have developed multiple kernel-based algorithms, which will be presented in this manuscript.

The latter is organised as follows: in Chapter (1) we give a thorough presentation of sleep staging, and introduce a few alternatives to the PSG; we then introduce Somno-Art Software, our data and the observed adversary effects on the classification task; we finally give the scope of the present doctoral project. Chapter (2) gives a shallow, taxonomic introduction to transfer learning, and clearly identifies which kind of transfer is adequate for our problem. We further introduce basic concepts around kernels. Thereafter, Chapter (3) presents our contributions around kernels, through TL and TSS, both illustrated on toy examples. We then analyse our data in Chapter (4), and further develop on our methods through experimental results on our application, in Chapter (5) on a few examples and in Chapter (6) on all target recordings. We finally give our conclusions and the perspectives of our research.

Chapter 1

Sleep Staging

Contents

1.1 Gold standard

The reference of sleep staging is the PolySomnoGraphy (PSG) [1], the simultaneous acquisition and study of multiple physiological signals. Nowadays, a PSG is usually constituted of multiple ElectroEncephaloGram (EEG) derivations, of two ElectroOculoGram (EOG - above the eyes) derivations, a chin ElectroMyoGram (EMG) and an ElectroCardioGram (ECG); additional signals may also be recorded, such as nasal or oral airflows, legs EMG, pulse oximetry, which may be useful in the case of certain sleep pathologies.
Once those signals have been recorded, an expert scorer will consider segments of 30s seconds of recording, and associate each epoch to a sleep stage following the American Academy of Sleep Medicine (AASM) guidelines [2]. Currently, distinction is made between five states: wakefulness (W), N1 sleep stage (a transitory state), N2 sleep stage (shallow sleep), N3 sleep stage (slow wave sleep, a deep sleep stage) and Rapid-Eye-Movement (REM) sleep stage (another deep sleep stage, also referred to as paradoxical sleep, as it presents characteristics of wakefulness along with ones of deep sleep).
These stages are in practice characterised by the presence or absence of certain patterns in the EEGs (i.e. spindles, delta waves), of eyes movements on the EOG and on the presence or absence of muscle tone on the EMG. AASM guidelines are regularly updated to improve inter-scorer reliability. The assessment of sleep stages is mainly done through the monitoring of the central nervous system (CNS) activation, through the EEG derivations.
Sleep analysis results in a hypnogram, the temporal representation of the sleep stages through the night, as illustrated by Figure (1.1). From this hypnogram are extracted several sleep parameters which are the main interest in the assessment of sleep pathologies and drug effects. Some sleep parameters of interest are listed in Table (1.1); *lights off* (LOFF) denotes the moment when the lights are turned off − the moment the recorded person decides to try to sleep and from which are measured many latencies of interest. Regarding the sleep architecture, it is usually

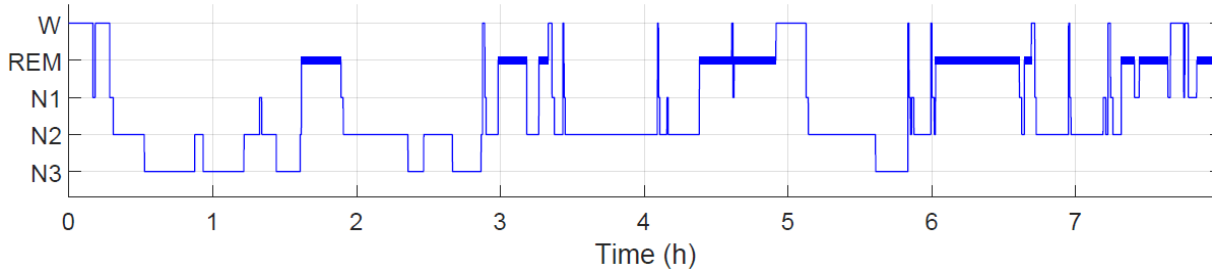


Figure 1.1: Example of a hypnogram, temporal representation of the sleep stages through the night. *x*-axis: time, in hours; *y*-axis: sleep stage.

Figure 1.2: Polysomnography (left) versus Somno-Art Device (right).

composed of multiple sleep cycles, each marked by the end of each REM period, for a duration of between 90 and 120 minutes each.

1.2 Alternatives

Recording PSG signals is unpractical: the subject is covered with numerous electrodes on the scalp, temples and chin. In most cases, the PSG is done in a medical environment (sleep centres, hospital), with low comfort. In order to improve the subject's comfort, some solutions propose a reduced montage, recording fewer signals; some go further and record other, more easily assessable signals such as the heart rate or pulse rate. Most of these simplified solutions are also designed to be user-friendly and to enable recording from home. Improving the user's comfort has also a second advantage: most first nights of recording cannot be used as the subject is stressed and does not sleep in their usual conditions, which will affect their sleep parameters; these are generally denoted as *habituation*

Sleep parameter	Description
Sleep latency	Elapsed time between LOFF and the first sleep stage (N1 or N2).
REM latency	Elapsed time between the first sleep stage (N1 or N2) and the first REM sleep stage.
Persistent latencies	Latencies between LOFF and the first uninterrupted period of 5 minutes of given sleep stage.
Sleep efficiency	Percentage of time spent in sleep stages, i.e. not spent in wakefulness.
Stages percentages	Percentages of time spent in each of the sleep stages.
Awakenings after sleep onset	Number of awakenings of at least 1 minute duration after sleep onset.
Stage shifts per hour	Number of stage transitions per hour.

Table 1.1: (Non-exhaustive) List of sleep parameters of interest.

nights [3]. Allowing better comfort and to sleep from home may reduce such effects.
Sleep staging is both time-consuming and expensive: it requires up to multiple hours of manual scoring by a sleep expert or trained physician (worse with pathological subjects as the sleep structure is disturbed). Such application is an obvious candidate for automatisation through machine learning, which not only drastically reduces the analysis time (to a few minutes, at most), but also the cost. The coherence between scoring would be improved, as there would only be one automated scorer, and so would the intra-scorer reliability.
Most alternatives do both the automatisation of scoring and the improvement of the user's comfort. In the following two subsections, we will distinguish solutions close to the PSG (based on the CNS assessment) and solutions based on the heart-rate or pulse-rate, which monitors in fact the activation of the Autonomic Nervous System (ANS). We give a simplistic overview of the main sleep staging methods in Table (1.2), along with their discriminative capabilities.

1.2.1 CNS-based

Most CNS-based alternatives to the PSG rely on a reduced, facilitated montage [4, 5], which yet still contains most of the useful information required to do sleep staging. Other alternatives simply rely on the full traditional PSG, and only the scoring technique changes.
For instance, the EOG is required for the detections of ocular movements, which are necessary for the scoring of the Rapid-Eye-Movements sleep stage. Some reduced montages are capable, to a certain extent, to estimate the ocular movements from the EEG and thus to score the REM stage.
Regarding the automation of these methods, they are done through expert-based extraction of discriminative features (signal processing) and/or they rely on deep learning and neural networks, among others.

1.2.2 ANS-based

There is some degree of correlation between the activity of the ANS and part of the sleep stages. The ANS activity can be estimated by monitoring the heart-rate (HR) or the pulse-rate (PR); the spectral domain of cardiac activity (heart or pulse rate) contains discriminative information, among others. HR is computed from R-R intervals (RRI), where R is the peak part of the QRS complex (see upper part of Figure 1.3), and is easily determined using signal processing techniques; equivalently, PR is computed from P-P intervals (PPI), where P denotes the peak of the pulse waveform.
The ANS is composed of two sub-systems:

- The sympathetic nervous system, which positively stimulates the cardiac activity by increasing the heart rate; for instance, it is responsible for cardiac arousals.
- The parasympathetic nervous system, which negatively stimulates the cardiac activity by decreasing the heart rate.

It is the sympathovagal balance, the equilibrium between these two sub-systems activities, which helps discriminating sleep stages; this balance can be estimated through the spectral analysis of the heart rate [6]. It has been established that:

- The REM sleep stage is generally similar to wakefulness when it comes to the sympathovagal balance. Movements are useful in discriminating between these two states.
- During NREM sleep stages, the parasympathetic activity is generally dominant over the sympathetic one.
- There may be abrupt sympathovagal shifts at sleep onset (the first passage from wakefulness to sleep) and at certain transitions between NREM sleep to REM or wakefulness.

Between HR and PR

Pulse-rate based solutions rely on the same principle as heart-rate based solution (the estimation of the ANS activity) and further alleviate the heavy requirements of sleep staging, as the pulse-rate acquisition can be done with simple solutions such as, for instance:

Sleep staging method	Input signals	Discrimination capability
Full PSG	EEGs, EOG, EMG, ECG	Complete sleep staging.
Reduced PSG	EEGs	Complete sleep staging, lower accuracy expected.
Somno-Art	PR and Movements	Good accuracy in W/REM/NREM classification.
Actigraphy	Movements	Sleep/wake detection.

Table 1.2: Sleep staging methods and a qualitative description of their sleep staging capabilities.

- Photoplethysmography (PPG), optical measurement of volumetric variations due to blood flow. PPG-based solutions often rely on watches [7], armbands, rings [8], and so on and can easily be combined with pulse oximetry, which may be useful in the assessment of sleep apnoea.
- Ballistocardiography (BCG) [9], which takes advantage of the slight body recoil caused by heart beats.

However, there is a latency between a QRS complex measured at the heart, and the resulting pulse measured at the arm/wrist/finger: the pulse-transit time (PTT). PTT is subjected to variations due to blood pressure and other factors; as a consequence, there might be differences in HR-based and PR-based classification techniques. We give an illustration of an RRI, corresponding PPI and PTT in Figure (1.3).

1.2.3 Actigraphy

The absence of movements is a relatively good indicator of sleep, thus accelerometric recordings are sufficient for basic sleep staging: wakefulness versus sleep (all stages confounded) assessment [10].
Accelerometric information is often combined to the one furnished by the ANS assessment for improved sleep stage classification [11].

1.2.4 Somno-Art Solution

The Somno-Art Solution is the medical device developed by PPRS [12]. It relies on an armband (Somno-Art Device), which records the pulse-rate using PPG and the movements using an accelerometer, and on the exploitation of these measurements (Somno-Art Software).
The Somno-Art Software is based on machine learning for some of the stages determination, and on an expert-based rule system which orchestrates transitions between sleep stages using the output of base detectors and physiological events. Further description will be given in Section (1.3).

1.3 Somno-Art Data

This section presents the data with which we have been working in this doctoral project. We present the Somno-Art Solution process in Figure (1.4): raw data is acquired from the Somno-Art Device and pre-processed, in order to correct for outliers and abnormalities in the signal. There is also a quality review process at (1) (checking, for instance, correct synchronisation between accelerometry and PPG, that the signal is complete and that there is no serious abnormality) and only recordings of sufficient quality constitute the database presented in Subsection (1.3.2).
Thereafter, signals are processed and a first detection of some of the stages is given through machine learning (mostly support vector machines - SVMs); in parallel, physiological events such as cardiac arousals and movements are detected and confirmed.
The originality of the Somno-Art Software lies in the usage of a rule-based system, established through the expertise of sleep scientists.
As we work with a development database, recordings contain ground-truth hypnograms and associated parameters, which will be used to assess our system performances. In the doctoral project, **we focus on the stage pre-detectors**

(2), and particularly on the REM detector. We will develop general methods that might be applied to the other sleep stage detectors as well.

1.3.1 Signal processing: REM detector

The HR is extracted from the ECG RR intervals, and after pre-processing steps aiming to correct for potential outliers, we extract five of the most efficient REM-discriminative features from the literature [13, 14, 15]. In this subsection, we will give a shallow presentation of several key REM-discriminative features.
In Figure (1.5) is represented the Time-Frequency Analysis of the heart-rate for a given recording, computed using an auto-regressive model; the upper black line plot is the tracking of the Respiratory Sinus Arrhythmia (RSA) − influence of the respiratory cycle on the heart rate −, and the lower black line plot is the corresponding reference hypnogram (as a reminder, from top to bottom are wakefulness, and REM, N1, N2 and N3 sleep stages).
In Figure (1.6) are, for the same night, several REM-discriminating features; as can be seen on (1.5), red areas correspond to the REM sleep stage. These features are, from top to bottom:

- The two upper plots are extracted from the RSA:

  (1) "Emergence" of the RSA: power at the tracked frequency. Lower in REM (and W), as the RSA is more dispersed in the frequency band.

  (2) "Dispersion" of the RSA: width of the frequency band for which the power is above a threshold. Usually higher in REM.

- The two next plots are extracted from the Time-Frequency Analysis (TFA):

  (3) Ratio of high frequencies power to low frequencies power. Lower in REM sleep.

  (4) Average power in the low frequencies. It is very sensitive to cardiac arousals, which are usually associated with brutal transitions from sleep to wakefulness, and is lower in REM.

- The two final plots are directly computed from the HR (lowest plot). Both are higher in REM and W than in NREM sleep:

  (5) Power in the very-low frequency (<0.04Hz) band.

  (6) Local increase of the HR signal.

1.3.2 Database composition

The database in its latest state is divided in three subsets of a little more than fifty recordings each:

- LS, **learning** dataset: composed of 59 nights coming from 36 different subjects, used for the detectors training.
- VS, **validation** dataset: composed of 60 nights coming from 32 different subjects, used to tune the detectors and whole system parameters.
- TS, **testing** dataset: composed of 56 nights coming from 30 different subjects, used to assess the detectors robustness and performances.

One subject's recordings all belong to the same subset, and to one subset only, to ensure the datasets statistical independence. The subjects physiology is diverse: young and older, healthy and pathological, male and female, sportive and sedentary; as the recordings mainly come from drug studies, there are usually up to five recordings for a given subject, and diverse as well: screenings, nights with placebo, nights with drugs. In other words, there is a high **inter-individual variability** (differences between subjects) as well as a high **intra-individual variability** (differences between recordings from a given subject).
We point out that a great part of our recordings are from pathological subjects (mostly depressives and/or with respiratory problems), with and without medications. This is believed to be the main cause of the adversary data shifts, which will be presented in the upcoming subsection. Certain pathologies also disturb the sleep architecture, which may result in a loss of cyclicity. A shallow description of the database composition is given in Table (1.3).

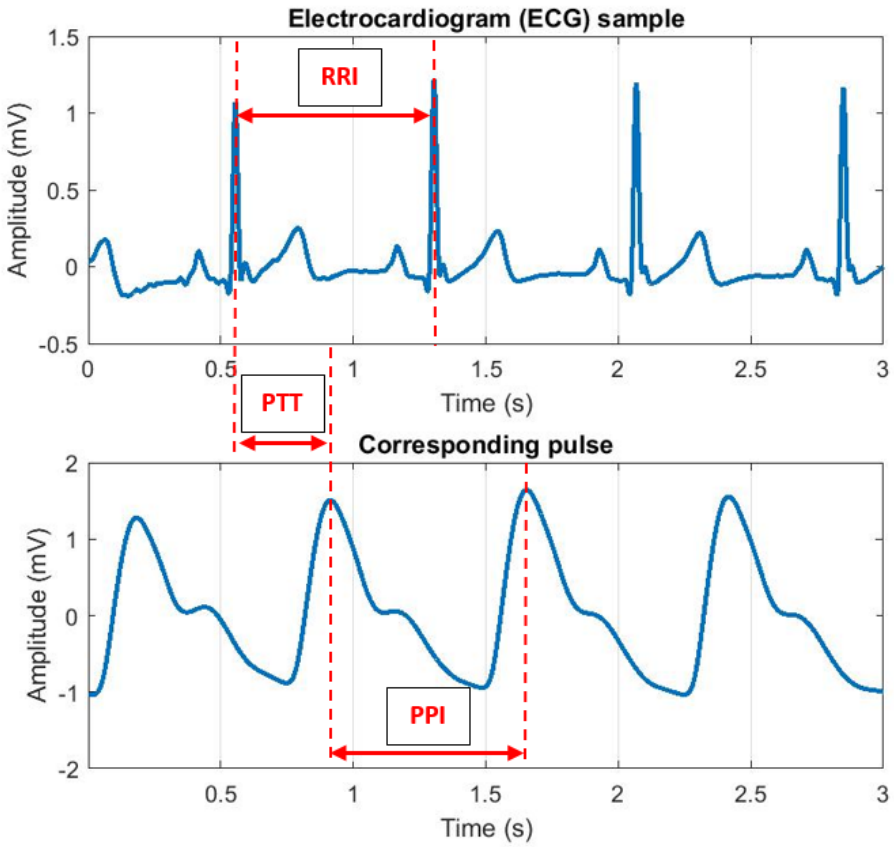


Figure 1.3: ECG and PPG: correspondence between heart-rate and pulse-rate. Further explanations given in Subsection 1.2.2.

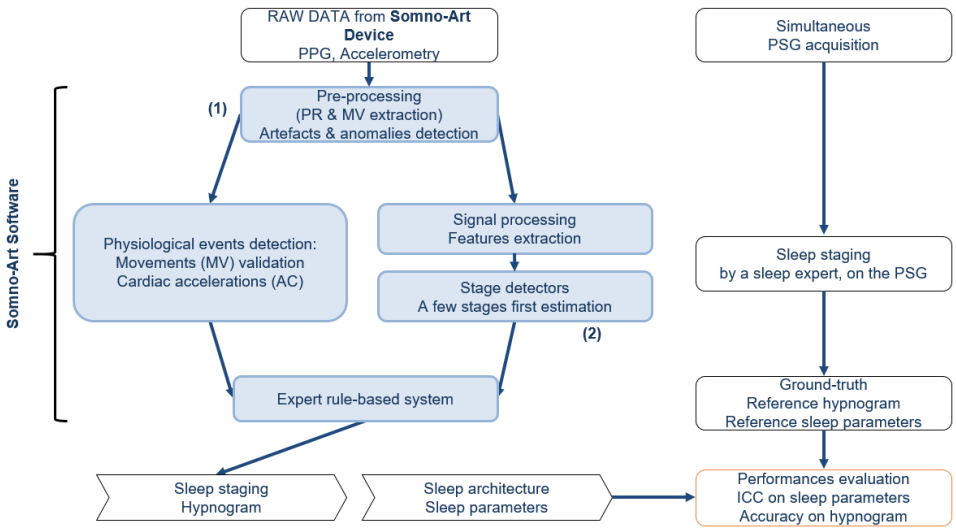


Figure 1.4: Somno-Art Solution, from data acquisition to classification performances estimation. Further explanations given in Section (1.3)

.

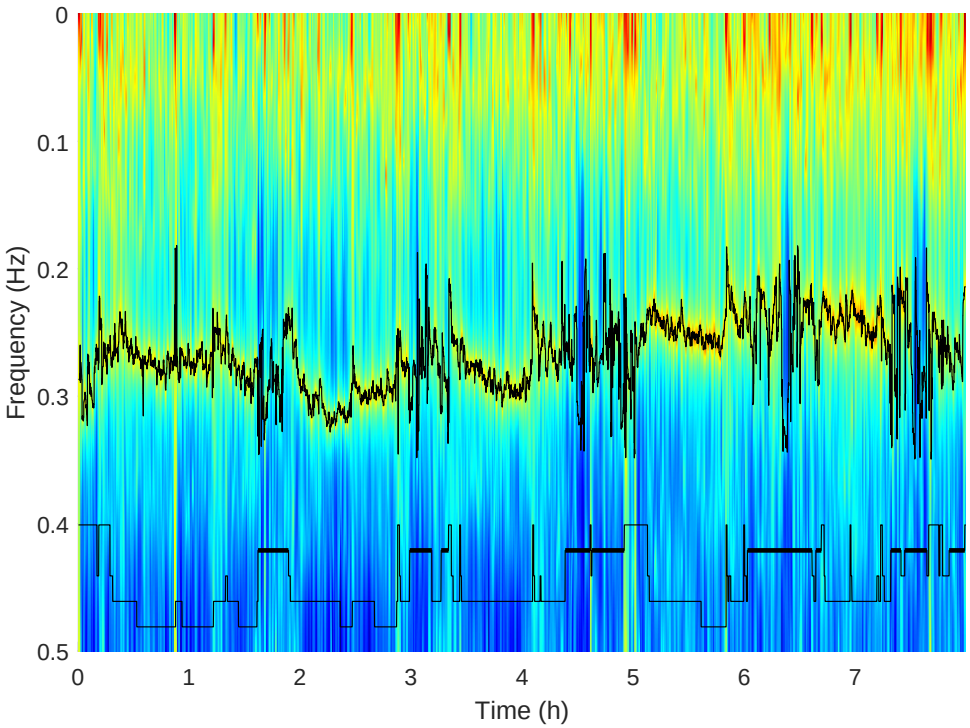


Figure 1.5: Illustration of a Time-Frequency Analysis. RSA tracking (upper black plot) and reference hypnogram (lower black plot); further explanations given in Section (1.3.1). *x*-axis: time, in hours; *y*-axis: frequency, in hertz.

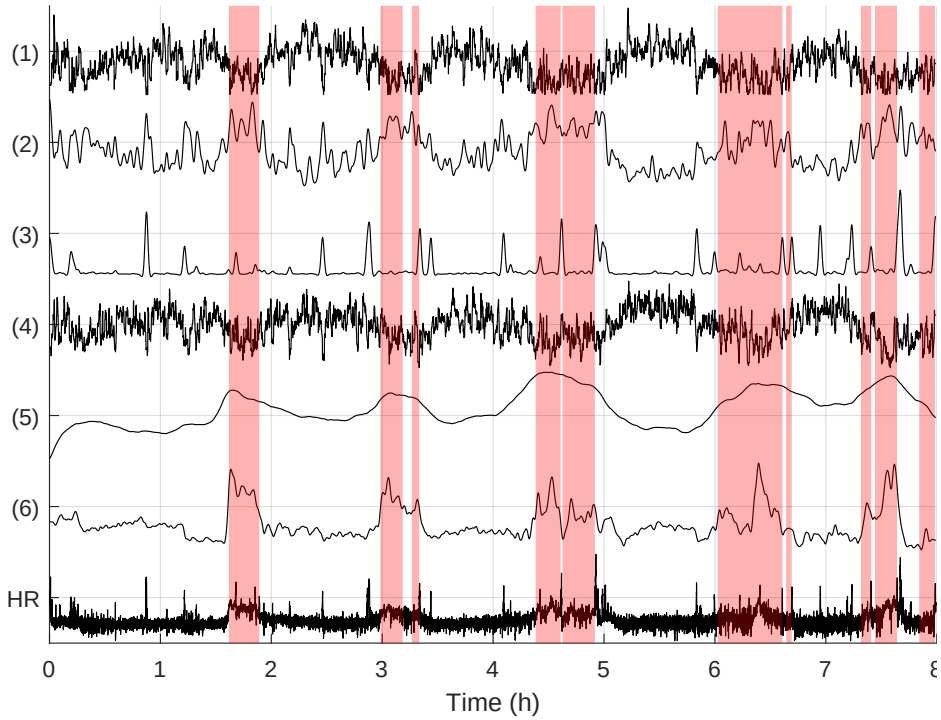


Figure 1.6: Six REM discriminative features, along with the input heart-rate signal (bottom plot), for the same recording as Figure (1.5). Red areas correspond to the REM periods to be discriminated from the rest. Further description in Section (1.3.1). *x*-axis: time, in hours.

Dataset	< 28 yo	$\in [28; 40[$ yo	$\in [40; 54[$ yo	≥ 54 yo
LS	9 (15)	10 (17)	8 (12)	9 (15)
VS	9 (17)	5 (12)	9 (14)	9 (17)
TS	7 (13)	9 (16)	6 (13)	8 (14)

Dataset	Healthy	Depressive	Resp. trouble	Male	Female
LS	8 (10)	19 (40)	9 (9)	19 (28)	17 (31)
VS	7 (14)	17 (38)	8 (8)	10 (21)	22 (39)
TS	6 (9)	16 (36)	8 (11)	12 (20)	18 (36)

Table 1.3: Datasets composition in terms of age, sex and pathology: number of subjects (number of recordings).

1.3.3 Data Shifts

In this manuscript, we will denote by X observations in their matrix form and Y the corresponding labels:

$$X = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix}; Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$$

where $\mathbf{x}_i$ denotes the i^{th} observation, x_{ij} the j^{st} feature of the i^{th} observation and N and d are the number of observations and of features, respectively. We will also denote by X' and Y' the transpose of matrix X and vector Y, respectively.
We have observed and measured, given that (X_S, Y_S) is a source data observations and ground-truth labels, and equivalently (X_T, Y_T) target observations and ground-truth labels:

- **Prior shift**, a difference in the classes prior probabilities: $p(Y_S) \neq p(Y_T)$.
- **Covariate shift**, a difference in the data distributions: $p(X_S) \neq p(X_T)$.
- **Concept shift**, a difference in the label distributions conditionally to the observations $p(Y_S|X_S) \neq p(Y_T|X_T)$.

Prior shift is immediate and has simply been observed by comparing multiple nights hypnograms; for instance, REM sleep stage prior probability is usually around 20% of a night's duration, but has been observed ranging from 0 to 40%.
Covariate shift has been measured using the Maximum Mean Discrepancy (MMD) metric, whose evaluation can easily be kernelized [16, 17], and by comparing multiple recordings data distributions.
Concept shift has been observed by comparing the data and labels of multiple nights, where for similar data distributions, the position of REM labels (and other stages) changes. It may also be marginally due to intra-scorer variability: two scorers may find different labels for a given epoch.
Further illustration of data shift is given by comparing Figures (1.7 and 1.8), where we use the three most discriminative features we have to show a situation of clear covariate and concept shift; such shifts have been observed repetitively on multiple couples of nights. Figure (1.9) illustrates data shift inside a given recording between different REM cycles, as an argument in favour of the intra-individual variability hypothesis.

1.3.4 Performances assessment

For our REM detector evaluation, we will be considering the hypnograms page-to-page agreement (accuracy), and metrics computed from the Receiver Operating Characteristics (ROC) curve: area under the curve, detection rate (DR) for a fixed false-alarm rate (FAR). The operating point is selected in order to get the best performances of the global sleep staging system.
For development purposes, we will focus on maximising the REM detector aforementioned detection rate for the given FAR.

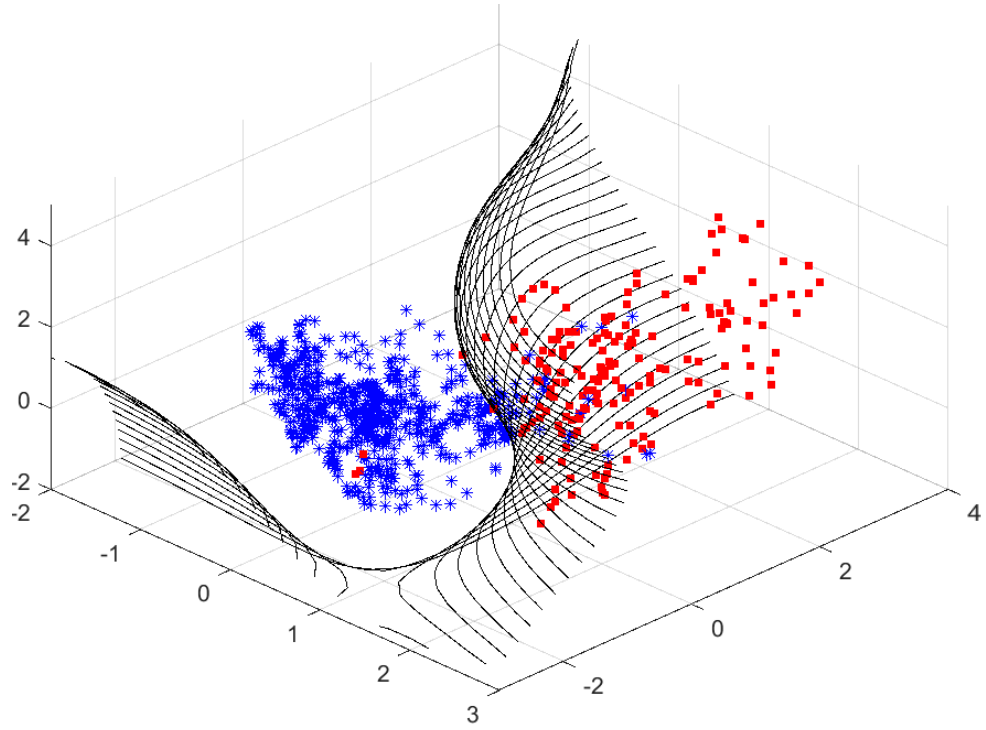


Figure 1.7: Training on one sleep recording: blue asterisks represent N2 and N3 sleep stages, red squares REM stage. Black lines represent the decision boundary of a SVM trained on this data.

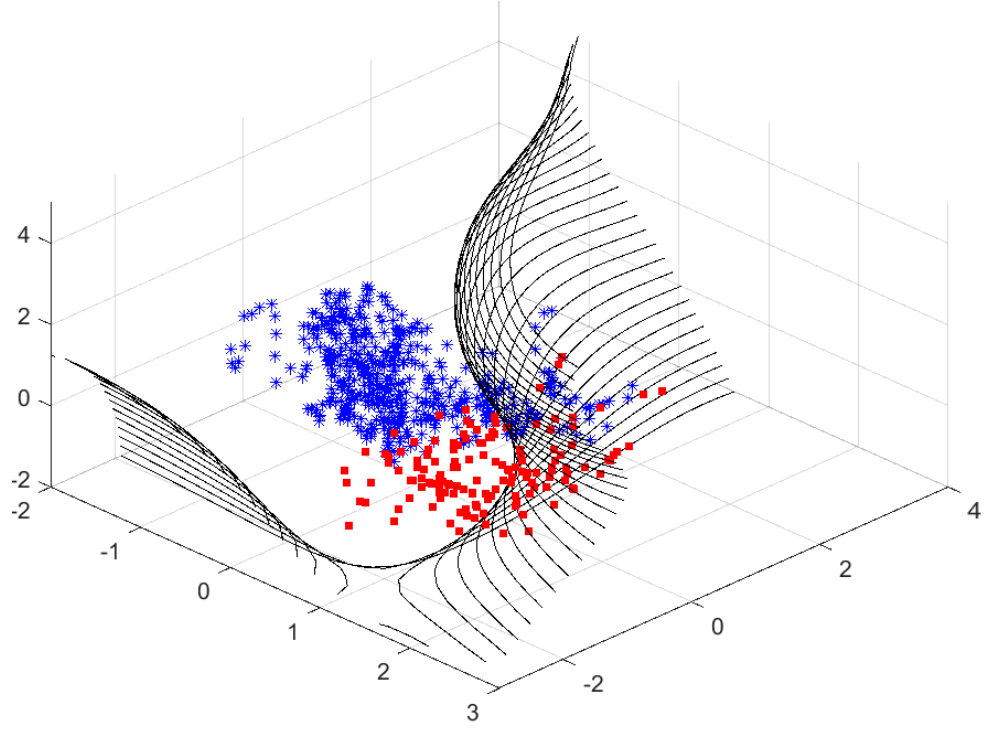


Figure 1.8: Evidence of data shift; on this other recording, blue asterisks represent N2 and N3 sleep stages, red squares REM stage. Decision boundary of Figure (1.7) would misclassify a high number of this other recording observations.

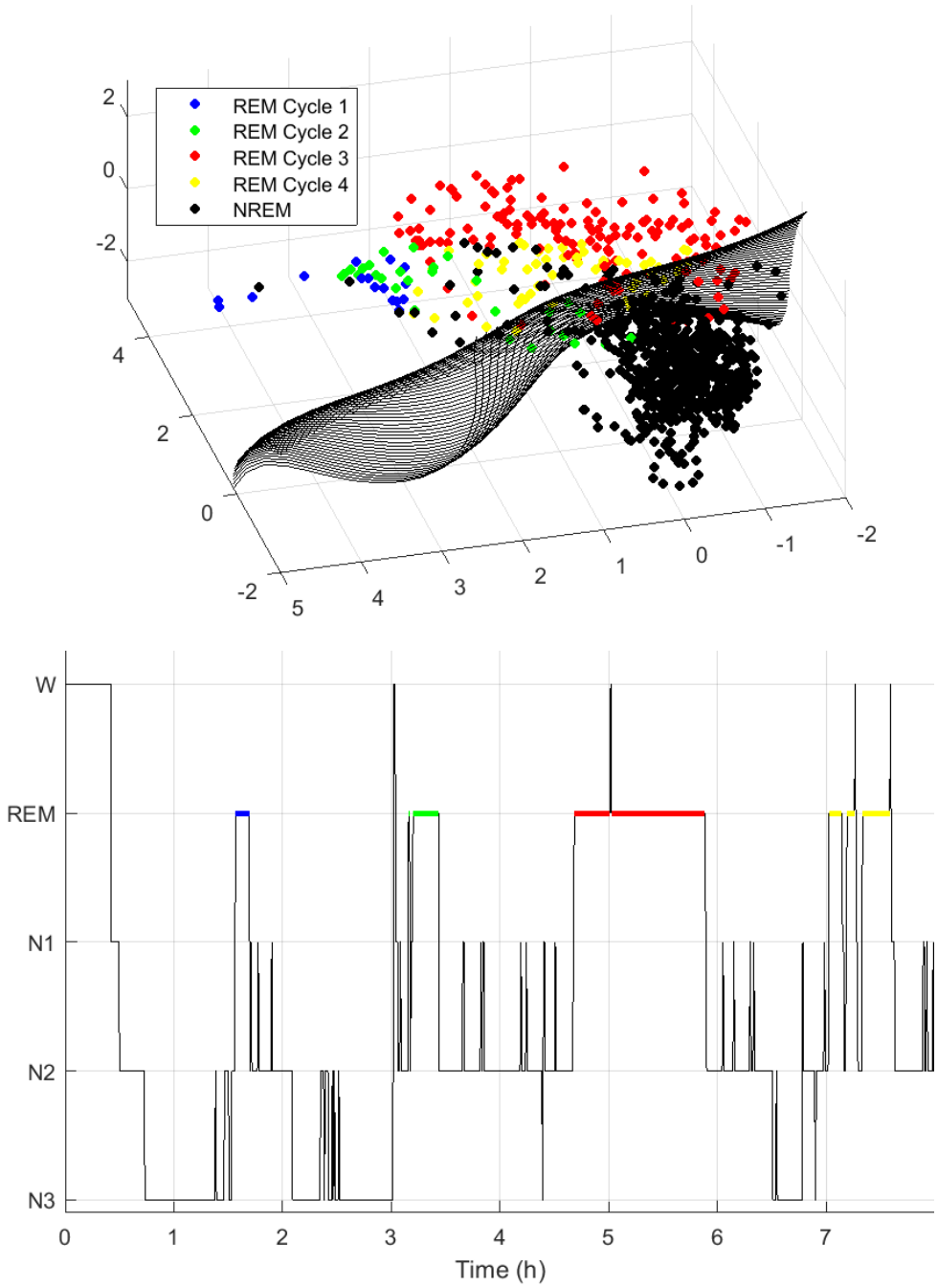


Figure 1.9: Evidence of data shift throughout a given night: different REM cycles are represented using different colours (first one in blue, second in green, third in red and fourth in yellow, as can be seen on the lower panel).

1.4 Thesis scope

The observed data shifts (prior, covariate, concept) are believed to be due to the aforementioned inter-and-intra variabilities. Such data shifts have clear, adversary effects on the classification task, and therefore result in lower performances on recordings too different from the ones used for training. The objective of this thesis is to propose methods and solutions to palliate these adversary effects.
We also indicate that there is a rather high inter-scorer variability in the establishment of the ground-truth; for instance, a scoring agreement of about 83% was estimated in [18]. This contributes to concept shift.
Approaches for the treatment of these adversary effects initially included:

- Adaptation of the system hyperparameters to meta-parameters (e.g. age, sex, body-mass index) and measured parameters (e.g. mean heart-rate, number of cardiac arousals, presence or absence of a high RSA).
- Research of groups of recordings sharing similar properties according to well-chosen metrics (e.g. MMD, correlation measures).

In this doctoral project, we focus on the establishment of general methods to treat our data shift problem, so that they may hopefully be useful for other similar problems. These solutions are mostly based on kernel methods and fall into two related categories:

- Homogeneous transductive transfer learning, mostly using kernel-based methods.
- Performances pre-estimation in order to enable source-selection, method-selection, and threshold selection (in a similar way as training set selection).

Such methods identification will be developed in the next Chapter, introducing key elements of Transfer Learning, identifying adequate solutions for our data shift problems and giving a rapid introduction to kernel methods, on which most of our research is based.

Chapter 2

Methodological background

Contents

2.1 Generalities on Transfer learning

Machine learning techniques usually rely on the implicit assumption that training and testing data come from the same underlying distribution, for a single given task. When either of these assumptions is unverified, it usually results in bad performances on the testing/target data. Transfer learning [19, 20, 21] is the field of machine learning tackling this type of situations: achieving good performances even if training and testing data come from different sources, and/or come from the same distribution but for different tasks.

The goal of this section is to give a shallow introduction to transfer learning; we first introduce core concepts, definitions, and present a few situations where transfer learning is adequate. Thereafter, in Section (2.2), the main ways of transferring knowledge are presented, as well as related fields of machine learning. We conclude on our need for transfer learning.

In the final section of this chapter, we give a quick introduction to kernel methods, as their use in transfer learning has been our focus in this project; in the next chapter, we will present our contributions to kernel-alignment-based transfer learning. We will also introduce core concepts around training set selection.

2.1.1 Definitions

Here are essential transfer learning concepts; we use the exact same notations and definitions as presented in [19], for their clarity.

- **Domain:** A domain $\mathcal{D}$ is composed of a set of observations $X = \mathbf{x}_{i,i\in\{1,...,N\}} \in \mathcal{X}$, $\mathcal{X}$ being the feature space, and of their marginal probability distributions $P(X)$. We abbreviate $\mathcal{D} = \{\mathcal{X}, P(X)\}$.

- **Task:** A task $\mathcal{T}$ is associated to a domain $\mathcal{D}$, and is composed of a set of labels $y_{i,i\in\{1,...,N\}} \in \mathcal{Y}$, $\mathcal{Y}$ being the label space, and their associated predictive function $f : \mathcal{X} \to \mathcal{Y}$, where most of the time, probabilistically speaking, $f(\mathbf{x}_i) = p(y_i|\mathbf{x}_i)$. We abbreviate $\mathcal{T} = (\mathcal{Y}, f(.))$

- **Source:** A source S is composed of a set of observations and their associated labels. We denote by $\mathcal{D}_S = \{(\mathbf{x}_i, y_i)\}_{i\in\{1,...,N_S\}} \in \mathcal{X}_S \times \mathcal{Y}_S$ the *source domain data*. In most cases, source labels are known and N_S, the number of available observations, is generally large.

- **Target:** A target T is composed of a set of observations, whose associated labels are usually unknown. We denote by $\mathcal{D}_T = \{(\mathbf{x}_i, y_i)\}_{i\in\{1,...,N_T\}} \in \mathcal{X}_T \times \mathcal{Y}_T$ the *target domain data*. It corresponds to the testing set on which we want to achieve good classification performances.

Note that, using these definitions, in a "traditional" machine learning set-up, source and target data come from the same domain, $\mathcal{D}_S = \mathcal{D}_T$, and for a unique task, $\mathcal{T}_S = \mathcal{T}_T$. In other words, in a "traditional" machine learning set-up, source and target domains and tasks are the same.

Definition 1 (Transfer Learning) *Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, **Transfer Learning** aims to help improve the learning of the target predictive function $f_T(.)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$, and/or $\mathcal{T}_S \neq \mathcal{T}_T$*

In many cases, the need for transfer learning is due to the target labels being difficult − sometimes impossible − or expensive to acquire. The situations where transfer learning is adequate are fairly diverse, including, e.g. text/image classification/clustering − as words and images are context-sensitive, domains and tasks may differ between applications −, sentiment classification for similar reasons, sensors shifts prediction − as the domain may change over time −, and so on. Physiological data in particular is highly propitious to transfer learning; this is the case of our application, as it is reminded in Subsection (2.3).

2.1.2 Sub-categories of transfer learning

We distinguish different subcategories of transfer learning, depending on the relation between source(s) and target:

Definition 2 (Transductive Transfer Learning) *Situation of transfer learning where source and target have a different-but-related domain, $\mathcal{D}_S \neq \mathcal{D}_T$, while the task is unique, $\mathcal{T}_S = \mathcal{T}_T$. In that case, either source and target are using different feature space: $\mathcal{X}_S \neq \mathcal{X}_T$, or the feature space is the same, $\mathcal{X}_S = \mathcal{X}_T$, but the marginal probability distributions differ: $P(X_S) \neq P(X_T)$*

Definition 3 (Inductive Transfer Learning) *Situation of transfer learning where source and target share the same domain, $\mathcal{D}_S = \mathcal{D}_T$, but their tasks are different: $\mathcal{T}_S \neq \mathcal{T}_T$. In that case, either source and target are using different label spaces, $\mathcal{Y}_S \neq \mathcal{Y}_T$, or the label space is the same, $\mathcal{Y}_S = \mathcal{Y}_T$, but the conditional probability distributions linking the observations to their associated labels differ: $P(Y_S|X_S) \neq P(Y_T|X_T)$.*

Definition 4 (Unsupervised Transfer Learning) *Situation of transfer learning where neither the domain nor the task are the same: $\mathcal{D}_S \neq \mathcal{D}_T$, $\mathcal{T}_S \neq \mathcal{T}_T$.*

It is also common to describe transfer learning as **homogeneous**, when source and target domains are based on the same feature space (or very similar ones), and in contrast, as **heterogeneous** when $\mathcal{X}_S \neq \mathcal{X}_T$.

2.1.3 The transfer of knowledge

There are numerous ways to transfer knowledge; as described in the preceding subsection, it depends on the relations linking $\mathcal{D}_S$ to $\mathcal{D}_T$, $\mathcal{T}_S$ to $\mathcal{T}_T$, on the availability of labels in S and T and on the specificities of what we want to classify.
There are four main approaches to transfer learning, which will be developed in Section (2.2):

- **feature-representation** transfer, where we seek a feature-representation in which $\mathcal{D}_S$ and $\mathcal{D}_T$ are closer.
- **instance** transfer, which principle is to reweigh labelled data in S to be reused in T.
- **parameters** transfer, where we find common parameters and priors for transfer.
- **relation-knowledge** transfer, consisting in mapping the relations linking S and T.

We define as **domain adaptation** any technique whose principle is to bridge S and T domains; we distinguish **symmetric** domain adaptation, where both source and target domains are transformed into a common domain D_C, from its **asymmetric** counterpart, where most of the time the source domain is transformed to resemble the target one. Domain adaptation purpose is mainly to tackle the three aforementioned adversary effects: prior shift, covariate-shift and concept shift (see Subsection (1.3.3)).

2.1.4 Additional considerations

It is of utmost importance to take into account source(s) and target datasets properties, as a way to choose the appropriate transfer learning technique. Considerations include:

- **The observations and labels availability:** In most set-ups, there are far more available observations in the source than in the target ($N_S \gg N_T$) and target labels are unknown. However, when even a few target labels are available, they might be extremely useful to improve the predictive function optimisation.
- **The links between S and T:** As described just above, there are different categories of transfer learning depending on how S and T relate. Different relations result in different appropriate methods.

Negative transfer happens when the knowledge transferred from the source has a detrimental impact on the target prediction function. This can be assessed by comparing the performances of the prediction function trained by combining information from $\mathcal{D}_S$ and $\mathcal{D}_T$ to the ones obtained using $\mathcal{D}_T$ only. When the latter clearly outperforms the former, negative transfer is most likely happening. Note that assessing performances in $\mathcal{D}_T$ is only possible if target labels are available.
For the sake of disambiguation, in this document, we use the following considerations:

- We describe as **supervised** the methods using source labels (most of the times, by means of learning), and by **unsupervised** the techniques using no labels at all (techniques of clustering and equivalent). We qualify as **semi-supervised** any technique composed of both parts, taking both the information from the source labels and the information contained within target observations.
- We define as **informed** the methods taking advantage of target labels, when available, and **uninformed** its counterpart.

2.2 Approaches to transferring knowledge

In this section, we will further develop on the four ways of transferring knowledge, by presenting the main approaches for each type of transfer and giving key references for such techniques.

2.2.1 Feature-representation transfer

Most feature-representation techniques rely on the assumption that source and target domains share a common underlying distribution; the source (resp. target) domain is finally composed of this common distribution and of a source-specific (resp. target-specific) distribution. Most techniques try to use this common distribution as a way to "bridge" source and target. In this section, we will present several key concepts around feature-representation transfer. There seems to be two main approaches:

- To transform both the source and target domain and data distribution to a common domain/distribution; some authors describe this approach as **symmetric**.
- To transform the source domain to try and match the target domain; this may be denoted as **asymmetric** by opposition to the previous approach.

In order to assess the "distance" between source and target distributions, a few metrics are usually considered, such as the Maximum Mean Discrepancy (MMD) [16], Kullback-Leibler divergences [22], Bregman divergence [23], among others. Subcategories of feature-representation transfer learning include:

- Feature **augmentation**: the objective is usually to increase the importance of the common subdomain shared by S and T, by replicating dimensions and using methods or functions that will penalise observations from different domains [24]. Most techniques of feature augmentation are *symmetric*.
- Feature **projection**: in this case, we look for a projection of S and/or T which will minimise one of the aforementioned distance metrics. This is most certainly the most direct approach.
- Feature **mapping**: these usually rely on feature extraction using, for instance, (Kernelized) Principal Component Analysis ((K)PCA), auto-encoders (some authors may specifically refer such techniques as *feature **encoding***), among others.
- Feature **selection**: the key idea is that part of the features have a similar behaviour in both S and T domains, and by reducing both dimensions to these so-called **pivot** features, bridging source and target gets easier.

2.2.2 Instance transfer

The general principle of instance transfer learning is to reuse source observations and associated labels for target estimation. Differences arise depending on whether a few target labels are known or not.
In the case of *informed* transfer learning, solutions take advantage of the known target labels to correct for concept shift, by matching source and target conditional distributions. Solutions usually also consider the classification given by multiple source detectors to establish the target detector, and importance may be given depending on the accuracy of each of the detectors. Mixed models may also dually take into consideration covariate shift as well as concept shift.
The case of *uninformed* transfer learning usually gives more importance to the consensus among multiple source detectors (when available).

2.2.3 Parameters transfer

Parameter-based transfer relies on the assumption that source and target models should share (hyper)parameters; this approach is close to the one of multi-task learning (which we will be quickly presenting in Subsection (2.2.5)). Parameter-based solutions include:

- Parameter **restricting**, which focuses on selecting and using only the parameters that can and should be used in both source and target tasks [25].
- Parameter **sharing**, which directly reuses some of the source parameters in the target (e.g. by freezing some layers in a neural network setup [26], or by identifying common information shared by source and target [27]).

2.2.4 Relation-knowledge Transfer

This approach is very specific to relation-based data, such as social networks and networks in general.
Solutions are based on first-order logic and similar reasoning: if domains are somehow similar, links can be established; for instance, the relation linking a card to a playing deck is similar to the one linking a word to a phrase; such similarities are used extensively.
This type of transfer will not be developed more as it clearly does not relate to our application problematic.

2.2.5 Related work

A few fields of machine learning are close to transfer learning:

- **Semi-supervised** learning: many semi-supervised algorithms may be adequate candidates for transfer learning, as they both take into account what was learned on a source and, independently, the information contained in a target observations distribution.
- **Multi-task** learning: the key difference with transfer learning is that multi-task learning focuses on maximising the classification performances of all tasks equally, while transfer learning is usually all about maximising the target performances.
- **Multi-source** learning: when multiple − different − sources are available, their information may be complementary and should be taken into account for the training of an efficient transfer learning solution.
- **Training set/source selection**: in the case of many shifts between different datasets, and while having multiple labelled datasets (sources) available, being able to select the most adequate sources for a given target may constitute a good substitute for transfer learning.

2.3 Our application

So far, we have introduced the key definitions and elements of transfer learning, and given a shallow description of the main four approaches to transferring knowledge. For our application detectors, we consider as source any of our training recordings, and as target any new recorded night of unknown labels. For the training and testing of our transfer learning methods, we will consider LS as our set of sources and VS as our set of targets (our learning and validation datasets, respectively, as described in Subsection (1.3.2)). Regarding our need for transfer learning, we have determined that:

- The input space is unique $\mathcal{X}_S = \mathcal{X}_T$, although we might consider heterogeneous solutions in the future.
- The domain is different since we have identified data shifts between many sources and targets, as described in Subsection (1.3.3).
- For each of the tasks, the label space is unique $\mathcal{Y}_S = \mathcal{Y}_T$, as we tend to discriminate REM against non-REM, sleep against wakefulness, and so on.
- All target labels will always be unknown; subsequently, the methods we will develop are *uninformed*.

With this setup, and in accordance with the previous definitions, we ought to focus on *homogeneous transductive transfer learning* techniques.
Prior to the doctoral project, multiple feature-based approaches of transfer learning have been tried with little success [28, 29]; the necessity of an efficient algorithm, capable of dealing with large amounts of data, has also been noted. With these considerations, we have focused our research on kernel-based methods, for kernels good properties (kernel trick, representer theorem); in the following section, we give a quick introduction to kernel theory.
As hinted in the previous Section (2.2.5), we will also consider training source selection as complementary strategy to *direct* solutions of transferring knowledge. This will be presented in Section (3.4). Our global strategy in this doctoral project may be summarised in three complementary approaches, for a given new testing, target recording:

- Select among all sources the most adequate(s) by means of training source selection.
- Estimate the need for transfer learning by giving a prediction of the expected performances without transfer.
- If needed, select among our transfer learning techniques the one that seems most adequate.

2.3.1 Relation to domain adaptation

In the literature, most domain adaptation techniques rely on resolution of constrained optimisation problems, and many solutions are iterative and usually do not scale well with great amounts of data [30, 31, 32]. In our case, as the amount of data - for a given target recording - is high and as we require fast computation, we propose objective functions that allow direct solutions.
Most - if not all - of the methods we propose focus on the correction/improvement of the *decision* of a reference detector, by taking into account the unsupervised information contained in a kernel matrix; we take advantage of both the good amount of available source data we have, and their potential similarities with new target recordings. As it will be shown, we adapt domains implicitly using similarity profiles in a Reproducing Kernel Hilbert Space (RKHS), easily computed using the kernel trick.
The methods we will be presenting in Chapter (3) share similarities with the work presented in [24], where feature augmentation empowers the common subdomain shared by source and target; in [33], relying on a invariant latent space by learning the structure of an Hilbert space specifically, allowing to minimise discrepancies between source and target domain and dissimilarities between labelled samples; finally, the work in [34], centred around feature transformation by finding a subspace in which source and target centroids are closer.

2.3.2 Relation to training source selection

Complementarily to our work on *decision* adaptation, we will also propose methods of training *source* selection, so to optimally use our available source recordings, and in some cases, alleviate our need for knowledge transfer.
Traditional *training set selection* [35, 36] is focused on the selection of a subset of training observations which will result in a successful predictor; such subset supposedly has lower redundancy in the selected observations and is free of outliers (which have adversary effects in the training task).
Our objective is in between traditional *training set selection* and *transfer learning*: to select for a given target night one - or multiple - reference source detector(s) which will result in good classification performances - we qualify as *training source selection* such methods.
One essential assumption for such a method to work is that our set of source detectors is large and diverse enough so that any new unlabelled target night can be correctly classified by at least one such source. Previous work sharing such objective include [37].

2.4 Generalities on Kernel methods

In many classification tasks, observations in input space cannot be linearly separated. The main interest of kernels is to allow easy computation in a feature space of higher (sometimes infinite) dimension, where observations become linearly separable with simple linear algorithms; more than that, thanks to the kernel trick, the transformation is implicit and computations are done directly in the input space, allowing an high computational efficiency. This section gives a rapid overview on kernel key definitions and properties.

2.4.1 Reproducing kernel through Mercer's theorem

Let us consider an Hilbert space $\mathcal{H}$ of associated inner product $\langle \cdot,\ \cdot \rangle_{\mathcal{H}}$ and continuous functions ϕ defined on $\mathcal{X}$. There exists an unique function k of $\mathbf{x}$: $k(\mathbf{x}_1,\ \mathbf{x}_2)$ so that [38], $\forall \phi \in \mathcal{H}$:

$$\phi(\mathbf{x}_2) = \langle \phi,\ k(\cdot,\ \mathbf{x}_2) \rangle_{\mathcal{H}} \tag{2.1}$$

k is the so-called reproducing kernel of $\mathcal{H}$, as $\{k(\cdot,\ \mathbf{x}_1), \mathbf{x}_1 \in \mathcal{X}\}$ generates $\mathcal{H}$. One will also notice that, $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$,

$$\langle k(\cdot,\ \mathbf{x}_1),\ k(\cdot,\ \mathbf{x}_2) \rangle_{\mathcal{H}} = k(\mathbf{x}_1,\ \mathbf{x}_2) \tag{2.2}$$

Kernel	Parameter(s)	Expression
Gaussian	σ	$exp(-\frac{\|\|\mathbf{x}_1-\mathbf{x}_2\|\|^2}{2\sigma^2})$
Polynomial	c, d	$(\mathbf{x}_1'\mathbf{x}_2 + c)^d$
Sigmoid	η, ν	$tanh(\eta\mathbf{x}_1'\mathbf{x}_2 + \nu)$

Table 2.1: Example of a few popular kernel functions (for two given observations $\mathbf{x}_1$ and $\mathbf{x}_2$).

k, as an inner product, is symmetric; it also gives, for any couple of elements in $\mathcal{X}$, the image of their inner product in $\mathcal{H}$, directly computed from the input space $\mathcal{X}$; this is a most practical property of reproducing kernels.
Any symmetric function $k(\mathbf{x}_1,\ \mathbf{x}_2) \in \mathcal{L}^2(\mathcal{X}^2)$ can be decomposed as [39]:

$$k(\mathbf{x}_1,\ \mathbf{x}_2) = \sum_i \gamma_i \psi_i(\mathbf{x}_1)\psi_i(\mathbf{x}_2) \tag{2.3}$$

where $\psi_i \in \mathcal{L}^2(\mathcal{X})$ and where γ_i are real valued scalars; these are the eigenfunctions and eigenvalues associated to k, respectively.
In order for $k(\mathbf{x}_1, \mathbf{x}_2)$ to constitute an inner product, it suffices that $\forall i,\ \gamma_i \geq 0$. Mercer's theorem gives an equivalent condition, that is, if and only if $\forall f \in \mathcal{L}^2(\mathcal{X})$:

$$\iint k(\mathbf{x}_1, \mathbf{x}_2) f(\mathbf{x}_1) f(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \geq 0 \tag{2.4}$$

then k is a so-called Mercer kernel.
The work in [40, 41] leads to the following theorem: *any Mercer kernel k can be associated to a real-valued* ***Reproducing Kernel Hilbert Space (RKHS)*** *$\mathcal{H}$. In that case, $k(\boldsymbol{x}_1,\ \boldsymbol{x}_2) = \langle \phi(\boldsymbol{x}_1),\ \phi(\boldsymbol{x}_2)\rangle_{\mathcal{H}}$ where $\phi(\boldsymbol{x}_1) = k(\cdot, \boldsymbol{x}_1)$, $\boldsymbol{x}_1 \in \mathcal{X}$.*

2.4.2 Kernel trick

The **kernel trick** denotes the fact that the mapping functions ϕ are only implicit and no computation is done in the associated features spaces; all computations are done in the input space through the kernel, allowing the use of linear algorithms even when the observations in input space are non-linearly separable.
As example, consider the mapping function $\phi : \mathbb{R}^3 \to \mathbb{R}^6; (x_1, x_2, x_3) \mapsto (x_1^2, x_2^2, x_3^2, \sqrt{2}x_1x_2, \sqrt{2}x_1x_3, \sqrt{2}x_1x_2)$.
Inner products in the transformed feature space can be replaced by the polynomial kernel in Table (2.1), with parameters $c = 0, d = 2$, and computation using the kernel will take half less operations as it would in the feature space.
In order to illustrate the strength of kernel methods, let us consider the classification task illustrated in Figure (2.1): it is impossible to linearly separate the blue class from the red class with the features of the left panel. In the 3D representation in the right panel, adding the third feature $x_3 = \sqrt{x_1^2 + x_2^2}$ allows the two classes to be linearly separated. The bold black line in the left panel has been obtained by training a Support Vector Classifier (SVC), which did not require the explicit transformation of the input feature space, thanks to the kernel trick.

2.4.3 Empirical risk and Representer theorem

For any classification task (or regression problem), we want the solution function f to be the most accurate when it comes to prediction; in order to estimate its performances, with the choice of a loss function $\mathcal{L}(\mathbf{x},\ y,\ f(\mathbf{x}))$ and knowing the joint observations-labels distribution $P(\mathbf{x},\ y)$, the *theoretical* risk [42] expresses as:

$$\begin{aligned} \mathcal{R}_{L,P}(f) &= \int_{X,\ Y} \mathcal{L}(\mathbf{x},\ y,\ f(\mathbf{x}))p(\mathbf{x},\ y)\mathrm{d}y\mathrm{d}\mathbf{x} \\ &= \mathbb{E}[\mathcal{L}(X, Y, f(X)] \end{aligned} \tag{2.5}$$

In practice, we (usually) do not have access to $P(\mathbf{x},\ y)$; instead, we use a training set of observations and associated known labels (X, Y) for which we can estimate the **empirical risk**, that is:

$$\hat{\mathcal{R}}_n(f) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\mathbf{x}_i,\ y_i,\ f(\mathbf{x}_i)) \tag{2.6}$$

In practice, an additional regularisation term $\Omega(f)$ is added to avoid over-fitting; indeed, any function complex enough would be able to achieve perfect classification on the training dataset but would then be inefficient on new testing data. By punishing functions too complex, a good compromise can be found allowing the selection of a model that will be efficient on both the training and testing data and will also avoid overfitting. The regularised empirical risk can then be expressed as:

$$\hat{\mathcal{R}}_n(f) + \lambda\Omega(f) \tag{2.7}$$

where λ is the compromise parameter allowing to modulate the complexity of f. The objective and principle of most classifiers is the search of an optimal function $\hat{f}$ that will effectively minimise the regularised empirical risk, that is:

$$\hat{f} = \underset{f\in\mathcal{F}}{argmin} \quad \hat{\mathcal{R}}_n(f) + \lambda\Omega(f) \tag{2.8}$$

The **representer theorem** [43, 44, 45] states practical expressions to high/infinite dimensional spaces regularisation problems (such as the minimisation of regularised empirical risk, among others) as finite dimensional subspaces obtained by using representers of the data (i.e. known observations). This allows practical and computationally-efficient methods; this is the base, among others, of Support Vector Machines (SVM). The scope of this doctoral project will be limited to regularisation problems in the RKHS, for which solutions will be of the form:

$$f^*(\cdot) = \sum_{i=1}^{N}\alpha_i k(\cdot,\ \mathbf{x}_i) \tag{2.9}$$

This will be notably useful in the case of kernel centring, as it will be introduced in Subsection (2.4.5) and further developed in Section (3.3).

2.4.4 Kernel-Target Alignment

So far, we have presented the main interests of kernels, but did not present practical methods of kernel selection and kernel tuning. This is the principal and traditional use of Kernel-Target Alignment. Alignment $\mathcal{A}$ is a measure of similarity between two matrices M_1 and M_2 and simply expresses as [46]:

$$\mathcal{A}(M_1,\ M_2) = \frac{\langle M_1,\ M_2\rangle_F}{\sqrt{\langle M_1,\ M_1\rangle_F\langle M_2,\ M_2\rangle_F}} \tag{2.10}$$

where $\langle\cdot,\ \cdot\rangle_F$ denotes Frobenius inner product. Alignment has values in $[-1;\ 1]$, 1 being attained when $M_1 = \alpha\ M_2, \forall\alpha > 0$. Polarisation $\mathcal{P}$ is the non-normalised counterpart of the alignment:

$$\mathcal{P}(M_1,\ M_2) = \langle M_1,\ M_2\rangle_F \tag{2.11}$$

For any classification problem, the *ideal* mapping function $\phi(\cdot)$ is the one that directly returns the correct label: $\phi(\mathbf{x}) = y$. In the case of binary classification (our detectors), such *ideal* function is associated to the *ideal* kernel matrix $K^* = YY'$; when using coding ±1, $K^*_{ij} = 1$ if y_i and y_j come from the same class and -1 if they are from different classes; Y is the label vector in column form.
The traditional use of KTA is kernel tuning/engineering, by maximising w.r.t. the kernel k the alignment of its kernel matrix K with the ideal kernel matrix K^* [46]:

$$\begin{aligned}\max_K \mathcal{A}(K,\ K^*) &= \max_K \frac{\langle K,\ K^*\rangle_F}{\sqrt{\langle K,\ K\rangle_F\langle K^*,\ K^*\rangle_F}} \\ &= \max_K \frac{Y'KY}{||K||_F||Y||^2}\end{aligned} \tag{2.12}$$

where $||\cdot||_F$ denotes Frobenius norm and $||\cdot||$ L_2 norm.
KTA applications are numerous [47, 48]; however, and as it will be developed in next chapter, our work has been focused on a different use of kernel-target alignment: in our situations, the kernel is fixed and *observed*, and is used as a way to transfer knowledge (in transfer learning) or to compare multiple training sets and select the one which seems most appropriate (training source selection).
Some applications also use the non-normalised counterpart of KCA, the so-called Kernel-Target Polarisation (KTP):

$$\max_K \mathcal{P}(K,\ K^*) = \max_K Y'KY \tag{2.13}$$

2.4.5 Kernel centring

The choice of the origin of the coordinate system has a dramatic impact on inner products, and therefore, on kernel values and matrices. We first illustrate this idea using a toy example, in Figure (2.2), where we consider two origins of the coordinate system, or *centres*, and see how it will affect the corresponding kernel matrices.
In order to discriminate the red circles from the blue stars (upper left panel), the *ideal* kernel matrix is the one represented in the upper right panel and is computed using the labels. We then compare two *observed* kernel matrices obtained by centring using centres C_1 and C_2, represented in the lower left and lower right panel, respectively. As can be seen, the one obtained with centre C_2 is way more resembling to the *ideal* kernel matrix, and is more class-discriminative. This can be assessed through KTA: there is an alignment of 0.05 between the *ideal* kernel matrix and the kernel matrix using C_1, while the alignment with the kernel matrix centred using C_2 is almost perfect, with a high value of 0.96.
Methods and interests of centring the kernel matrix have been the focus of a fair amount of research [49, 50]. In this subsection, we simply introduce the basics of kernel centring; elements of the centred kernel matrix express as:

$$K_{C_{ij}} = \langle \phi(\mathbf{x}_i) - C,\ \phi(\mathbf{x}_j) - C \rangle_{\mathcal{H}} \tag{2.14}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are two observations and C is the considered centre. We consider two strategies for the choice of the centre C, using the representer theorem or using directly an observation from input space.

In feature space

In this case our centre expresses as:

$$C = \sum_{i=1}^{N} \alpha_i \phi(\mathbf{x}_i)$$

where $\boldsymbol{\alpha}$ is a weighing vector. Once inputted in Equation (2.14):

$$\begin{aligned}
K_{C_{kl}} &= \left\langle \phi(\mathbf{x}_k) - \sum_{i=1}^{N} \alpha_i \phi(\mathbf{x}_i),\ \phi(\mathbf{x}_l) - \sum_{i=1}^{N} \alpha_i \phi(\mathbf{x}_i) \right\rangle_{\mathcal{H}} \\
&= \langle \phi(\mathbf{x}_k),\ \phi(\mathbf{x}_l) \rangle_{\mathcal{H}} - \sum_{i=1}^{N} \alpha_i \langle \phi(\mathbf{x}_i),\ \phi(\mathbf{x}_l) \rangle_{\mathcal{H}} \\
&\quad - \sum_{i=1}^{N} \alpha_i \langle \phi(\mathbf{x}_k),\ \phi(\mathbf{x}_i) \rangle_{\mathcal{H}} + \sum_{i=1}^{N_T} \sum_{j=1}^{N_1} \alpha_i \alpha_j \langle \phi(\mathbf{x}_i),\ \phi(\mathbf{x}_j) \rangle_{\mathcal{H}} \\
&= K_{kl} - \sum_{i=1}^{N} \alpha_i K_{il} - \sum_{i=1}^{N} \alpha_i K_{ki} + \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j K_{ij}
\end{aligned}$$

This can easily be put in matrix form, that is:

$$K_C = K - \Gamma'_{\boldsymbol{\alpha}} K - K\Gamma_{\boldsymbol{\alpha}} + \Gamma'_{\boldsymbol{\alpha}} K \Gamma_{\boldsymbol{\alpha}} \tag{2.15}$$

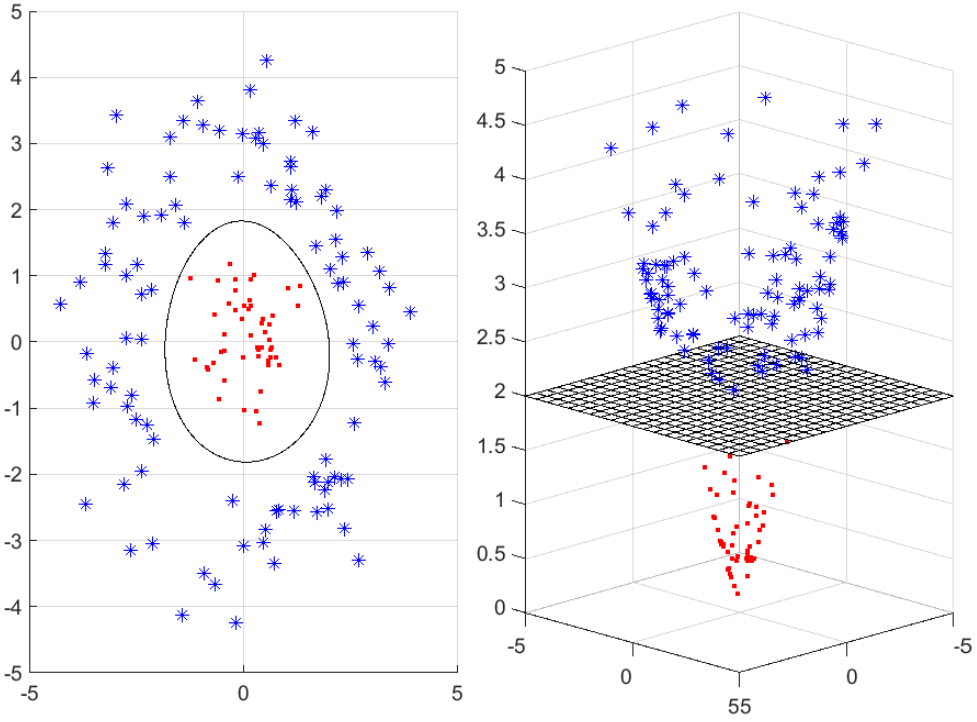


Figure 2.1: Illustration of the kernel trick: in left panel, red (dots) and blue (asteriks) classes are not linearly separable; adding the third feature as proposed in the right panel could solve this problem; kernels allow the computationally-efficient use of linear algorithms through *implicit* transformations, as all computations are done directly in the input space. The decision boundary (black line) was obtained by training a Support Vector Classifier (SVC).

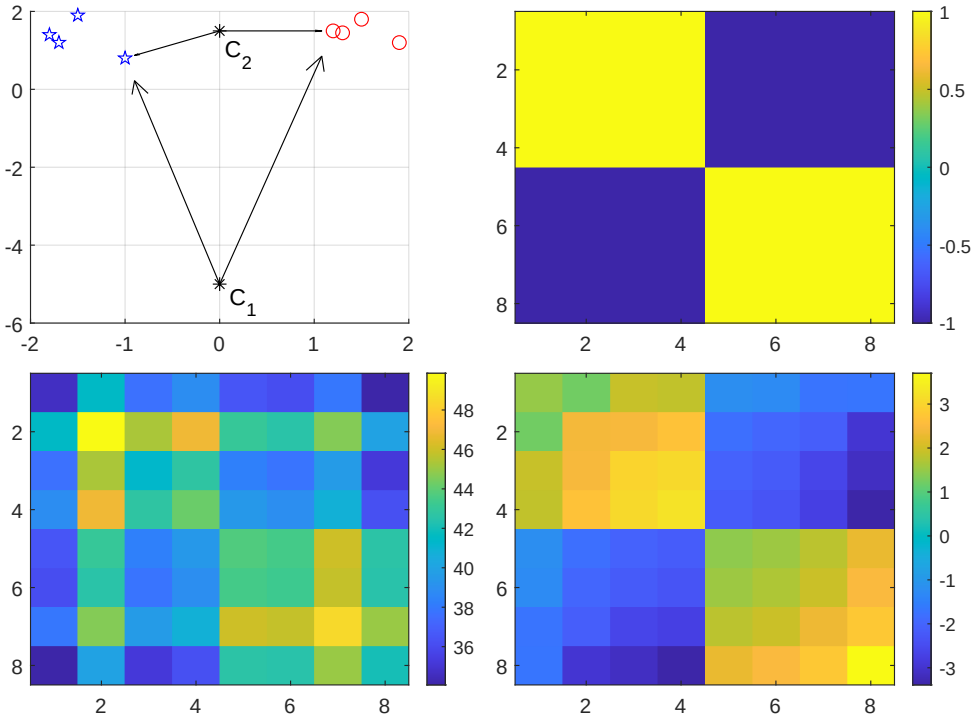


Figure 2.2: Illustration of the importance of choosing the right "centre"/origin of the coordinates system. Upper left: a few observations and two centres C_1 and C_2; upper right: ideal kernel matrix computed using the labels; lower left: obtained kernel matrix by using centre C_1; lower right: obtained kernel matrix by using centre C_2.

where $\Gamma_{\boldsymbol{\alpha}}$ is a matrix of size $N \times N$, composed of the weighing vector $\boldsymbol{\alpha}$ as:

$$\Gamma_{\boldsymbol{\alpha}} = \begin{pmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & \alpha_2 & \cdots & \alpha_2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{N_T} & \alpha_{N_T} & \cdots & \alpha_{N_T} \end{pmatrix}$$

Directly from input space

In that case we express our centre as:

$$C = \phi(\mathbf{x}_C)$$

where $\mathbf{x}_C \in \mathcal{X}$. By development and once put in matrix form, this leads to the following expression for the centred Gram matrix:

$$K_C = K - K_{\star C}\mathcal{J}_{1\times N} - \mathcal{J}_{N\times 1}K_{C\star} + \mathcal{J}_{N\times N} \tag{2.16}$$

where $K_{\star C}$ denotes the kernel vector between input observations and the centre in column form and $K_{C\star}$ in row form. Finally, $\mathcal{J}_{a\times b}$ denotes the all-ones matrix of size $a \times b$. This formulation is different from that of Equation (2.15); here we do not need to use the representer theorem as we can directly explore the input space $\mathcal{X}$.

Optimised centring

The choice of an optimal centre may have a dramatical impact on a classification algorithm performances, and has been the focus of previous researches. In [51], good optimisation criteria are proposed in order to select an efficient centre. In [49], an optimisation algorithm is proposed for the maximisation of KTA w.r.t. the centre.
Most optimisation problems require knowledge of some labels; while in many situations we have access to an estimate of the labels of interest, it may be ill-advised to try and optimise centring based on these estimates. In Section (3.2) we will be introducing an extension of KTA to transfer learning, and in Section (3.3) the optimisation of such extension w.r.t. centring.

Mean centring

In most of our work, and given that there is no *true* knowledge of the labels, we centre using the *mean* centre in feature space, that is with $\alpha_i = \frac{1}{N} \forall i \in \{1, \cdots, N\}$; such centring method is easily done using the centring matrix C_N:

$$C_N = \mathcal{I}_N - \frac{1}{N}\mathcal{J}_N \tag{2.17}$$

where $\mathcal{I}_N$ is the identity matrix and $\mathcal{J}_N$ is the all-ones matrix, both of size $N \times N$. The centred kernel matrix then simply expresses as:

$$K_C = C_N K C_N \tag{2.18}$$

While such a centre may be suboptimal, in most cases it proved to be better than not to centre at all; it also has the advantage of being easily computed.

Summary

In this chapter we have introduced and given essential definitions and concepts around Transfer Learning (TL). We have identified what subcategories of TL might help solve our data shift problem, and also considered related fields of machine learning (training source selection, semi-supervised solutions). In Section (2.4) in particular, we have presented the main elements of kernel theory, which will be particularly useful when presenting our contributions to transfer learning/training source selection in the next chapter.
The next chapter will start by introducing Quadratic Loss Transfer Learning (QLTL), a method of semi-supervised transfer learning based on a different use of KTA; it will then present Kernel Alignment Transfer Learning (Kx-ATL), further methods of transferring knowledge, and Kernel-Cross Alignment (KCA), an extension of KTA to transfer learning problems. Thereafter, a centring algorithm will be presented, as to further improve our KCA-based methods. Finally, our contributions will conclude on all our work around Training Source Selection, which

has been extensively based on our methods of transfer learning (as it uses alignment or measures similar to alignment).

Chapter 3

Our contributions to Transfer Learning

Contents

So far we have presented our objectives and the associated literature solutions that ought to achieve them. In this third section, we propose new − semi-supervised − methods of transfer learning. We consider the following situation:

- One target recording of unknown labels Y_T, which we want to assess. The estimation of the target labels, obtained through our methods, is denoted by $\hat{Y_T}$ throughout the chapter.
- Multiple source recordings of known labels, which are used to train N_S source detectors. The output of source detector S_i on target observations is denoted by $Y_T^{\{S_i\}}$. One source is associated to one recording and the associated detector obtained through training.
- Most of our methods rely on a clearly supervised part, giving a first estimation of the target labels, and an unsupervised (or semi-supervised) one, which will be used as a way to correct our estimate by taking into account the data distribution.

This chapter introduces our methods, Quadratic Loss Transfer Learning (QLTL) in Section (3.1) - a semi-supervised method of transfer learning using multiple sources; this inspired a more direct approach with better results: Kernel Alignment Transfer Learning (KxATL), in Section (3.2) - two simple methods of transfer learning between a source and a target; as we noticed the strong impact of kernel centring on our methods, we proposed a method to optimise kernel-centring in Section (3.3). Finally, as our available set of recordings is big enough and should cover many possible variabilities, we propose a few predictors allowing source selection (choosing the best source to predict labels of a given target) in Section (3.4).

3.1 Quadratic Loss Transfer Learning

In [52] we have introduced Quadratic Loss Transfer Learning (QLTL), an algorithm for multi-source, semi-supervised transfer learning. Our objective is to improve the decision given by a set of source detectors on a given target by doing a compromise with the information contained in a fixed kernel matrix. It is achieved through the resolution of the following minimisation problem:

$$\hat{Y_T} = \underset{Y_T}{argmin} \sum_{i=1}^{N_S} \left(||Y_T - Y_T^{\{S_i\}}||^2 \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \tag{3.1}$$

where $Y_T^{\{S_i\}}$ is the estimation of target labels given by the i^{th} source detector, K_{TT} is the kernel matrix computed between target observations, N_S is the number of source detectors and N is the number of target observations; $\lambda \geq 0$ in an hyper-parameter to balance the relative importance of both parts of the problem, and $\hat{Y_T}$ is our estimate of the sought target labels vector Y_T.
This is semi-supervised, as the minimisation of the quadratic loss is entirely supervised, while the maximisation of kernel-target polarisation ($\langle K_{TT}, Y_T Y_T' \rangle_F = Y_T' K_{TT} Y_T$) is entirely unsupervised, as only the target observations are involved. This is also a multi-source situation, as we take into account the output of N_S individual detectors.
The solution of this optimisation problem is obtained directly by zeroing the gradient of the expression:

$$\begin{aligned} &\nabla_{Y_T} \left(\sum_{i=1}^{N_S} \left(||Y_T - Y_T^{\{S_i\}}||^2 \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \right) \\ &\quad = 2(N_S \mathcal{I}_N - \lambda K_{TT}) Y_T - 2 \left(\sum_{i=1}^{N_S} Y_T^{\{S_i\}} \right) = 0 \\ &\equiv \hat{Y_T} = (N_S \mathcal{I}_N - \lambda K_{TT})^{-1} \left(\sum_{i=1}^{N_S} Y_T^{\{S_i\}} \right) \end{aligned} \tag{3.2}$$

where $\mathcal{I}_N$ denotes the identity matrix of size N and $\nabla_{Y_T} f$ the gradient column vector V of the cost function on Y_T components, of general term $V_i = \frac{\partial f}{\partial Y_{T_i}}$.
The convexity of the problem depends on the positive-definiteness of the matrix $H = N_S \mathcal{I}_N - \lambda K_{TT}$ in the left part of the solution. Let us denote by $(\mu_i)_{i \in \{1,\dots,N\}}$ the eigenvalues of K_{TT} and μ_{max} its greatest eigenvalue; these are positive as K_{TT} is a kernel matrix. H is positive-definite if all its eigenvalues are positive, and one may notice that if $(\mu_i,\ V_i)$ is eigenpair of K_{TT}, then $(N_S - \mu_i \lambda_i,\ V_i)$ is eigenpair of H. Positive-definiteness of H can therefore be ensured, $\forall i \in \{1, \dots, N\}$:

$$(N_S - \lambda \mu_i) \geq (N_S - \lambda \mu_{max}) > 0 \tag{3.3}$$

This leads to the following constraint on λ:

$$(0 \leq) \lambda < \frac{N_S}{\mu_{max}} \tag{3.4}$$

3.1.1 Asymptotic behaviour

We denote by $\lambda_{rel} \in [0,\ 1[$ the normalised hyperparameter:

$$\lambda_{rel} = \lambda \frac{\mu_{max}}{N_S} \tag{3.5}$$

For $\lambda_{rel} = 0$, the optimisation problem is directly the minimisation of the quadratic loss between all detectors output and the sought target labels. The solution is directly given by:

$$\hat{Y_T} = \frac{1}{N_S} \left(\sum_{i=1}^{N_S} Y_T^{\{S_i\}} \right) \tag{3.6}$$

This can also directly be deduced from Equation (3.2).
For $\lambda_{rel} \to 1$, the solution behaviour is given by the eigenvalues γ_i of H^{-1}, which are $(\forall i \in \{1, \dots, N\})$:

$$\gamma_i = \frac{1}{N_S - \lambda\mu_i} = \frac{1}{N_S(1 - \lambda_{rel}\frac{\mu_i}{\mu_{max}})} \tag{3.7}$$

meaning that, if we denote γ_{max} the eigenvalue of H^{-1} associated to μ_{max}, we have:

$$\gamma_{max} \underset{\lambda_{rel}\to 1}{\longrightarrow} \infty$$

Let us consider the eigendecomposition of H; one can notice that it shares the same eigenvectors V_i as K_{TT}, $(\forall i \in \{1, \dots, N\})$,

$$(N_S\mathbb{I}_N - \lambda K_{TT})V_i = (N_S - \lambda\mu_i)V_i = \gamma_i^{-1}V_i.$$

and therefore we have:

$$H^{-1} = \sum_{i=1}^{N} \gamma_i V_i V_i' \underset{\lambda_{rel}\to 1}{\approx} \gamma_{max} V_{max} V_{max}'$$

where V_{max} denotes the eigenvector associated to γ_{max} and μ_{max}. In other words, the solution in this case is equivalent to:

$$\hat{Y_T} \underset{\lambda_{rel}\to 1}{\approx} \gamma_{max} V_{max} \left(V_{max}' \sum_{i=1}^{N_S} Y_T^{\{S_i\}} \right) \tag{3.8}$$

meaning that it is directly proportional to the eigenvector of K_{TT} associated to its greatest eigenvalue; the detectors output may only invert the classes, by changing the sign of the right part of the expression in parenthesis, which is a scalar.
To synthesise, QLTL solution is fully supervised at $\lambda_{rel} = 0$ − as it only depends on the detectors output − and fully unsupervised when $\lambda_{rel} \to 1$.

Illustration

In Figure (3.1), we illustrate QLTL on a toy example: upper left panel represents a source dataset and upper right panel a target dataset; on both panels, red pluses represent the positive class and blue circles the negative class; the black continuous line is the decision boundary of a detector trained on the source dataset. Lower left panel represents the target kernel matrix; as observations are sorted by label, the discriminative information contained in the kernel matrix appears clearly. Lower right panel gives the output of QLTL for three values of λ_{rel}: 0, in this case the problem is fully supervised and it is simply the output of the source detector, represented by the blue line with squares; 1, in that case the problem is fully unsupervised and the solution is the eigenvector associated to the greatest eigenvalue of the kernel matrix (it is blind to the classes, which may appear inverted), represented by the red line with triangles; 0.5, finally represents the compromise between both pieces of information through semi-supervision, as represented by the dashed line. The full black line is the ground-truth to be detected.
As can be seen, the unsupervised information contained in the Gram matrix is discriminative of the classes; there is much more mixture in the supervised information given by the source detector, and all observations would be detected as the negative class. By combining both pieces of information, we enable transfer learning and would achieve perfect detection on this toy example, for $\lambda_{rel} = 0.5$ and for a decision threshold set at zero on the dashed-line.

3.1.2 Extension: Bregman divergences

We sought a more general expression than the Quadratic Loss for the supervised part of our optimisation problem. Quadratic loss is a particular case of a more general measure: Bregman divergences.
A Bregman divergence is defined for a strictly convex, continuously-differentiable and real-valued function f, defined on a closed convex set Ω. For two points $\mathbf{x}, \mathbf{y} \in \Omega$, its general expression is:

$$B_f(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla_{\mathbf{y}} f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \tag{3.9}$$

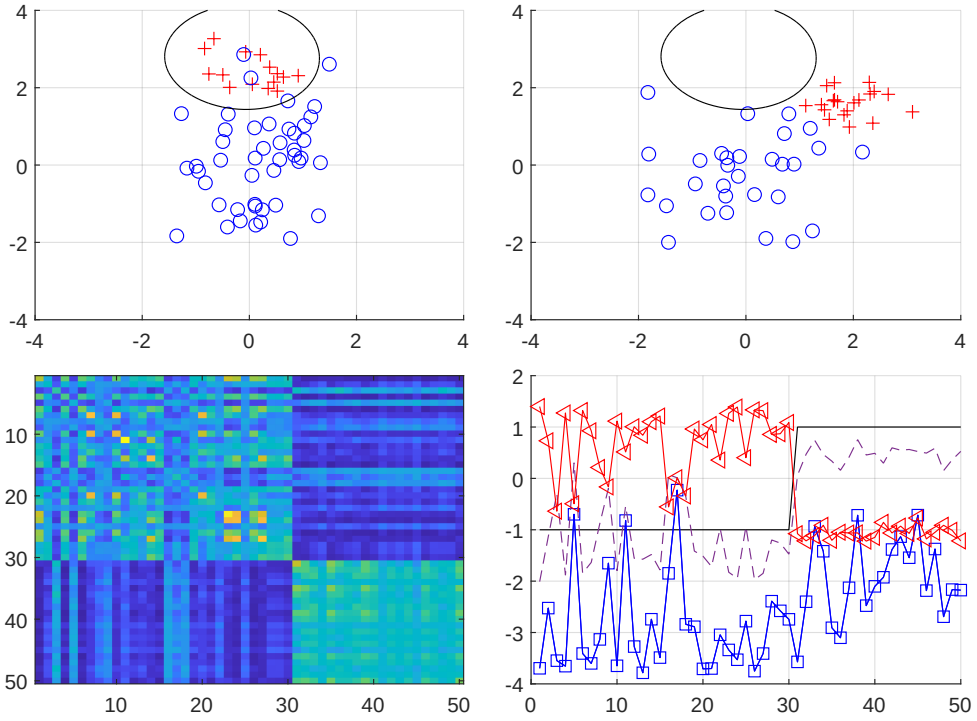


Figure 3.1: Illustration of QLTL on a toy example. Upper left panel: source dataset; upper right panel: target dataset. The red pluses class shifts between source and target datasets. Lower left panel: target kernel matrix; lower right panel: QLTL output for $\lambda_{rel} = 0$ (blue line with squares), for $\lambda_{rel} = 1$ (red line with triangles) and for $\lambda_{rel} = 0.5$ (dashed line). Further explanations given at the end of Subsection (3.1.1).

Domain Ω	Function f	Divergence B_f
$\mathbf{R}$	Squared function x^2	Norm $\mathcal{L}_2$ $(x-y)^2$
$\mathbf{R}^N$	Squared norm $\|\|\mathbf{x}\|\|_2^2$	Squared norm $\|\|\mathbf{x}-\mathbf{y}\|\|_2^2$
$\mathbf{R}$	Exponential $exp(x)$	Exponential loss $exp(x)-(x-y+1)exp(q)$
$\mathbf{R}^N$	Negative entropy $\sum_{i=1}^N x_i \log(x_i)$	Generalised Kullback-Leibler div. $\sum_{i=1}^N x_i \log(\frac{x_i}{y_i})$
$\mathbf{R}^N \backslash \{0_N\}$	$-\sum_{i=1}^N \log(x_i)$	Itakura-Saito distance $\sum_{i=1}^N \frac{x_i}{y_i} - \log(\frac{x_i}{y_i}) - 1$

Table 3.1: Common Bregman divergences from the literature.

We give a visual representation of a Bregman divergence in Figure (3.2) for a toy chosen function f, and give a short list of the most commonly used Bregman divergences in Table (3.1), along with their corresponding base function f.
Most Bregman functions are not symmetric; as a consequence, we have to consider two separate expressions for our extended QLTL method:

$$\hat{Y_T} = \underset{Y_T}{argmin} \sum_{i=1}^{N_S} \left(B_f(Y_T,\ Y_T^{\{S_i\}}) \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \tag{3.10a}$$

$$\hat{Y_T} = \underset{Y_T}{argmin} \sum_{i=1}^{N_S} \left(B_f(Y_T^{\{S_i\}},\ Y_T) \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \tag{3.10b}$$

In the two unnumbered subsections below, we will zero the gradient of each expression and check that we find the correct solution in the case of quadratic loss.

Sought labels on the left

$$\begin{aligned} &\nabla_{Y_T} \left(\sum_{i=1}^{N_S} \left(B_f(Y_T,\ Y_T^{\{S_i\}}) \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \right) \\ =& N_S \nabla_{Y_T} f(Y_T) - \sum_{i=1}^{N_S} \nabla_{Y_T^{\{S_i\}}} f(Y_T^{\{S_i\}}) - 2 K_{TT} Y_T = 0 \\ \equiv& N_S \nabla_{Y_T} f(Y_T) - 2 K_{TT} Y_T = \sum_{i=1}^{N_S} \nabla_{Y_T^{\{S_i\}}} f(Y_T^{\{S_i\}}) \end{aligned} \tag{3.11}$$

In the case of quadratic loss, $f(Y_T) = Y_T' Y_T$ and $\nabla_{Y_T} f(Y_T) = 2 Y_T$. By replacing accordingly in the above expression, we indeed obtain the solution of Equation (3.2).

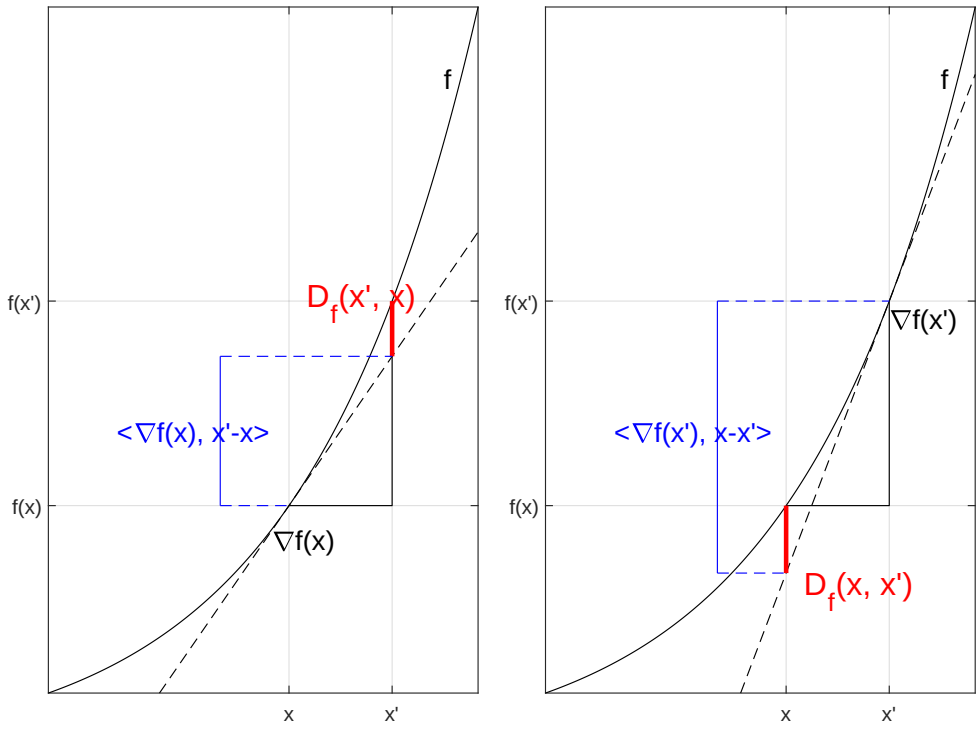


Figure 3.2: Visual representation of a Bregman divergence for a given couple of observations x and x' and a strictly convex and real-valued function f; Bregman divergences are generally non-symmetric; see how for this function f $B_f(\mathbf{x'}, \mathbf{x}) \neq B_f(\mathbf{x}, \mathbf{x'})$, as represented in bold red in both panels.

Sought labels on the right

$$\nabla\Big(\sum_{i=1}^{N_S}\big(B_f(Y_T^{\{S_i\}},\, Y_T)\big) - \lambda\langle K_{TT}, Y_T Y_T'\rangle_F\Big)$$

$$= -N_S\nabla f(Y_T) + N_S\nabla^2 f(Y_T)Y_T + N_S\nabla f(Y_T) - \nabla^2_{Y_T} f(Y_T)\sum_{i=1}^{N_S} Y_T^{\{S_i\}} - 2K_{TT}Y_T = 0 \tag{3.12}$$

$$\equiv N_S Y_T - 2(\nabla^2_{Y_T} f(Y_T))^{-1} K_{TT} Y_T = \sum_{i=1}^{N_S} Y_T^{\{S_i\}}$$

where $\nabla^2_{Y_T} f$ denotes the Hessian matrix $H_{Y_T}(f)$ of general term $H_{ij} = \frac{\partial^2 f}{\partial Y_{T_i} \partial Y_{T_j}}$.
In the case of quadratic loss, $\nabla^2_{Y_T} f(Y_T) = 2\mathcal{I}_N$, and here again this leads to the solution of Equation (3.2) and verifies the validity of this expression.

Conclusion on Bregman divergences

In Equations (3.11 and 3.12) we have given the expressions to be zeroed for any Bregman divergence, as extension of the quadratic loss; we also verified that in the case of quadratic loss, which is symmetric, both expression indeed lead to the solution of Equation (3.2).
Any considered Bregman divergence (e.g. Kullback-Leibner divergence, the squared Mahalanobis distance, among others) will then have to be solved individually by replacing f, ∇f and $\nabla^2 f$ in the expressions and by solving. As tested divergences gave similar or worse results than quadratic loss, and as there were more promising research leads, we moved on to other considerations, as presented in the next subsection.

Comment

During the review of the manuscript, one reviewer pointed out that the quadratic loss is the only symmetric Bregman divergence and that this property is most likely to have an impact on the performances. Considering that this comment was done after the end of our research work, it appears to be an excellent perspective to try connecting the asymmetry property of Bregman divergences (with the only exception of the quadratic loss) to the loss of performance of our methods.

3.2 Kernel Alignment Transfer Learning

In this section, we first introduce Kernel-Cross Alignment (KCA), an extension of KTA to measure the similarity between the cross-gram matrix K_{TS} computed from the target and source observations, and the cross-labels matrix computed using the estimated target and known source labels vectors, respectively. To our knowledge, this is new and has never been introduced in the literature before. We thereafter propose two methods of transfer learning simply relying on the maximisation of alignment with respect to the sought target labels vector Y_T. We denote by Kernel-Cross Alignment Transfer Learning (KCATL) the maximisation of KCA and Kernel-Target Alignment Transfer Learning (KTATL) the one of KTA [53]. We finally comment on KCATL limitations and on both methods operational setup.

3.2.1 Kernel-Cross Alignment

Some methods in the literature rely on the merging of source and target datasets and on the construction of a joint kernel matrix:

$$\begin{pmatrix} K_{TT} & K_{TS} \\ K_{ST} = K'_{TS} & K_{SS} \end{pmatrix} \tag{3.13}$$

where K_{TT} is the kernel matrix computed within target observations (as it was the case in QLTL, for instance), K_{SS} within source ones and K_{TS} between target and source observations.
As our problem is to transfer knowledge between source and target and to do so efficiently (computationally-wise), we initially focused on the matrix K_{TS} as a semi-supervised information conveyor for QLTL. Thereafter, we realised that there was a simpler way of using K_{TS} and even K_{TT} in order to transfer knowledge.
We begin by introducing Kernel-Cross Alignment (KCA), an extension of KTA to measure the similarity between K_{TS} the cross-gram matrix computed between source and target observations, and the matrix computed from source and target label vectors, respectively. KCA directly expresses as:

$$\begin{aligned} \mathcal{A}(K_{TS},\, Y_T Y'_S) &= \frac{\langle K_{TS},\, Y_T Y'_S \rangle_F}{\sqrt{\langle K_{TS},\, K_{TS} \rangle_F \langle Y_T Y'_S,\, Y_T Y'_S \rangle_F}} \\ &= \frac{Y'_T K_{TS} Y_S}{||K_{TS}||_F ||Y_T|| ||Y_S||} \end{aligned} \tag{3.14}$$

where K_{TS} denotes the cross-Gram matrix computed between target and source observations, X_T and X_S, respectively; Y_S denotes the source labels in column vector and Y_T the target ones. Equivalently to KTA and KTP, we denote by Kernel-Cross Polarisation (KCP) $\mathcal{P}$ the non-normalised equivalent to KCA:

$$\mathcal{P}(K_{TS},\, Y_T Y'_S) = \langle K_{TS},\, Y_T Y'_S \rangle_F = Y'_T K_{TS} Y_S \tag{3.15}$$

KxATL

If we further compare Kernel-Target Alignment and Kernel-Cross Alignment, both measures are composed of a clearly non-supervised part (the kernel matrices K_{TT} and K_{TS}, respectively) and of a (semi)-supervised part (the label matrices $Y_T Y'_T$ and $Y_T Y'_S$, respectively). By **directly** maximising these alignments with respect to the sought target labels Y_T, we obtain semi-supervised solutions that will take advantage of both the non-supervised information contained in the Gram matrix and the supervised information given by either an estimate of the target labels (using a detector trained on the source) in the case of KTA maximisation, or directly using the source labels in the case of KCA maximisation.

We therefore proposed in [53] two simple methods of kernel-based transfer learning, namely Kernel-Cross Alignment Transfer Learning (KCATL) and Kernel-Target Alignment Transfer Learning (KTATL). As it has just been introduced, the former is based on the maximisation of KCA w.r.t. the sought target labels:

$$\hat{Y_T} = \underset{Y_T}{argmax}\ \mathcal{A}(K_{TS}, Y_T Y_S') \tag{3.16}$$

The maximisation of KCP $(Y_T' K_{TS} Y_S)$ w.r.t. Y_T is obtained for Y_T collinear to $K_{TS}Y_S$, meaning that there is an infinity of solutions; this will lead to an issue of selecting the right threshold of decision for this method. We temporarily evacuate this problem by imposing a unit-norm constraint on the sought solution, that is to impose $||\hat{Y_T}|| = 1$. Accordingly, $\hat{Y_T}$ is then given by:

$$\hat{Y_T} = \frac{K_{TS}Y_S}{||K_{TS}Y_S||} \tag{3.17}$$

With the same constraints, KCA is maximised w.r.t. Y_T by the same solution, indeed:

$$\hat{Y_T} = \underset{Y_T}{argmax} \frac{Y_T' K_{TS} Y_S}{||K_{TS}||_F\ ||Y_S||\ ||Y_T||} = \underset{Y_T}{argmax} \frac{Y_T'}{||Y_T||} K_{TS} Y_S \tag{3.18}$$

Meaning that the output of KCATL is:

$$\hat{Y_T} = \frac{K_{TS}Y_S}{||K_{TS}Y_S||} \tag{3.19}$$

Similarly, in the case of KTATL we maximise KTA w.r.t. the target labels, and we use a source detector output on the target data, $Y_T^{\{S\}}$, to input supervision:

$$\hat{Y_T} = \underset{Y_T}{argmax}\ \mathcal{A}(K_{TT}, Y_T^{\{S\}} Y_T') \tag{3.20}$$

Using similar logic, the solution is given by:

$$\hat{Y_T} = \frac{K_{TT}Y_T^{\{S\}}}{||K_{TT}Y_T^{\{S\}}||} \tag{3.21}$$

Contrary to QLTL, where KTA is a quadratic function of Y_T (optimisation of which leads to the resolution of a linear system), here the optimisation problem is linear in Y_T, the other part of KTA being given by a source detector and constituting the supervised information of the problem. Moreover, QLTL considered the output of multiple sources; in the case of KxATL, a way to simply compromise between multiple sources could be to weight the decisions using, for example, each obtained alignment.
Also, it is noteworthy that due to the infinity of solutions to Equations (3.16 and 3.20), the unit-norm constraint only serves as to select a unique solution.

3.2.2 Limitations of KCATL

In the case of a radial-basis function (our case, the gaussian kernel), the solution of KCATL corresponds to an estimator of Bayes' detector using a gaussian Parzen window. Indeed, by development of Equation (3.19) for a target observation $\mathbf{x}_{T_i}$, KCATL output is (ignoring a multiplication factor):

$$\hat{y_{T_i}} \approx \sum_{j=1}^{N_S} y_{S_j} K_{T_i S_j} = \sum_{y_{S_k}=1} K_{T_i S_k} - \sum_{y_{S_l}=-1} K_{T_i S_l} \tag{3.22}$$

where N_S is the number of observations in the source S. This is, in the case of a radial-basis kernel acting as the Parzen window and when using the true ± 1 source labels, an estimator of:

$$\begin{aligned} p(y_{S_j}=1)p(\mathbf{x}_{T_i}|y_{S_j}=1) - p(y_{S_j}=-1)p(\mathbf{x}_{T_i}|y_{S_j}=-1) \\ \equiv p(y=1|\mathbf{x}_{T_i}) - p(y=-1|\mathbf{x}_{T_i}) \end{aligned} \tag{3.23}$$

which corresponds to Bayes detector with costs $0,\ 1$. **However**, while this seems to indicate that KCATL does not allow for knowledge transfer, it is only true in the case of labels coding $\in \{\pm 1\}$ and for a non-centred kernel matrix. In our experiments, kernel matrices are always centred on at least the average observation in feature space; this is done using Equation (2.18) for kernel-target matrix centring, or using the adaptation below for cross-kernel matrix centring:

$$K_{TSC} = C_{N_T} K_{TS} C_{N_S} \tag{3.24}$$

where C_N is the centring matrix (refer to Equation (2.17)) of size $N \times N$.
KCATL effet and mean-centring benefit are illustrated in Figure (3.3): upper left and upper right figures represent a source and target toy datasets, respectively, where blue circles represent the negative class and red pluses the positive one. Both bold lines represent the *ideal* decision boundary - ideal as it was obtained by training on the target data - here, using a support vector machine - if we had knowledge of the ground-truth labels. On both lower panels are represented outputs of KCATL in transfer from the upper left source to the upper right target; however, lower left panel is for a non-centred kernel and lower right panel is for an optimised centre (as developed in the upcoming section). As can again be seen, kernel centring plays a critical role in our method performances and allows the transfer of knowledge in the case of KCATL (as it would not have been the case for a non-centred kernel, as it was just explained).
In practice, we observed overall improved performances on the application (as compared to the current reference REM detector). Moreover, the key idea of Kernel-Cross Alignment remains highly useful in the case of Training Source Selection, as this is presented further in this chapter, in Section (3.4).

3.2.3 Operational setup

As to optimise KxATL performances, we have considered multiple approaches:

- For the inputted supervised information, whether to use the labels or (binarised or not) detectors output. Overall, decision statistics resulted in marginally better results in transfer.
  - Ground-truth labels Y_S or re-estimated labels $Y_S^{\{S\}}$ (by application of the source detector on the source data), in the case of KCATL.
  - Binarised output $sign(Y_T^{\{S\}})$ or direct output $Y_T^{\{S\}}$ in the case of KTATL.
- How to improve the *quality* of the Gram matrix ? We have considered multiple approaches to that problem (a few are cited below); however, it mainly resulted in negligible effects on both methods performances.
  - Rejecting outliers by removing observations in low density regions, using, for instance, the average of the Gram matrix by column or row, using the similarity between this average and the probability density function estimate using Parzen window.
  - De-noising, in the case of KTATL, by reconstructing the Gram matrix using the N *first* /greatest eigenvalues and associated eigenvectors.
- Centring the Gram matrix is a subject on its own. It is done using the work in [49] in the case of KTATL and we propose an adaptation in the case of KCATL in [54], which is extensively presented in the following section.

Illustration

In Figure (3.4) we illustrate both methods on the same toy dataset as in Figure (3.1); on the upper left and lower left panels are represented the kernel-target matrix and the kernel-cross matrix, respectively. These lead to the red curves (with triangles) on the corresponding right panels, which are the decision statistics outputted by the respective methods, KTATL and KCATL. As can be seen, both will outperform the reference method (SVM as presented in Figure (3.1)) by a lot. However, the decision statistics of KTATL seem slightly better than the one of KCATL, which is confirmed by separability measures and by the ROC curves.

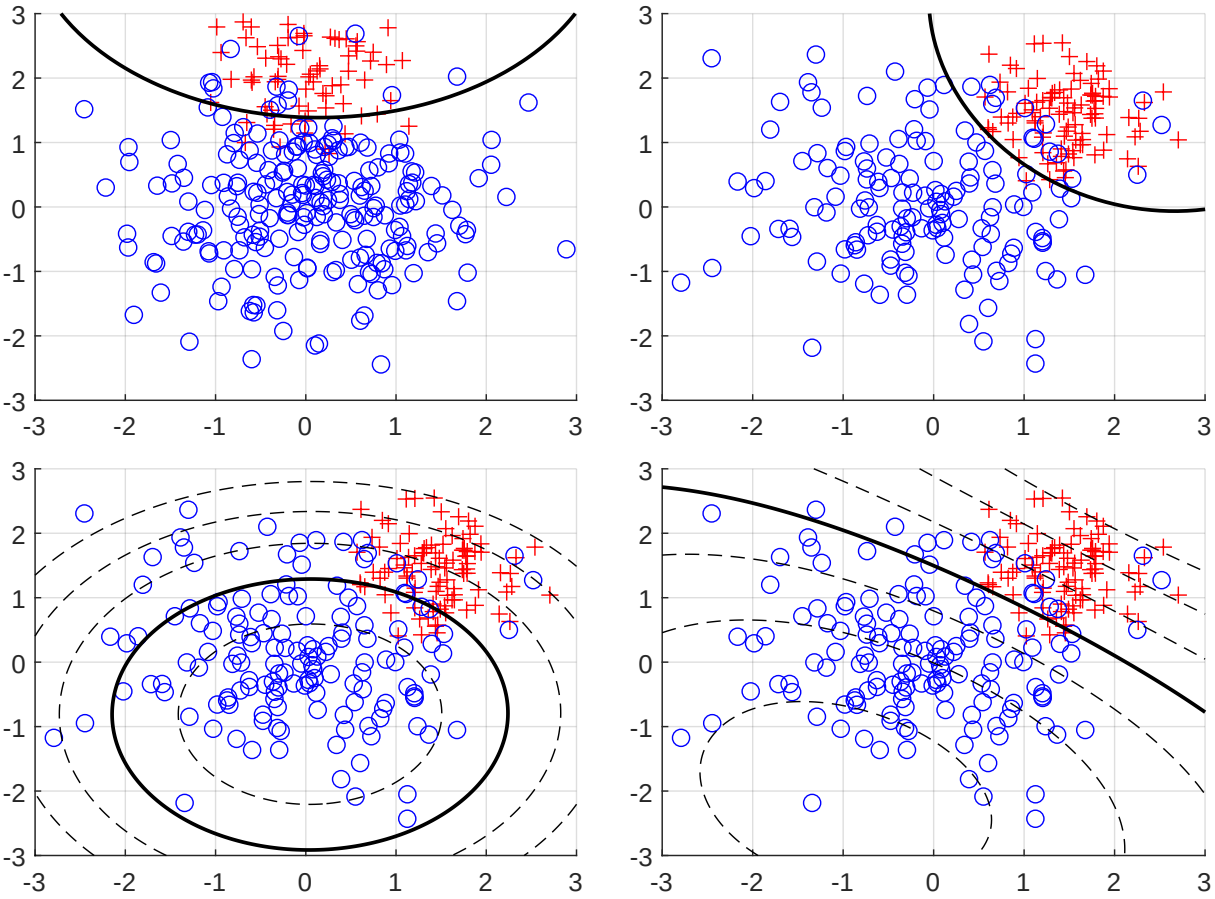


Figure 3.3: Illustration on how KCATL transfers knowledge in the correct operational setup. Blue circles represent the negative class, red pluses the positive one. Upper left panel: source dataset with trained decision boundary; upper right: target dataset with ideal decision boundary (if labels were known). Lower left panel: KCATL decision boundaries for a non-centred kernel; lower right: for an optimised centred-kernel matrix.

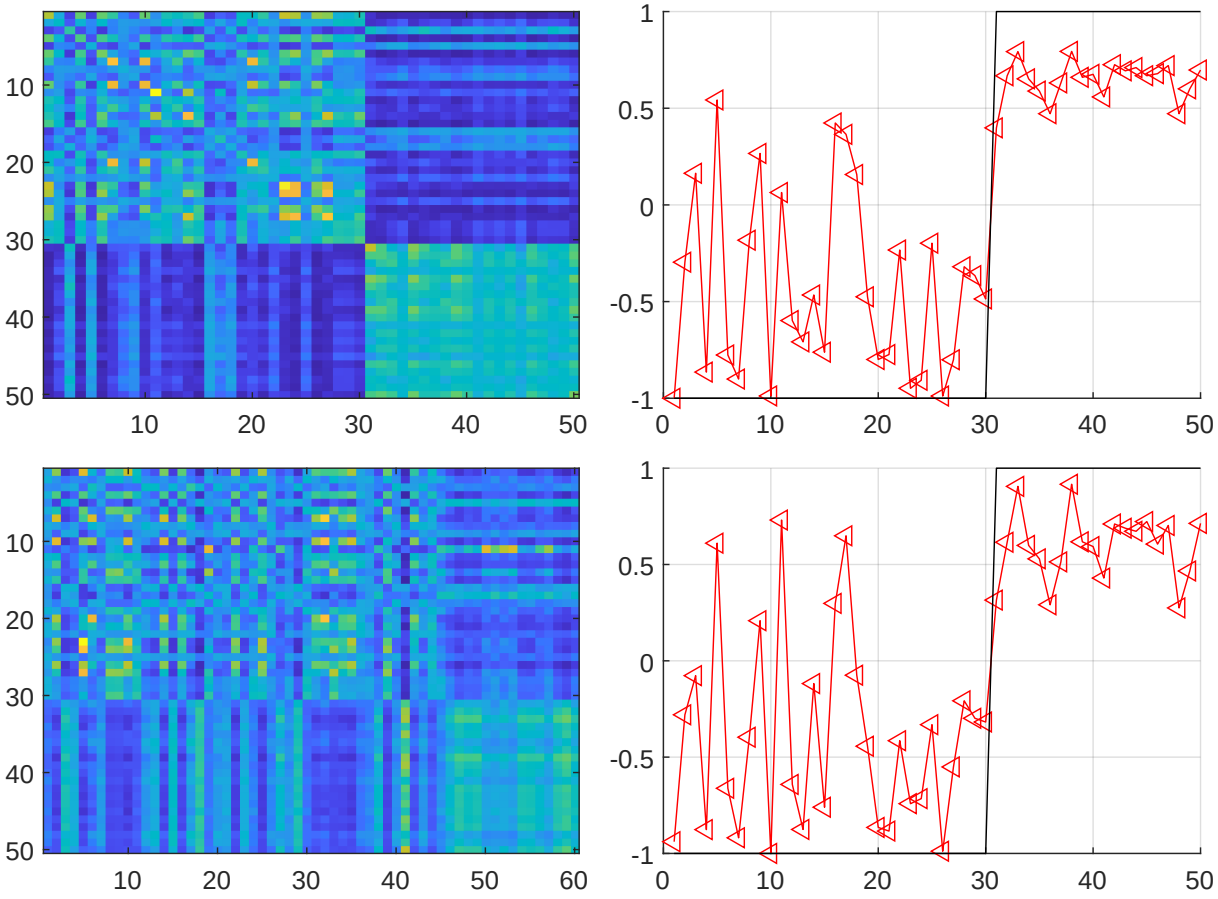


Figure 3.4: Illustration of KxATL - explanations given in the last paragraph of Subsection (3.2.3). Upper left panel: target kernel matrix K_{TT}, lower left panel: cross-kernel matrix K_{TS}. Upper right panel: output of KCATL, lower left panel: output of KTATL.

3.3 Optimisation through Kernel centring

In Subsection (2.4.5) we have given an introduction to kernel centring in the case of a target kernel matrix, computed within a set of target observations. As we have introduced KCA in Subsection (3.2.1), we update here the tackling of kernel centring, as there are now two centres to consider: one for the source, one for the target [54]. Thereafter, in the second part of this section - subsection (3.3.2), we present a direct application of kernel centring for transfer learning purposes.

3.3.1 Kernel cross-matrix centring

We update Equation (2.14) for kernel cross-matrices, whose terms now are:

$$K_{T_i S_j} = \langle \phi(\mathbf{x}_{T_i}) - C_T,\ \phi(\mathbf{x}_{S_j}) - C_S \rangle_{\mathcal{H}} \tag{3.25}$$

where $\mathbf{x}_{T_i}$ is a given target observation, $\mathbf{x}_{S_j}$ a source one, and C_T and C_S are the target and source centres, respectively. As for the traditional kernel matrices, we now consider two strategies of kernel centring:

Using the representer theorem

In this case our centres express as:

$$C_T = \sum_{i=1}^{N_T} \alpha_i \phi(\mathbf{x}_{T_i}) \qquad C_S = \sum_{j=1}^{N_S} \beta_j \phi(\mathbf{x}_{S_j})$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are weighing vectors. Once inputted in Equation (3.25):

$$\begin{aligned} K_{C_{kl}} &= \left\langle \phi(\mathbf{x}_{T_k}) - \sum_{i=1}^{N_T} \alpha_i \phi(\mathbf{x}_{T_i}),\ \phi(\mathbf{x}_{S_l}) - \sum_{j=1}^{N_S} \beta_j \phi(\mathbf{x}_{S_j}) \right\rangle_{\mathcal{H}} \\ &= \langle \phi(\mathbf{x}_{T_k}),\ \phi(\mathbf{x}_{S_l}) \rangle_{\mathcal{H}} - \sum_{i=1}^{N_T} \alpha_i \langle \phi(\mathbf{x}_{T_i}),\ \phi(\mathbf{x}_{S_l}) \rangle_{\mathcal{H}} \\ &\quad - \sum_{j=1}^{N_S} \beta_j \langle \phi(\mathbf{x}_{T_k}),\ \phi(\mathbf{x}_{S_j}) \rangle_{\mathcal{H}} + \sum_{i=1}^{N_T} \sum_{j=1}^{N_S} \langle \phi(\mathbf{x}_{T_i}),\ \phi(\mathbf{x}_{S_j}) \rangle_{\mathcal{H}} \\ &= K_{T_k S_l} - \sum_{i=1}^{N_T} \alpha_i K_{T_i S_l} - \sum_{j=1}^{N_S} \beta_j K_{T_k S_j} + \sum_{i=1}^{N_T} \sum_{j=1}^{N_S} K_{T_i S_j} \end{aligned} \tag{3.26}$$

This can easily be put in matrix form, that is:

$$K_C = K_{TS} - \Gamma'_{\boldsymbol{\alpha}} K_{TS} - K_{TS} \Gamma_{\boldsymbol{\beta}} + \Gamma'_{\boldsymbol{\alpha}} K_{TS} \Gamma_{\boldsymbol{\beta}} \tag{3.27}$$

where $\Gamma_{\boldsymbol{\alpha}}$ and $\Gamma_{\boldsymbol{\beta}}$ are matrices of size $N_T \times N_T$ and $N_S \times N_S$, respectively:

$$\Gamma_{\boldsymbol{\alpha}} = \begin{pmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & \alpha_2 & \cdots & \alpha_2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{N_T} & \alpha_{N_T} & \cdots & \alpha_{N_T} \end{pmatrix} \qquad \Gamma_{\boldsymbol{\beta}} = \begin{pmatrix} \beta_1 & \beta_1 & \cdots & \beta_1 \\ \beta_2 & \beta_2 & \cdots & \beta_2 \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{N_S} & \beta_{N_S} & \cdots & \beta_{N_S} \end{pmatrix}$$

Directly from input space

In that case centres express as:

$$C_T = \phi(\mathbf{x}_{C_T}) \qquad C_S = \phi(\mathbf{x}_{C_S})$$

where $\mathbf{x}_{C_T}$ and $\mathbf{x}_{C_S}$ are elements from the input space.
Similarly, by development and once put in matrix form, we obtain the following expression for the centred cross-Gram matrix:

$$K_C = K_{TS} - K_{TC_S}\mathbb{I}_{1\times N_S} - \mathbb{I}_{N_T\times 1}K_{C_TS} + K_{C_TC_S}\mathbb{I}_{N_T\times N_S} \tag{3.28}$$

where K_{TC_S} denotes the kernel between target observations and source centre in column form, K_{C_TS} between target centre and source observations in row form, and $K_{C_TC_S}$ between both centres (a scalar). Finally, $\mathbb{I}_{a\times b}$ denotes the all-ones matrix of size $a \times b$.

Mean centring

Mean centring can also be done in the case of two centres. It uses the same centring matrix C_N as presented in Subsection (2.4.5), and is simply given by:

$$K_C = C_{N_T}KC_{N_S} \tag{3.29}$$

3.3.2 Dual optimisation w.r.t. centres and labels

In the previous subsection, we have provided the reader with the centred cross-kernel expressions for both centring strategies (i.e. in input space and using the representer theorem). In this subsection, we now use these expressions to optimise KCA (or KCP) optimisation w.r.t. the sought target labels and the centres, as we did in [54]. Our *general* optimisation problem expresses as:

$$\max_{C_T,\, C_S,\, Y_T} J(Y_T, K_C(C_T,\, C_S)) \tag{3.30}$$

where J is the quantity to be maximised (as mentioned above, KCA or KCP), Y_T are the sought target labels, and K_C is the observed centred matrix which depends on the choice of both centres C_T and C_S.
As it was demonstrated in Section 3.2, the optimisation w.r.t. Y_T is immediate and analytical (refer to Equation (3.19)):

$$\hat{Y_T} = \underset{Y_T}{argmax}\ J(Y_T, K_C(C_T,\, C_S)) = \frac{K_CY_S}{||K_CY_S||} \tag{3.31}$$

We therefore propose an alternate optimisation scheme and will now resolve the optimisation w.r.t. the centres for a fixed sought labels vector.
We now consider the two centring strategies, either using the representer theorem (which corresponds to an optimisation in the RKHS) or directly by selecting elements of the input space.

Optimisation in the RKHS

In this case the optimisation will be done on the weighing vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ and for fixed target labels Y_T; we solve w.r.t. both weighing vectors separately; first on the target centre:

$$\max_{\boldsymbol{\alpha}}\ \ J(K_C(\boldsymbol{\alpha},\, \boldsymbol{\beta})) \tag{3.32}$$

We develop the gradient in two separates parts:

$$\begin{aligned}\nabla_{\boldsymbol{\alpha}}J &= \sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\frac{\partial J}{\partial K_{C_{ij}}}\nabla_{\boldsymbol{\alpha}}K_{C_{ij}} \\ &= \sum_{i=1}^{N_T}\sum_{j=1}^{N_S}G_{ij}\nabla_{\boldsymbol{\alpha}}K_{C_{ij}}\end{aligned} \tag{3.33}$$

The right part (gradient of K_C on $\boldsymbol{\alpha}$) is of general term (obtained from Equation (3.26)):

$$\frac{\partial K_{C_{ij}}}{\partial_{\alpha_l}} = -K_{lj} + \sum_{k=1}^{N_S}\beta_kK_{lk} \tag{3.34}$$

Then, one can notice that:

$$\begin{aligned}\sum_{i=1}^{N_T}\sum_{j=1}^{N_S} G_{ij}K_{lj} &= \sum_{j=1}^{N_S} K_{lj}\sum_{i=1}^{N_T} G_{ij} \\ &= \left(\sum_{j=1}^{N_S} K_{lj}\right)(G_{j\star}\mathbf{1}_{(N_T,\,1)}) \\ &= \mathbf{1}_{(1,\,N_T)}GK_{\star l}\end{aligned} \tag{3.35}$$

where $G_{j\star}$ denotes the j^{st} row of the gradient matrix G and similarly $K_{\star l}$ the l^{st} column of the non-centred kernel matrix K. Finally, by completing the development of the optimisation problems with the above expressions, and once put in matrix form, we obtain:

$$\nabla_{\boldsymbol{\alpha}} J = K(\lambda\boldsymbol{\beta} - G'\mathbf{1}_{(N_T,\,1)}) \tag{3.36}$$

where λ is the sum of all elements of G: $\lambda = \mathbf{1}_{(1,N_T)}G\mathbf{1}_{(N_S,1)}$. The updated target centre therefore expresses as:

$$\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \eta K(\lambda\boldsymbol{\beta} - \mathbf{G'}\mathbf{1}_{(N_T,\,1)}) \tag{3.37}$$

where η is the gradient step size.
Optimising on the source centre is done following a similar development:

$$\begin{aligned}\max_{\boldsymbol{\beta}} \quad & J(K_C(\boldsymbol{\alpha},\,\boldsymbol{\beta})) \\ \frac{\partial K_{C_{ij}}}{\partial_{\beta_k}} &= -K_{ik} + \sum_{l=1}^{N_T}\alpha_l K_{kl} \\ \sum_{i=1}^{N_T}\sum_{j=1}^{N_S} G_{ij}K_{ik} &= K'_{\star k}G\mathbf{1}_{(N_S,\,1)} \\ \nabla_{\boldsymbol{\beta}} J &= K'(\lambda\boldsymbol{\alpha} - G\mathbf{1}_{(N_S,\,1)})\end{aligned} \tag{3.38}$$

which finally leads to the following expression for updating the source centre:

$$\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} + \eta K'(\lambda\boldsymbol{\alpha} - G\mathbf{1}_{(N_S,\,1)}) \tag{3.39}$$

Finally, the gradient matrix G must be developed; our main interests are the maximisation of KCA and KCP; we therefore give the corresponding matrix G for both problems:

$$\begin{aligned}G_{KCA} &= \nabla_{K_C}\mathbf{KCA}(Y_T,\,Y_S,\,K_C) \\ &= \nabla_{K_C}\left(\frac{Y_T'K_CY_S}{||K_C||_F||Y_S||\;||Y_T||}\right) \\ &= \frac{Y_TY_S'}{||K_C||_F||Y_T||\;||Y_S||} - \frac{Y_T'K_CY_S||Y_T||\;||Y_S||K_C}{(||K_C||_F||Y_T||\;||Y_S||)^2||K_C||_F} \\ &= \frac{Y_TY_S'}{||K_C||_F||Y_T||\;||Y_S||} - \mathbf{KCA}(Y_T,\,Y_S,\,K_C)\frac{K_C}{||K_C||_F^2}\end{aligned} \tag{3.40}$$

$$\begin{aligned}G_{KCP} &= \nabla_{K_C}\mathbf{KCP}(Y_T,\,Y_S,\,K_C) \\ &= \nabla_{K_C}(Y_T'K_CY_S) = Y_TY_S'\end{aligned} \tag{3.41}$$

Optimisation in the input space

We can use a similar approach as in feature space here by considering the kernel $k(\mathbf{x}_i,\,\mathbf{x}_j) = \mathbf{x}_i'\mathbf{x}_j$. However, we opted for a different approach by looking directly for an element of input space as centre. Consequently, in the case

of input space centring, the gradient matrices are the same as above; however, that second centring strategy requires resolution for each specific type of kernel. We give here the gradient expressions in the case of the gaussian kernel:

$$k(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(- \frac{||\mathbf{x}_1 - \mathbf{x}_2||^2}{2\sigma^2} \right) \tag{3.42}$$

We have, when optimising w.r.t. the target centre:

$$\nabla_{\mathbf{x}_{C_T}} K_{ij} = \frac{K_{C_T S_j}(\mathbf{x}_{C_T} - \mathbf{x}_{S_j}) - K_{C_T C_S}(\mathbf{x}_{C_T} - \mathbf{x}_{C_S})}{\sigma^2} \tag{3.43}$$

where $K_{C_T S_j}$ is the kernel between the target centre and the j^{st} source observation and $K_{C_T C_S}$ between both centres. By using a similar strategy as in the RKHS, this leads to the following update of the target centre:

$$\mathbf{x}_{C_T} \leftarrow \mathbf{x}_{C_T} + \frac{\eta}{\sigma^2}\left(- \mathbf{1}_{(1,N_T)}\mathbf{G}\mathbf{1}_{(N_S,1)} K_{C_T C_S}(\mathbf{x}_{C_T} - \mathbf{x}_{C_S}) + \sum_{j=1}^{N_S} \mathbf{1}_{(1,N_T)}\mathbf{G}_{*j} K_{C_T S_j}(\mathbf{x}_{C_T} - \mathbf{x}_{S_j}) \right) \tag{3.44}$$

Resolution for the source centre uses again similar logic:

$$\nabla_{\mathbf{x}_{C_S}} K_{ij} = \frac{K_{T_i C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{T_i}) - K_{C_T C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{C_T})}{\sigma^2} \tag{3.45}$$

$$\mathbf{x}_{C_S} \leftarrow \mathbf{x}_{C_S} + \frac{\eta}{\sigma^2}\left(- \mathbf{1}_{(1,N_T)}\mathbf{G}\mathbf{1}_{(N_S,1)} K_{C_T C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{C_T}) + \sum_{i=1}^{N_T} \mathbf{G}_{i*}\mathbf{1}_{(N_S,1)} K_{T_i C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{T_i}) \right) \tag{3.46}$$

3.3.3 Resulting algorithm and comments

Whatever the chosen centring strategy, the optimisation algorithm unfolds as follows:

1. Initialise both centres C_{T_0}, C_{S_0} and the target labels Y_{T_0}.
2. Compute K_C, and $\mathbf{G}$.
3. Update C_T and K_C accordingly. Compute the new $\mathbf{G}$.
4. Update C_S and K_C accordingly. Compute the new $\mathbf{G}$.
5. Update Y_T using Equation (3.17).
6. Repeat steps 2. to 5. until convergence.

In the remainder of this subsection, we comment on certain interesting aspects of this algorithm and transfer learning method: KCATL ROC curve independence to the target centre, and concentration of information in a single scalar in case of centring in the RKHS. We thereafter propose all considered operational setups for kernel centring (e.g. one could choose not to centre on the target), and conclude by illustrating the algorithm and the results in transfer learning.

ROC curve independence to the target centre

In the case of KCATL, the target centre $\mathbf{x}_{c_T}$ has no *direct* impact on the ROC curve. This is easily shown by developing the centred kernel matrix in Equation (3.17) using Equation (3.27):

$$\hat{Y_T} = \frac{(K_{TS} - \Gamma_{\boldsymbol{\alpha}}^T K_{TS} - K_{TS}\Gamma_{\boldsymbol{\beta}} + \Gamma_{\boldsymbol{\alpha}}^T K_{TS}\Gamma_{\boldsymbol{\beta}})Y_S}{||K_C Y_S||}$$

One can observe that:

$$(-\Gamma_{\boldsymbol{\alpha}}^T K_{TS} + \Gamma_{\boldsymbol{\alpha}}^T K_{TS}\Gamma_{\boldsymbol{\beta}})Y_S = C \times \mathbf{1}_{(N_T,\,1)}$$

where

$$C = \sum_{j=1}^{N_S}\sum_{i=1}^{N_T}(\alpha_i + \alpha_i\beta_j)K_{T_i S_j}Y_{S_j}$$

meaning that for different target centres, decision statistics are bijectively related and the resulting ROC curves are identical.

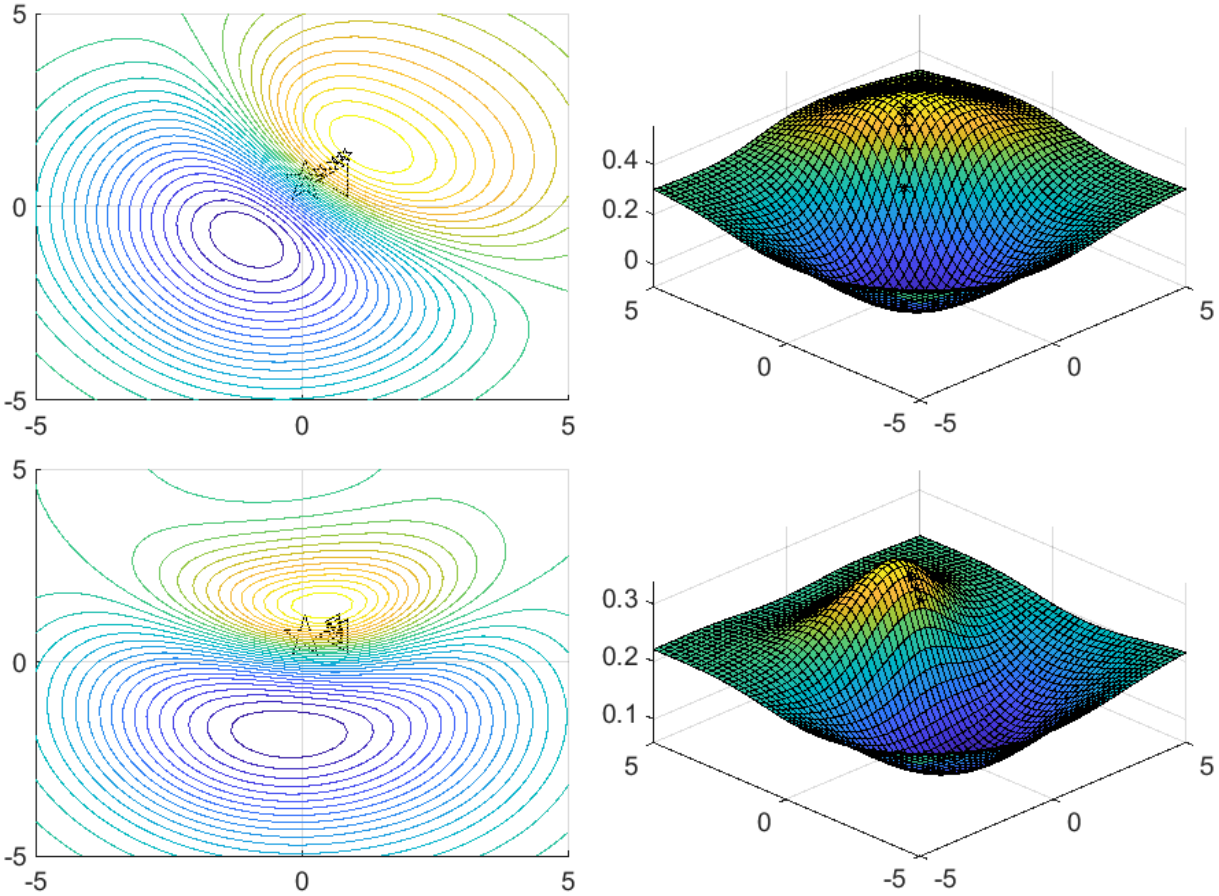


Figure 3.5: Illustration of the good behaviour of the optimisation algorithm in direct space, in the case of the radial-basis function kernel, when maximising KCA on one centre with the other centre fixed. Upper panels: optimisation w.r.t. the source centre; lower panels: optimisation w.r.t. the target centre; left panels: contour lines, corresponding to the surface (KCA values depending on the centres) displayed in the right panels.

Solution dependence to a single scalar

An interesting result is that, in the case of centring in feature space, the ROC curve will be driven by a single scalar:

$$C = \sum_{j=1}^{N_S} \beta_j y_{S_j} \tag{3.47}$$

Illustrations

In Figure (3.5) we illustrate the case of KCA optimisation in direct space w.r.t. each centre separately:

Conclusion: multiple setups for kernel centring

Centring in the RKHS are generally ill-posed problems (local maximisers). We have extensively studied multiple set-ups for such a problem, for instance:

- Optimisation w.r.t. the sole source centre or w.r.t. both centres (as the solution only *indirectly* depends on the target centre).
- Optimisation of KCA or of KCP (while we gave expression for any optimisation problem, these two are our main interest).

- Optimisation using the representer theorem or directly in input space, as described in Equations (3.27 and 3.28), respectively.
- Addition of constraints on the centres.

While, as we have shown in [54] and rapidly developed in Annex B, there is a clear interest in the method (one of the setups clearly outperformed previously used centring methods), there are a few remaining methodological locks yet to be resolved:

- How to choose the initial centre(s) to ensure optimal results ?
- Which data properties may help to select the best centring method/transfer learning method to ensure improved performances ?
- Which value of threshold to apply to the decision statistic obtained through KxATL (strong dependence to the target labels priors, which are unknown) ?

3.4 Training Source Selection

We remind the objective of Training Source Selection: as there is a plethora of source recordings, and as they should share similar statistical properties with the targets, for a given target, there should exist a selection of sources that will result in improved classification performances. There are two complementary points of view:

- *Optimal* sources should have reduced data shift as compared to other sources, for a given target. If that is the case, there should be a high correlation between the reference detector output and the ones given by the transfer learning methods, as there is less to transfer.
- For optimal sources, the - unsupervised - information contained in the cross-Gram matrix should relate very well with the - (semi-)supervised - one contained in the cross-labels matrix. One direct and easy way to assess such relatedness is through the measure of alignment.

Be it with either methods, we are able to compute measurements that should relate with the actual classification for a source applied to a given target. We have considered, as candidates for such measurements:

- The *observed* alignment (be it KCA or KTA) where the labels are estimated directly by the source detector.
- The *maximised* alignment (KCA or KTA) where target labels are the ones that maximise alignment - that is, the outputs of KCATL/KTATL.
- The *observed* correlation between the output of the reference source detector and the (binarised, or not) output of any of the transfer methods.
- The *maximised* correlation; same as previous bullet point but correlation is maximised w.r.t. threshold of decision on the target and/or on the source estimated labels, and the detector output is binarised using this maximising threshold.

The operational setup of these measurements includes the binarisation or not of the outputted decision statistics and the centring of the Gram matrices.
Once the measurement, or *predictor*, has been estimated for each source for a given target, the output of training set selection is:

- Either the output of the source for which the predictor is maximal.
- Or the fused output of a pre-determined number N of sources for which the predictor is highest; fusion may be through simple (weighted) mean of selected sources output, while other methods may perform better (QLTL).

As it will be demonstrated in the chapter presenting experimental results, quality of performances estimation and training source selection seems to be highly linked to target (hidden) properties: for some target recordings, our predictions are highly accurate, while for others it is not the case. Our work around training source selection has resulted in the filing of a patent at the European Patent Office (EPO) [55].

3.4.1 Quality of the selection

We consider that there are three capital aspects that will determine the quality of Training Source Selection (TSS):

- The quality of the predictor. Obviously, the better the predictor relates to the actual performances, the better the selection. On both our toy example and real application, observed KCA and KTA seem to outperform other measurements.
- The quality of the source detectors. The set of considered sources should ideally be diverse (as to make the selection interesting) and constituted of efficient detectors (that is, on the recordings on which it will be applied).
- The quality of the target recording. If there is a lot of mixture in the data and associated labels, there is nothing to be done: every source detector will fail.

Concept shift: an unsolvable problem

An unsolvable problem – that is, without meta/outside information that may help – is concept shift: all our proposed predictors DO NOT depend on the target true labels (as they are unknown/inaccessible). If for some reason, ground-truth target labels were to be shuffled, all our predictors would output EXACTLY the same values. These are sensitive to prior and covariate shifts, but are absolutely insensitive to a *pure* concept shift (exactly the same observations but with different labels).

Improving the selection

In order to improve TSS, one ought to tackle each of its components separately, that is:

- Study the predictors extensively for their improvement through operational setup.
- Cherry-pick the source detectors by creating an efficient sub-selection of sources: ideas to do so include:
  - Methods of clustering to potentially merge sources sharing statistical similarities.
  - Study each source overall performance to identify underperforming sources.
  - Methods of combining information efficiently, as aforementioned (voting, weighing using the predictors values, and so on).
- There is not much to do to assess a target recording quality (apart from studying the signals' overall quality). However, one may be encouraged to use the predictors to do **poor target rejection**: if for a given target, a given predictor is low for all considered sources, then there are good odds that performances will be low for that target, no matter which source is selected. This is a secondary application of TSS.

3.4.2 Results on a toy example

We generate four source datasets, represented in Figure (3.6) and nine target datasets - in Figure (3.7) - on which we want to achieve good classification performances. The sources properties have been chosen to ensure their diversity including prior shift (difference in labels prior probabilities); target properties have been randomised, except for target number nine (lower right panel) whose properties have been chosen to give an example where no source will perform well. The resulting sources and targets are similar to our application data in terms of class mixture and data shift.
We then compute, for each (source, target) couple, the observed KTA, the observed KCA, the observed correlation between outputs with and without transfer (resp. "Cross-correlation" when using KCATL as the method of transfer and "Target-correlation" when using KTATL).
In Figure (3.8) we represent the results of TSS: in x-axis the predictor values to be compared to the actual performances in the y-axis. The correlation of both axes is resp. 0.4896 for the observed KTA (upper left panel), 0.4348 for the observed KCA (upper right), 0.5608 for the correlation between KTATL and reference detector outputs (an SVM - lower left), and 0.5608 for the correlation between KCATL and reference (lower right).

Predictor	Best source	Second best	Second worst	Worst source
Obs. KTA	2	3	0	4
Obs. KCA	5	1	0	3
Cross-correlation	6	1	0	2
Target-correlation	6	1	0	2

Table 3.2: Count of how many times training source selection selects the best, second best, second worst and worst source, for each predictor and every target.

More importantly, we study each target individually to assess whether or not the predictor will allow a good source selection. In all panels of Figure (3.8), we have squared in red (prediction/performances) couples relating to target number one (upper left panel in Figure 3.7) and circled in blue those relating to target nine (lower right panel):

- All predictors are in agreement and would select source number two (upper right panel in Figure (3.6) to classify target number one. This is indeed the right choice (compare source two's decision boundary to the ideal one for target one).
- All predictors take low values in the case of target nine; also, whatever the chosen source, performances will be poor for this target, as the positive class shifts a lot compared to each source. **This may be used to predict and potentially reject targets for which performances will be low, no matter the considered detector.**

In Table (3.2) we have counted for each predictor how many times it chooses the best, second best, second worst and worst source, for each of the nine target datasets. As can be seen, selection works well, especially using the observed correlation predictors. However, at least for two of the nine targets, the worst source detector is selected. This happens systematically for target nine (which is not surprising, given that no source would achieve good performances on this target) and for target three. This latter case is a bit more complicated: source two would perform very well (94.23% agreement) while other sources result in lower, yet acceptable, performances (around 75% agreement).
This is partially explained by 1) the similar distribution profiles between source four and target three and 2) the prior probabilities of both considered distributions. To be also noted that the best source (number 2) would be selected second in that case. This is an argument in favour of multi-source decision, as previously proposed as an alternative to only selecting the best source according to our predictors.

Summary

In this Chapter we have introduced and presented our main methodological contributions: a first method of semi-supervised transfer learning (QLTL); KCA, an extension of KTA to measure the similarity between a source and a target; two direct and simple methods of transferring knowledge using alignment, that is KCATL and KTATL; an alternate scheme gradient algorithm to maximise KCA w.r.t. both source and target centres; and finally, a general method of training source selection based on alignment or alignment-derived measures.
In the upcoming Chapter, we focus on the application of these methods to sleep data classification, on REM versus NREM detection specifically. In Chapter (4) we present miscellaneous research and data analysis in order to better understand sleep data and the current system limitations; thereafter, in Chapter (5) we focus on a few validation recordings on which we will study the impact of our transfer methods, and try to understand in which case it works, in which not, and why. Finally, in Chapter (6) we present performance evolution on the whole validation dataset, for each main transfer method along with the results of training source selection.

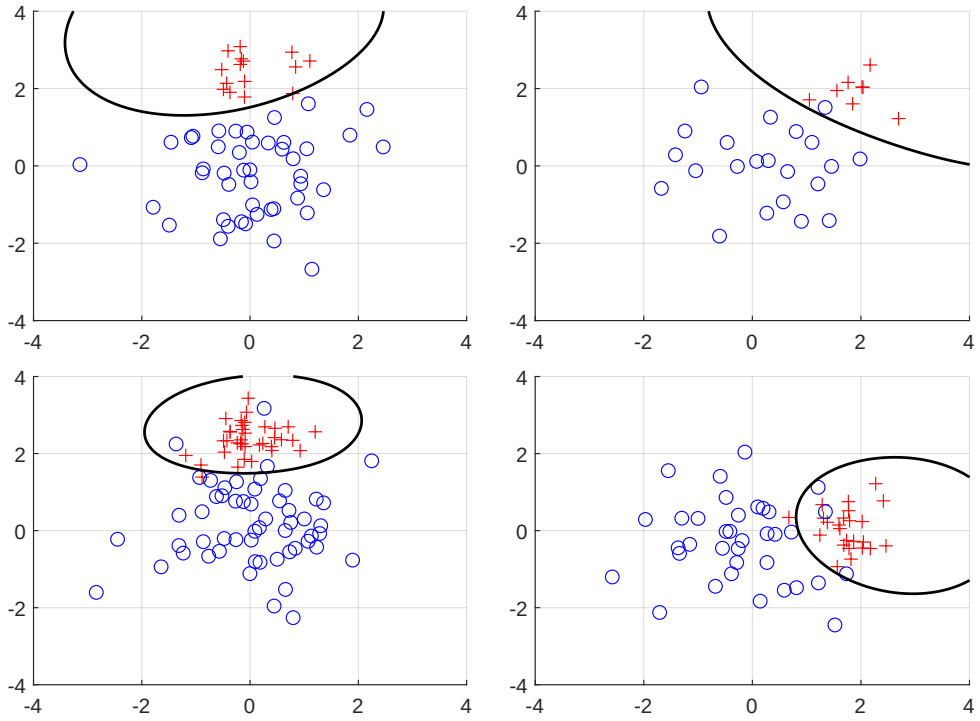


Figure 3.6: A set of four source datasets; blue circles and red pluses represent the negative and positive class, respectively. The bold black line is the decision boundary obtained through training.

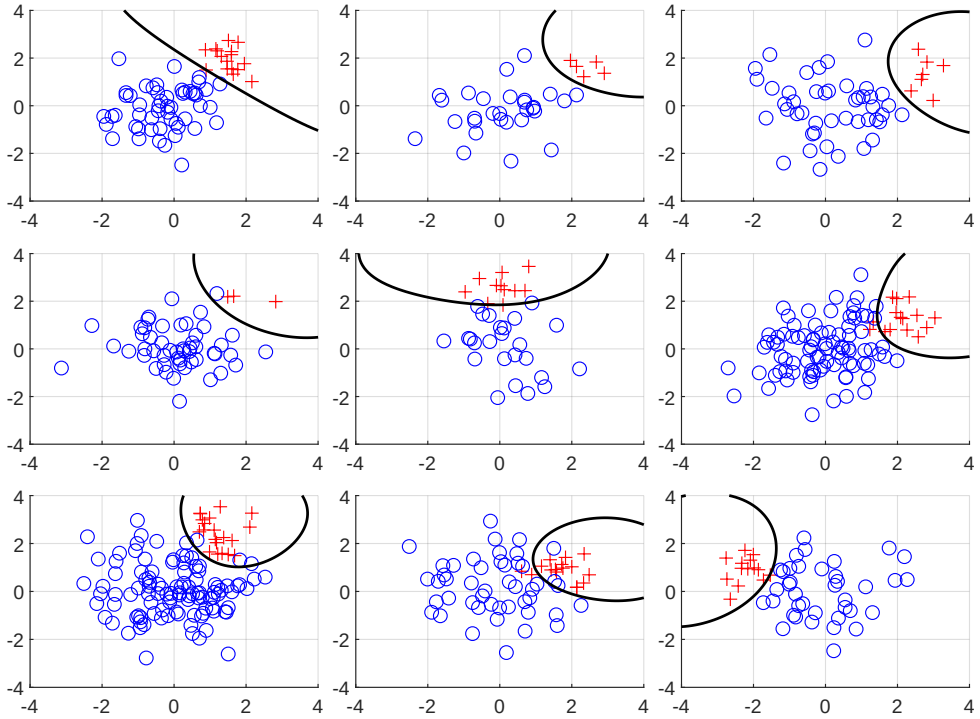


Figure 3.7: A set of nine target datasets; blue circles and red pluses represent the negative and positive class, respectively. The bold black line represents the *ideal* decision boundary. We want to choose for each target dataset the source in Figure (3.6) that results in the best classification performances.

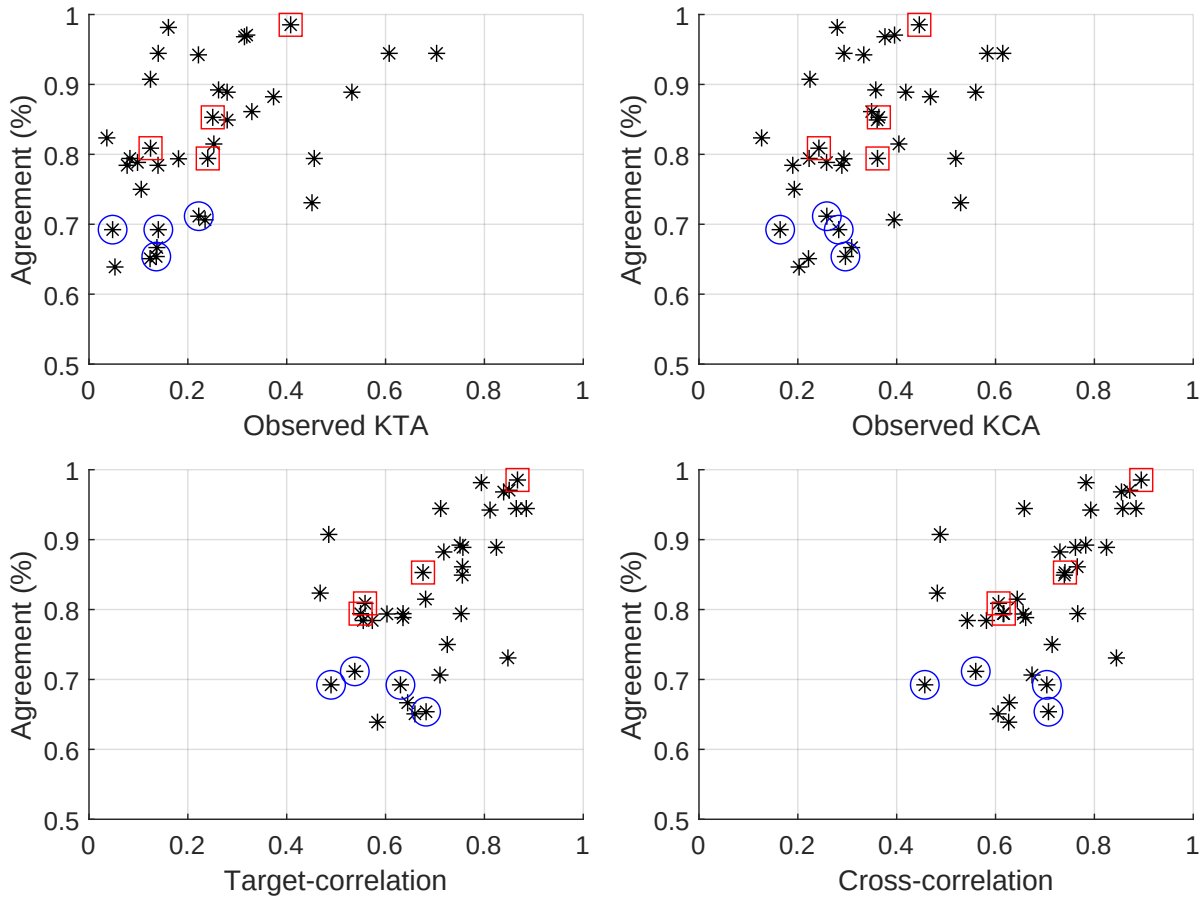


Figure 3.8: Effective training source selection: x-axis, given predictor; y-axis, actual performances. Red squares: upper left target (in Figure 3.7) predictions for each of the four sources; blue circles: lower right target predictions.

Chapter 4

Data analysis through kernels

Contents

In this chapter, we analyse sleep data mainly through the scope of kernel matrices, as they are central to our methodological contributions (both in transfer learning and training source selection). This chapter is organised as follows: in Section (4.1) we remind the reader of kernel matrices properties and hyper-parameters, so to optimally understand upcoming data analyses. Thereafter, Section (4.2) decomposes in Subsections (4.2.1, 4.2.2 and 4.2.3), in which we introduce certain challenges of REM classification, respectively REM resemblance to Wakefulness, the 1st REM period detection issue, and effects of prior shift. We then give additional comments on the set of features limitations for some nights. Finally, in Section (4.3), additional proofs of data shift are presented, which will serve as transition to Chapters (5 and 6) where we try our methodological contributions on the application.

4.1 Kernel matrix interpretation and properties

In the following, we consider kernel matrices computed within source recordings and/or within target recordings, using a Gaussian kernel of parameter $\sigma = 2$. Such choice is motivated by the Gaussian kernel's good properties, and the value for σ ensures adequate representation, for which REM/NREM alternating pattern is easily visible. We illustrate this latter remark below.
In this subsection in particular, we use the same recording for all figures and representations, as the sleep pattern is quite standard.

4.1.1 Choice of the kernel parameter(s)

In this document and in our research, we consider the standard Gaussian kernel to compute our kernel matrices; such kernel is parametrised by σ, which affects the dynamic of the Gram matrix, ranging from:

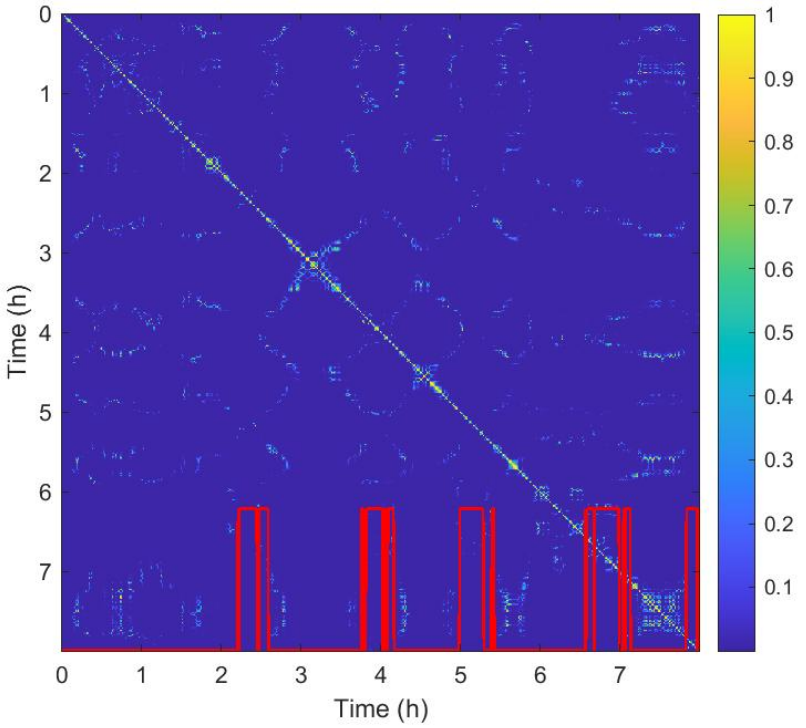


Figure 4.1: Gaussian Gram matrix for $\sigma = 0.2$, which illustrates a situation where σ is too low and the resulting dynamic is too sharp. The red line is REM/NREM ground-truth.

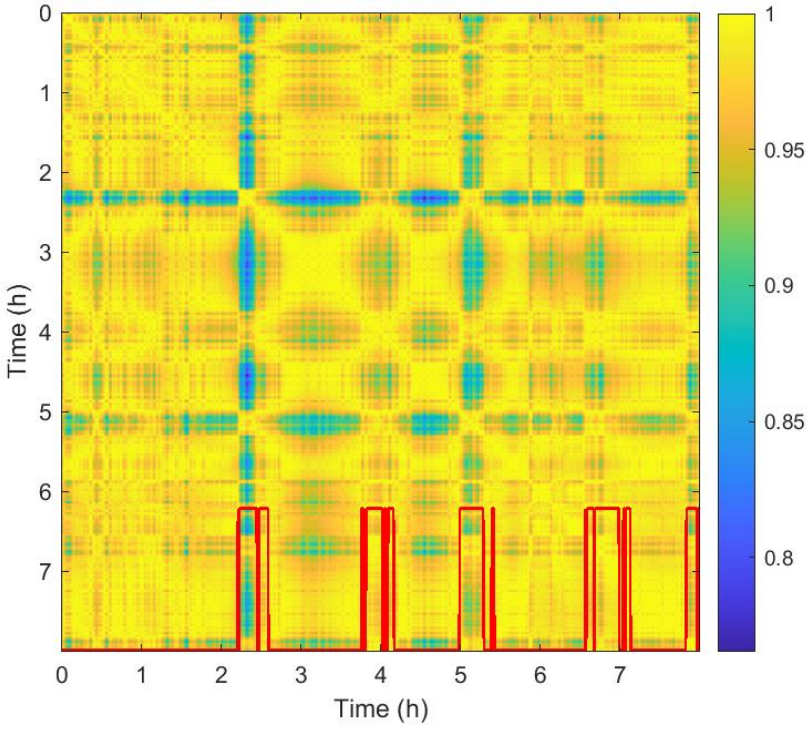


Figure 4.2: Same representation as Figure (4.1) for $\sigma = 10$, which is too high and results in a dynamic too smooth.

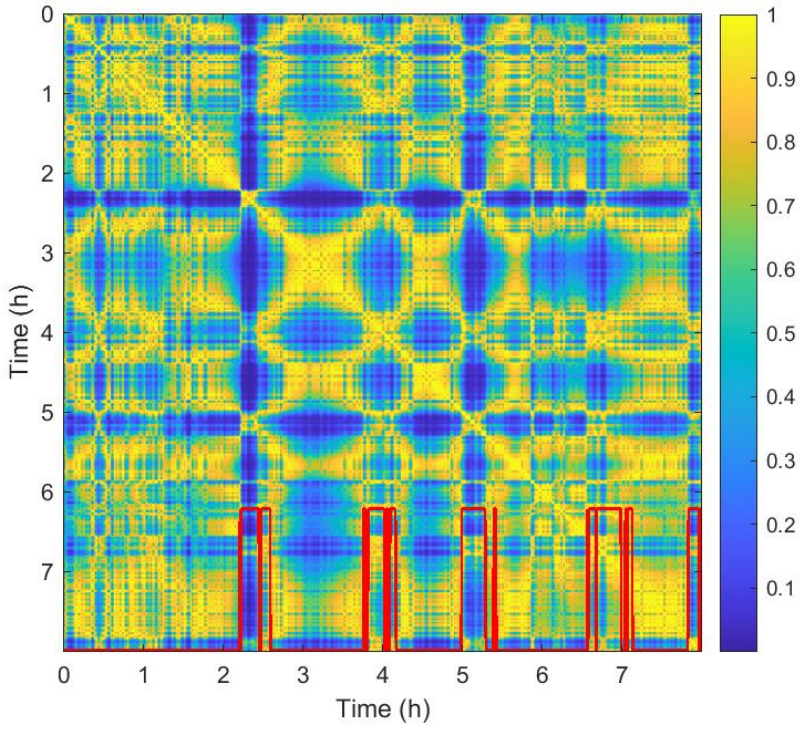


Figure 4.3: Same representation as Figures (4.1 and 4.2), now with $\sigma = 2$, which illustrates a situation where σ is adequate and results in a dynamic sharp enough to discriminate REM from NREM, and also smooth enough not to be sensitive to small changes in the signal.

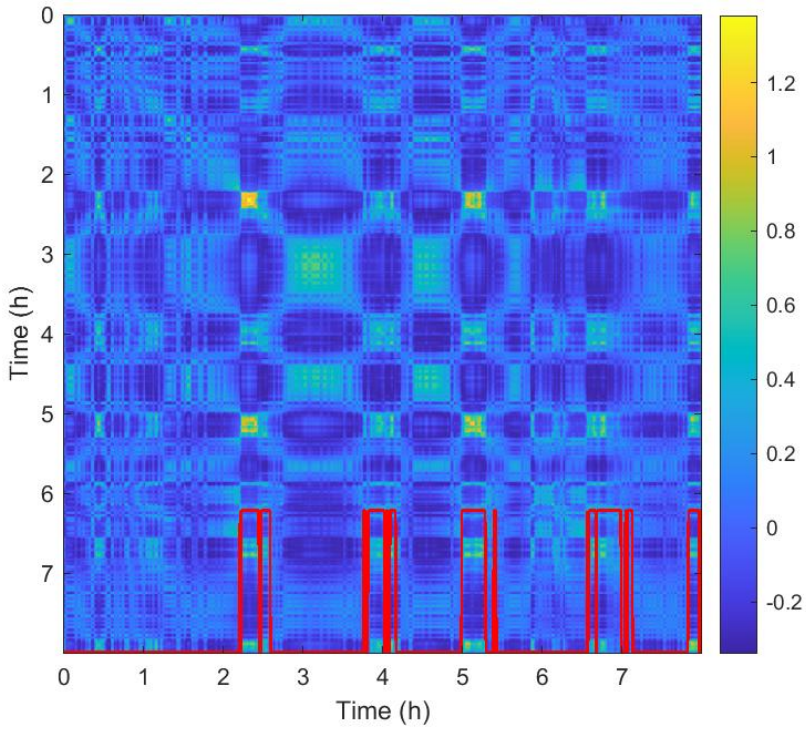


Figure 4.4: Same representation as Figure (4.3) using this time the *mean* centred Gram matrix. Values may now be negative, which may also be interesting when measuring similarities to the decision given by a detector in coding ± 1.

- The all-ones matrix $\mathcal{J}$ for $\sigma \to +\infty$.
- The identity matrix $\mathcal{I}$ for $\sigma \to 0$.

This means that greater values of σ give a smoother dynamic while lower values give a *sharper* one. We illustrate this behaviour in Figures (4.1 to 4.3).
Other kernel values different than zero only appear for couples of identical observations. Also, in the case of the cross-Gram matrix K_{TS}, it is almost identical to the all-ones matrix for $\sigma \to +\infty$ and the all-zeroes matrix for $\sigma \to 0$ as the probability to find two identical observations independently drawn from continuous distribution(s) equals zero.
In the end, the selection of a good value for σ is determined by both the machine (or transfer) learning technique, and by the dimensionality of the feature vectors used for class-discrimination. From this point on and while not specified otherwise, we use $\sigma = 2$ to compute our kernel matrices used for representations.

Centring of the kernel matrix

Most techniques of centring presented so far are based on some knowledge (or estimation) of the labels. We mainly use *mean* centring for our methods, and give visual comparison of the mean-centred kernel matrix to the non-centred one in Figures (4.3 and 4.4).

4.1.2 Illustration of the discriminative information contained in Gram matrices

In the case of a square Gram matrix, computed *within* a set of observations, it can easily be decomposed as:

$$\begin{aligned} K_{TT} &= V\Sigma V' = \sum_{i=1}^{N} \lambda_i V_i V_i' \\ K_{TT_C} &= V_C \Sigma_C V_C' = \sum_{i=1} \lambda_{C_i} V_{C_i} V_{C_i}' \end{aligned} \tag{4.1}$$

where Σ is the diagonal matrix containing K_{TT} eigenvalues: $\lambda_i, \forall i \in \{1,\ N\}$; V contains their associated orthogonal eigenvectors $V_i, \forall i \in \{1,\ N\}$. For the sake of simplicity, we suppose that eigenvalues are sorted from largest to smallest: $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_N \geq 0$.
V_C, Σ_C, $(\lambda_{C_i}, V_{C_i}), \forall i \in \{1,\ N\}$ denote their counterparts when decomposing the centred target Gram matrix K_{TT_C}. We illustrate the interest of such a decomposition in Figures (4.5 to 4.8), using centred target kernel matrices:

- Figures (4.5,4.6 and 4.7) illustrate the eigenvectors associated with the three greatest eigenvalues $\lambda_{C_1}, \lambda_{C_2}$ and λ_{C_3}, respectively. As can be seen for this recording, V_{C_1} **and** V_{C_2} **are highly discriminative of the REM stage**. We remind that this is the whole principle behind QLTL and KTATL: to **take advantage of the unsupervised information** contained in the kernel matrix (as was illustrated on a toy example in Figure (3.1)). This constitutes a clear and strong example of this idea, on the application. However, eigenvector V_3 is not REM discriminative at all.
- In Figure (4.8), we partially reconstruct the Gram matrix using the *first* three eigenvalues and eigenvectors:

$$K_{TT}|_{RECON_3} = \sum_{i=1}^{3} \lambda_i V_i V_i' \tag{4.2}$$

  which, in this example, explains 74.7% of the variance. This idea may be used for de-noising, which might be useful to improve our transfer methods and/or our selection methods. This will be considered in the operational setup further in the document.

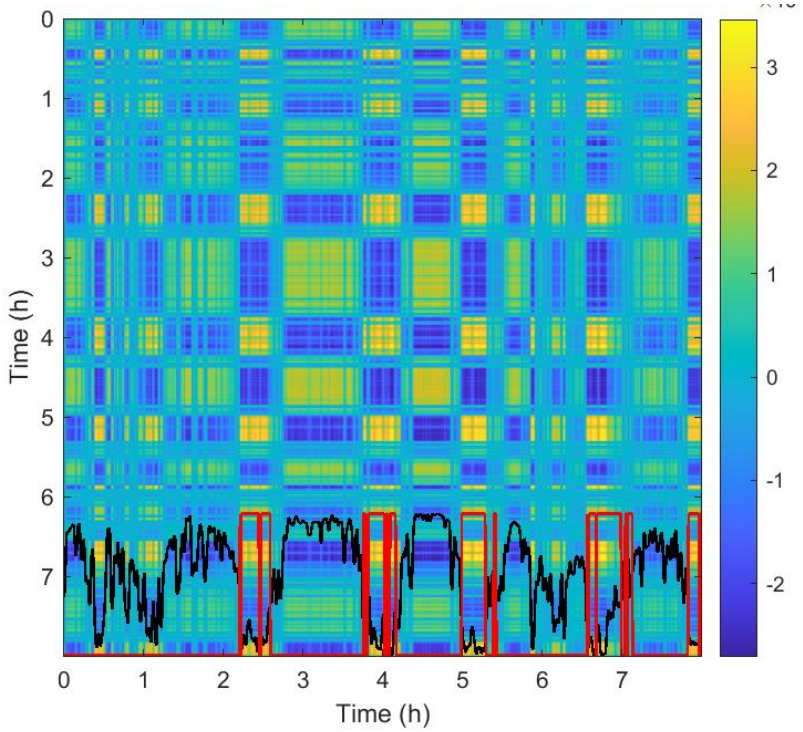


Figure 4.5: In this figure is illustrated the eigenvector V_{C_1} of K_{TT_C} associated to its greatest eigenvalue λ_1. The matrix in colour-scale represents $V_{C_1}V'_{C_1}$, while V_{C_1} is the plotted black line. The red line represents here again the REM/NREM ground-truth. λ_{C_1} explains 40.8% of the variance, and there is a correlation of -0.66 between V_1 and the Ground-truth.

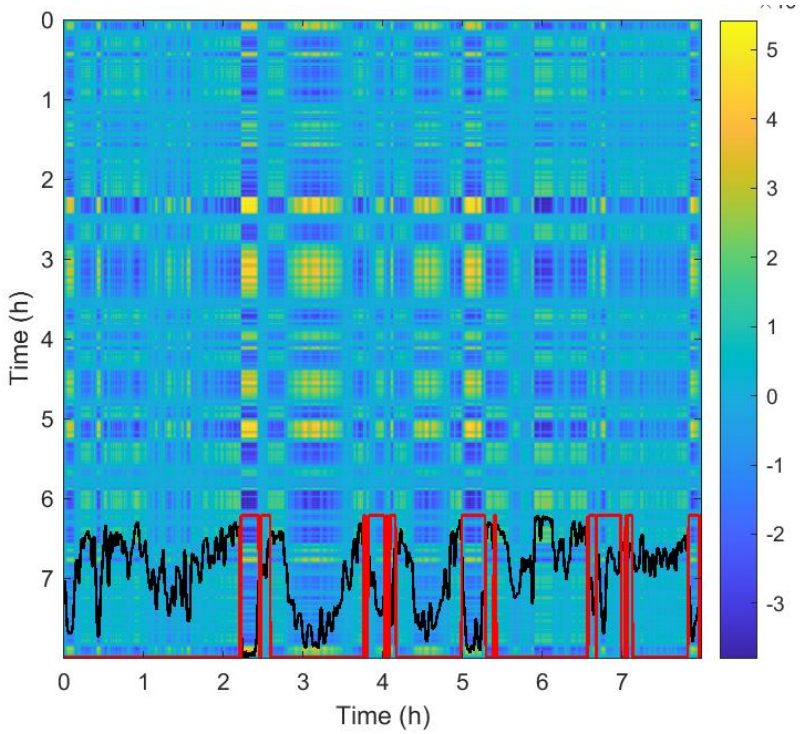


Figure 4.6: Same representation as Figure (4.5) for λ_{C_2} and V_{C_2}, the second greatest eigenvalue and associated eigenvector. λ_{C_2} explains 20.2% of the variance, and the correlation between V_2 and the Ground-truth is equal to -0.33.

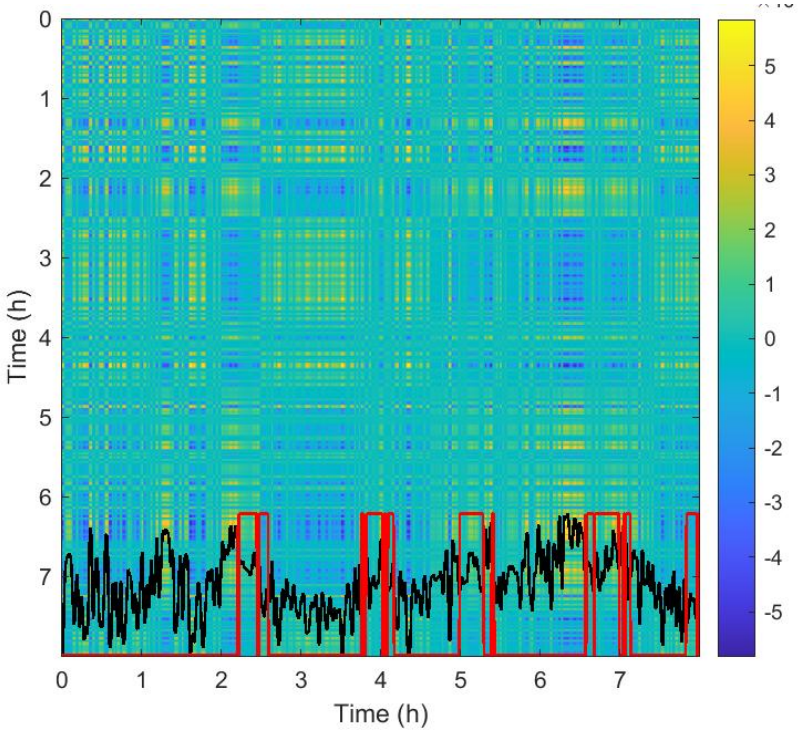


Figure 4.7: Same representation as Figures (4.5 and 4.6) for λ_{C_3}, representing 13.7% of the variance. For this third eigenvector, the correlation with REM/NREM equals 0.14 only: it yields almost no information to discriminate REM from NREM.

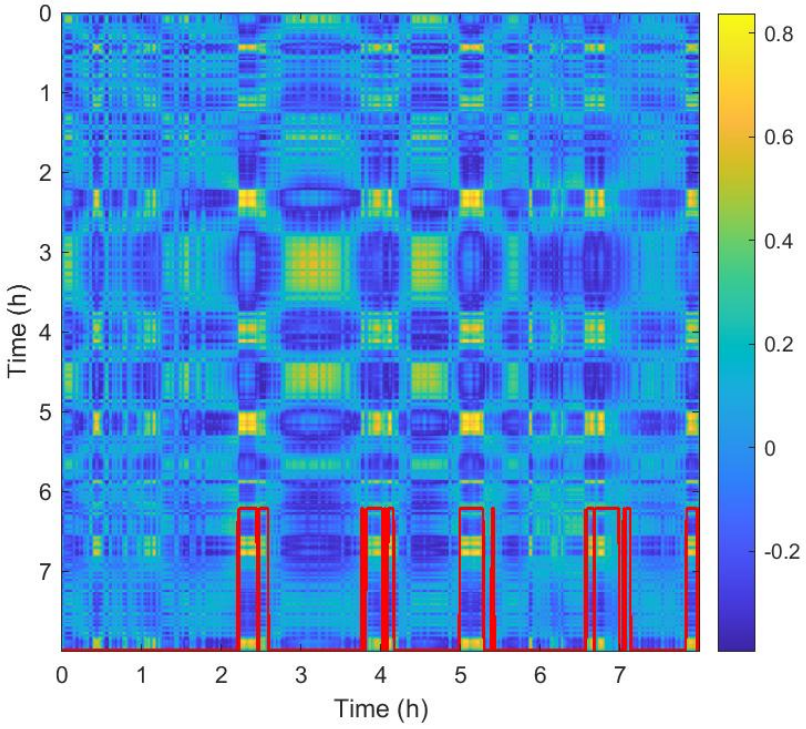


Figure 4.8: Reconstruction of the centred Gram matrix in Figure (4.4) using its three greatest eigenvalues and associated eigenvectors (74.7% of the variance).

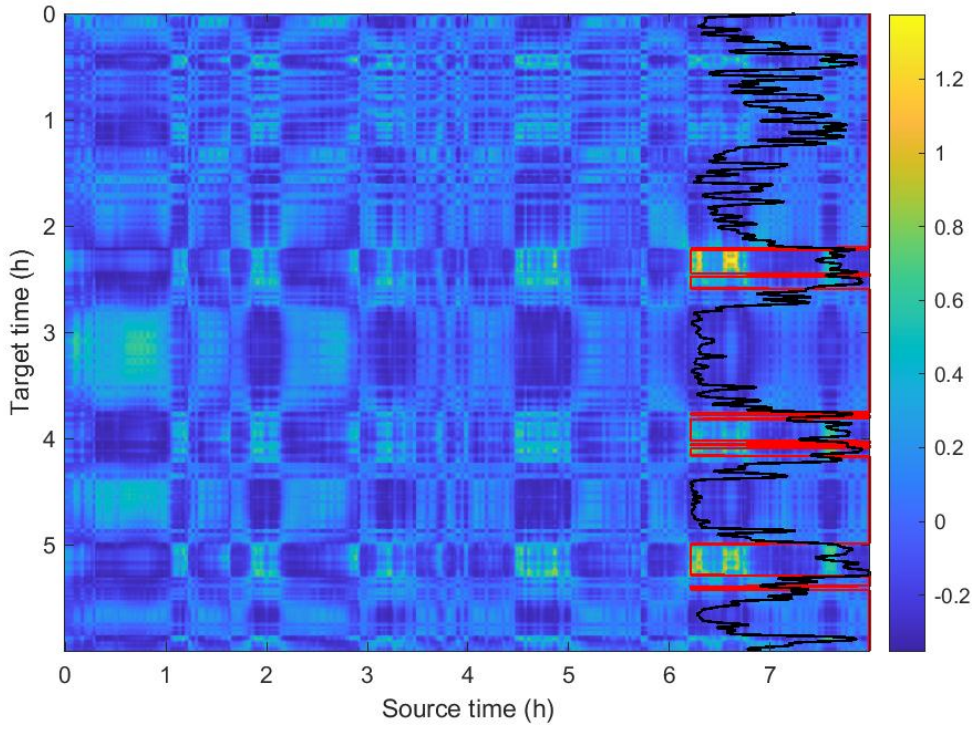


Figure 4.9: Illustration of K_{TS_C} and U_{C_1} the left-singular vector of U_C associated to the greatest singular values (black line) along with **target** REM/NREM ground-truth (red line). There is a correlation of -0.62 between the red and the black line.

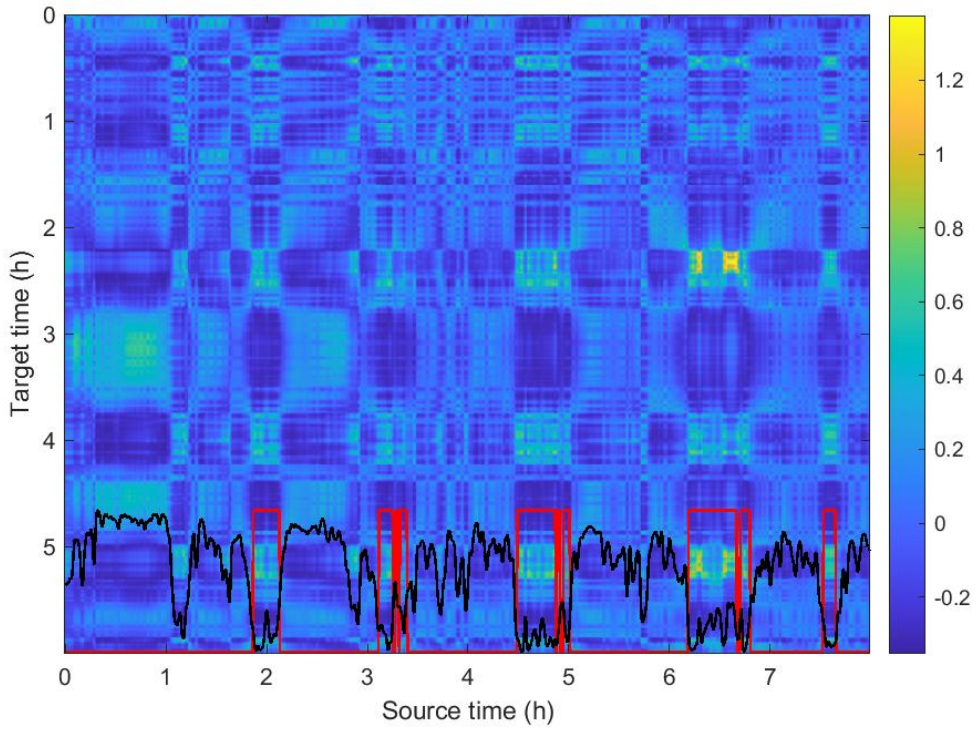


Figure 4.10: Illustration of K_{TS_C} and V_{C_1} the right-singular vector of V_C associated to the greatest singular value (black line) along with **source** REM/NREM ground-truth (red line). There is a correlation of -0.76 between the red and the black line.

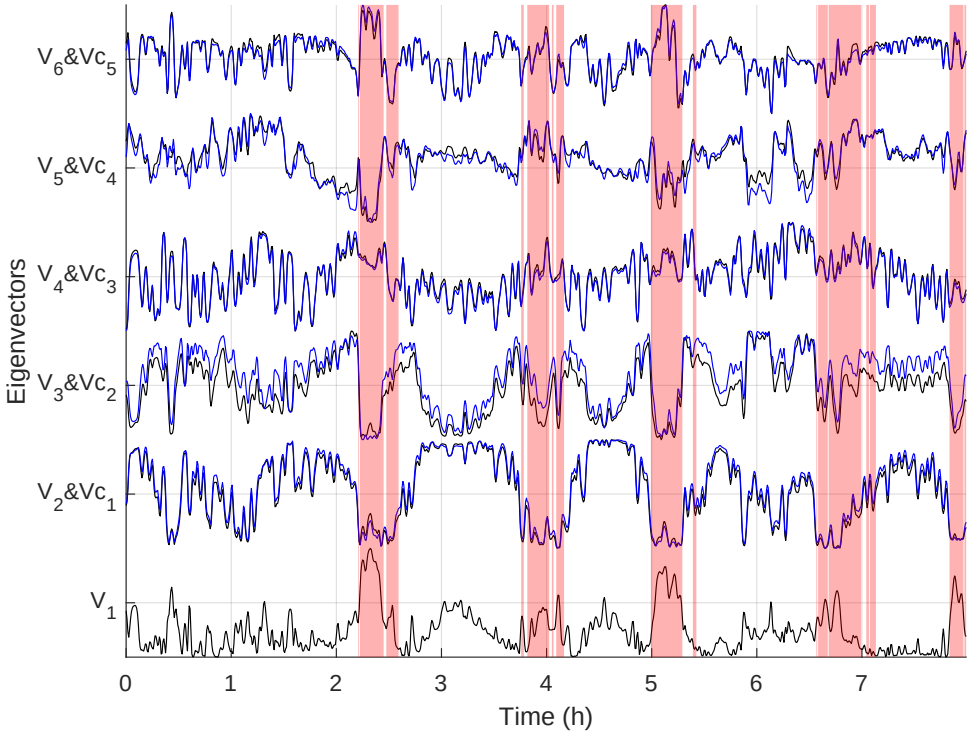


Figure 4.11: Illustration of the effect of mean-centring on K_{TT} eigenvectors, which become shifted: the second eigenvector of K_{TT} is similar to the first of K_{TT_C}, the second to the third, and so on. This property is studied in [56]; red areas correspond to REM in the corresponding target ground-truth.

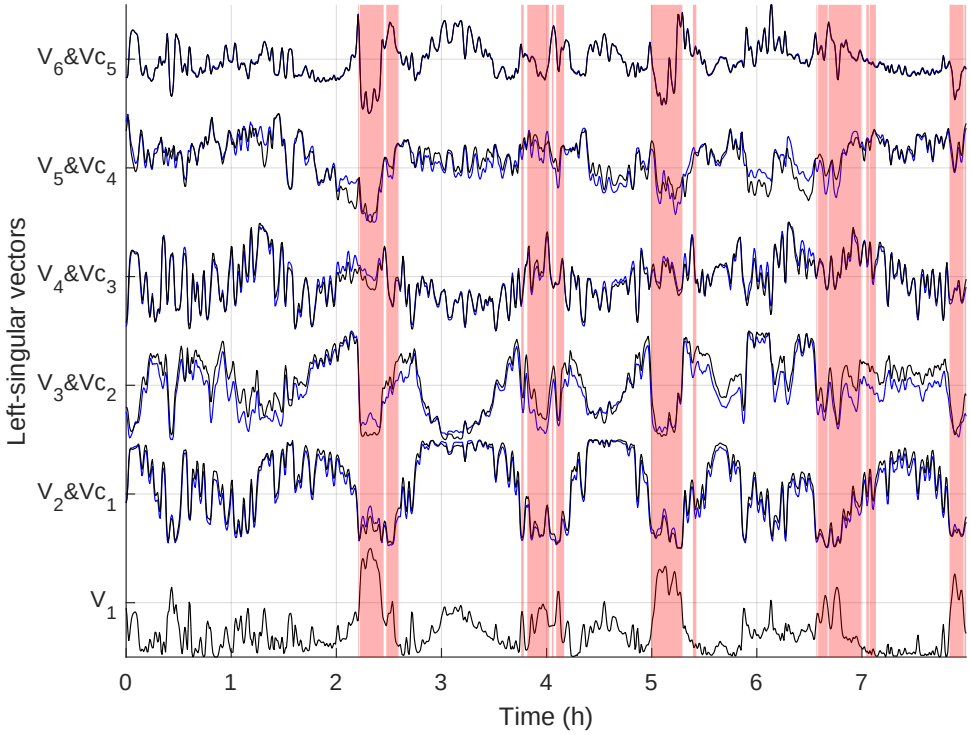


Figure 4.12: Illustration of the effect of mean-centring on K_{TS} left-singular vectors, which become shifted in the same way as eigenvectors in Figure (4.11). Representations here relate to the same target as Figure (4.11), and red areas correspond to target ground-truth REM.

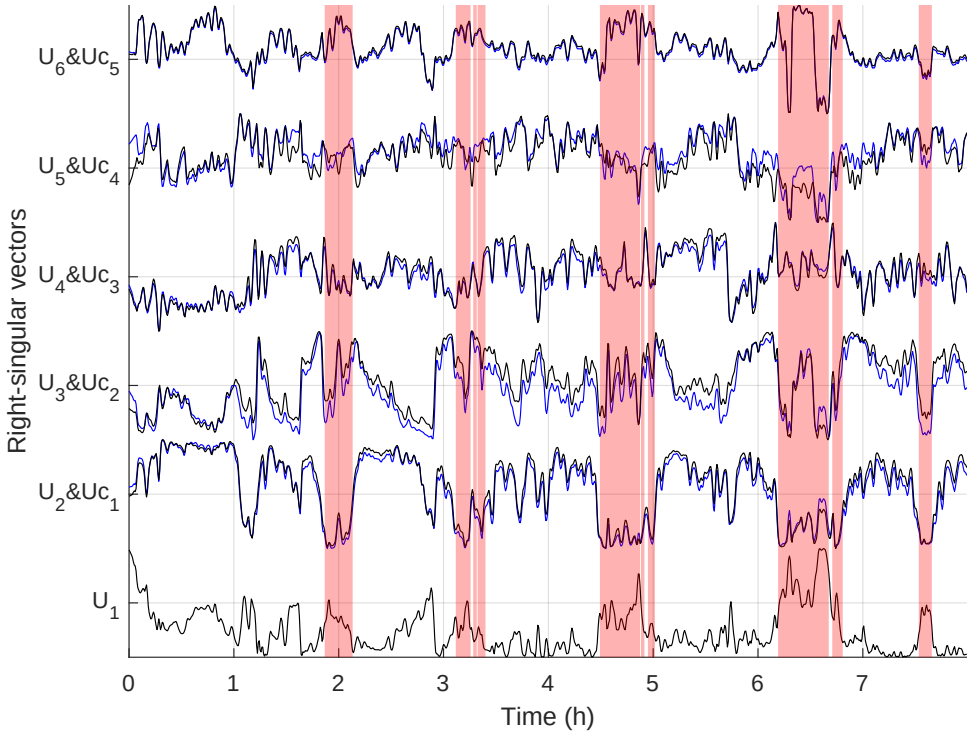


Figure 4.13: Illustration of the effect of mean-centring on K_{TS} right-singular vectors, which become shifted in the same way as eigenvectors in Figures (4.11 and 4.12). Representations here relate to the source, red areas corresponding to source ground-truth REM.

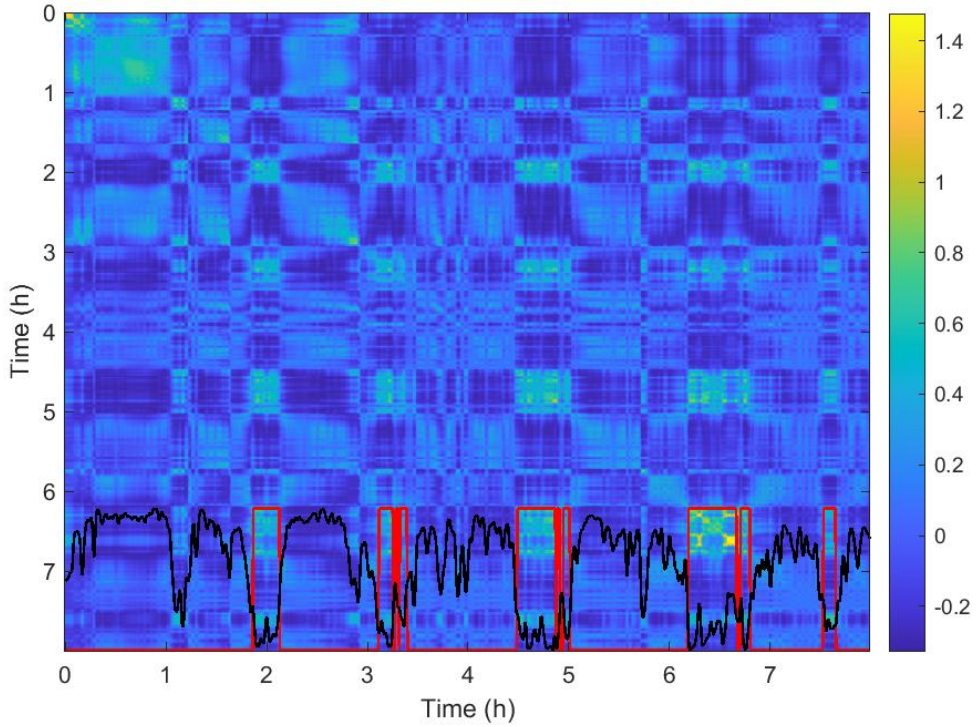


Figure 4.14: Illustration of K_{SS_C} the centred source kernel matrix computed using the same source as in Figures (4.9 and 4.10). The black line is the plot of K_{SS_C} eigenvector associated to its greatest eigenvalue; it achieves a correlation of -0.80 with the source REM Ground-truth. As can be seen, it is almost identical to K_{TS} second - or K_{TS_C} first - right-singular eigenvector, as displayed in Figure (4.13).

4.1.3 Decomposition in the case of cross-kernel matrices

In the case of cross-Gram matrices, they can be seen through the scope of Singular Value Decomposition (SVD), which can be seen as a generalisation of the eigendecomposition:

$$\begin{aligned} K_{TS} &= V\Sigma U' = \sum_i \lambda_i V_i U_i' \\ K_{TS_C} &= V_C \Sigma_C U_C' = \sum_i \lambda_{C_i} V_{C_i} U_{C_i}' \end{aligned} \tag{4.3}$$

where V and U are orthogonal matrices of sizes $N_T \times N_T$ and $N_S \times N_S$, respectively, N_T being the number of target observations and N_S the number of source ones; and where Σ is a $N_T \times N_S$ matrix containing the singular values of K_{TS}.
We denote by $(V_i,\ U_i)$ the vectors composing V and U, respectively, and corresponding to the i^{st} column in both cases. These are often referred to as the left-singular and right-singular vectors of K_{TS}, respectively.
Here again, V_C, U_C, Σ_C, $[\cdots]$ denote the counterparts of V, U, Σ, $[\cdots]$ for K_{TS_C} the centred cross-Gram matrix. In the remainder of the document, and when not specified otherwise, we always consider the mean-centred Gram matrices.
In Figures (4.9 and 4.10), we represent the left-singular and right-singular vectors associated to the greatest singular value, which correspond to the target and to the source, respectively. As can be seen, these correlate well with the REM/NREM ground-truths, with correlations of -0.62 for the target and -0.76 for the source, respectively. Such correlation between K_{TS} singular vectors and the ground-truth justifies our interest in using cross-Gram matrices as tools of transfer learning and source selection.

4.1.4 Effects of mean centring

Comparison between mean centred target kernel matrix K_{TT_C} and non-centred target kernel matrix K_{TT} is introduced, developed and illustrated extensively in [56]. Among interesting properties:

- Eigenvalues of a non-centred kernel matrix and a centred one are interlaced, that is:

$$\lambda_1 \geq \lambda_{C_1} \geq \lambda_2 \geq \lambda_{C_2} \geq \lambda_3 \geq \lambda_{C_3} \cdots \tag{4.4}$$

  where λ_* denote K_{TT} eigenvalues and λ_{C_*} those of K_{TT_C} - their mean-centred counterparts. We have checked this property on our data and illustrate it on our target example: eigenvalues are displayed in Table (4.1).

- Shifted eigenvectors are highly correlated: $\forall i \in \{1,\ ...,\ N\}$, $V_{C_i} \approx V_{i+1}$. This is once again illustrated in Figure (4.11).

As we are also using cross-Gram matrices, we numerically checked if such properties hold for these matrices. This seems to be indeed the case, as illustrated in Table (4.2) and Figures (4.12 and 4.13):

- Singular values of K_{TS_C} and K_{TS} are interlaced.
- Left-singular vectors highly correlate between decomposition of K_{TS_C} and K_{TS}: $\forall i \in \{1,\ ...,\ N\}$, $V_{C_i} \approx V_{i+1}$, as illustrated in Figure (4.12). So do right-singular vectors (see Figure (4.13)).
- As we used the same target recording to compute K_{TT_C} and K_{TS_C}, we also observe a high correlation between K_{TS_C} left-singular values and K_{TT_C} eigenvectors (corresponding vectors between Figure (4.11) and Figure (4.12)).
- This is also the case if we consider K_{SS_C} the source-Gram matrix computed using the source dataset which was used to compute K_{TS_C}. We illustrate K_{SS_C} in Figure (4.14), along with the eigenvector associated to K_{SS_C} greatest eigenvalue. It is also almost identical to U_2/U_{C_1} in Figure (4.13).

However, as we currently have no practical applications of these properties for our methods, we did not extend the results of [56] to the case of cross-Gram matrices yet. We focused on the improvement of our methods through other means.

$N =$	1	2	3	4	5
λ_N	592.88	157.12	67.01	53.64	25.33
λ_{CN-1}		160.27	79.58	53.88	32.24

Table 4.1: Interlacing of K_{TT} and K_{TT_C} eigenvalues, as described in [56].

$N =$	1	2	3	4	5
λ_N	579.96	143.25	68.16	46.15	25.67
λ_{CN-1}		148.04	77.93	46.59	36.14

Table 4.2: Interlacing of K_{TS} and K_{TS_C} singular values.

Additional representation

In Figures (4.15 and 4.16), we further illustrate the shift between eigenvectors of K_{TT} and K_{TT_C}, using the decompositions:

$$\begin{aligned} K_{TT} &= VDV' = V\sqrt{D}(V\sqrt{D})' = \Phi\Phi' \\ K_{TT_c} &= V_cD_cV_c' = V_c\sqrt{D_c}(V_c\sqrt{D_c})' = \Phi_c\Phi_c' \end{aligned} \tag{4.5}$$

where $\sqrt{D}$ and $\sqrt{D}_c$ are the diagonal matrices composed of the square roots of K_{TT} and K_{TT_C} eigenvalues, respectively.
Φ and Φ_C can be seen as the coordinate matrices in the RKHS up to an orthogonal transform. We represent in 3D the three first vectors of the centred matrix in Figure (4.15) and vectors "two" to "four" of the non-centred one in Figure (4.16). As can be seen, this results in highly similar representations.

4.2 Observations on REM detection

In the first Section of this Chapter, we developed on our use of kernel matrices to analyse our recordings. In this second Section, we mainly use these matrices to present a few recurring issues that may complicate REM detection; Figures (4.17 to 4.22) all represent the target-kernel matrices of a few example recordings, mean-centred and for $\sigma = 2$; ground-truth REM and a REM detector output are both represented as a red and a black line, respectively. Ground-truth Wakefulness is also represented by a yellow line, when relevant.

4.2.1 REM resemblance to Wakefulness

In traditional sleep staging, the REM sleep stage is also referred to as *paradoxical* sleep, as it presents many of the characteristics of wakefulness in the brain waves pattern captured by the EEG, while also corresponding to an absence of muscular activity in the EMG.
This similarity between Wakefulness and REM also appears in our HR-based signals, and is clearly identified in the literature. We illustrate such similarity in Figures (4.17 and 4.18):

- The first illustration corresponds to a typically-good recording, with clear sleep cycles of $\approx 2h$ duration each, and only two short wake-ups after sleep onset. The subject has a sleep latency of a little more than half an hour; this first half hour of Wakefulness resembles a lot to each of the REM phases occurring through the night. This is also captured by the trained REM detector (black line) used as example, which detects this first half hour as if it was REM.

- The second figure illustrates a more problematic case, where one may observe many short wake-ups after sleep onset. Once again, there is a high similarity in the Gram matrix patterns between REM and Wakefulness, but it is also the case in the time period $\in [1h10;\ 1h40]$ which unfortunately corresponds to NREM. This will result in false-alarm and is most-likely due to the two wake-ups around this period.

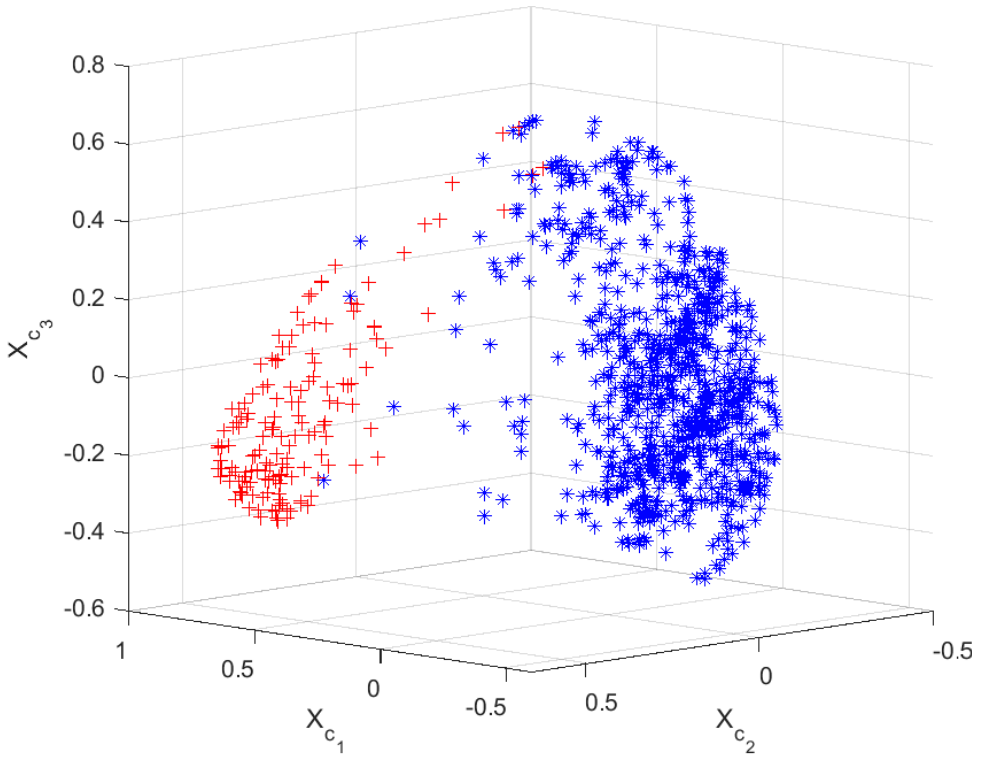


Figure 4.15: Partial representation of sleep data in the RKHS: three first coordinates in the case of the centred cross-Gram matrix K_{TT_c}. Red crosses correspond to REM observations and blue asterisks to NREM.

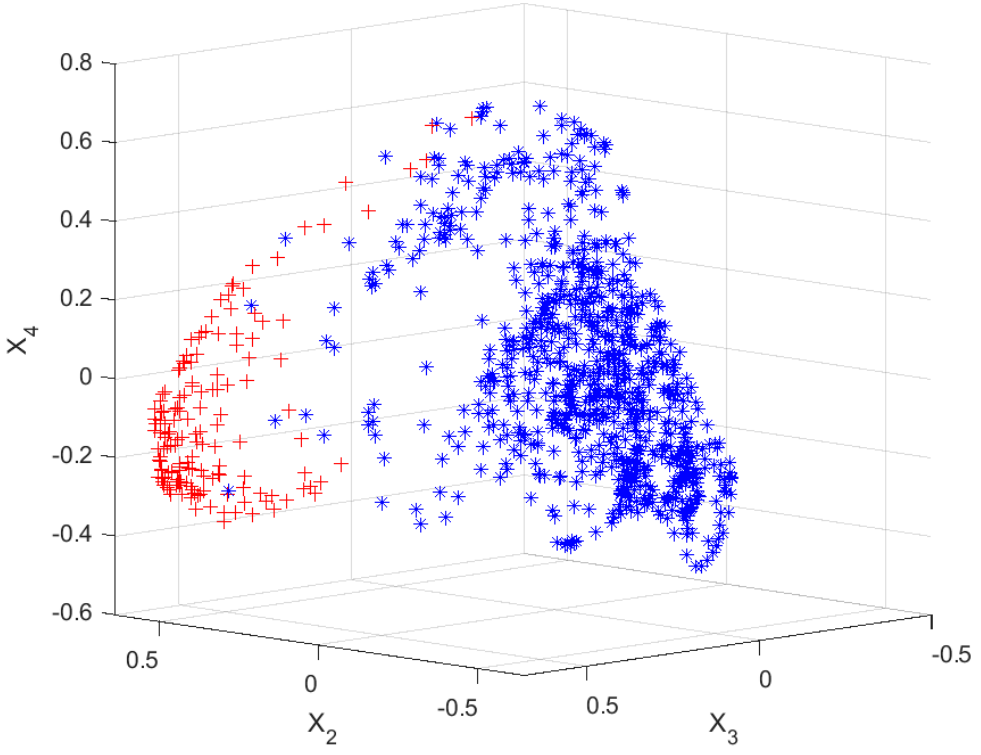


Figure 4.16: Partial representation of sleep data in the RKHS: second, third and fourth coordinates in the case of the non-centred cross-Gram matrix K_{TT}. Red crosses correspond to REM observations and blue asterisks to NREM. This leads to a highly similar representation as Figure (4.15), which demonstrates the effect of mean-centring: a shift in eigenvectors.

In both cases, there is a Wakefulness-specific detector and confusions between REM and Wakefulness are solved by comparing both detectors output. In the first example, REM detection would almost be perfect (every REM period is accurately detected and there is little-to-no false-alarm during NREM sleep), and the Wakefulness detector has priority over the REM detector, meaning that in the full system, there would be no false-alarm during Wakefulness. However, in the second case, the time period $\in [1h10;\ 1h40]$ corresponds to NREM and will be misclassified as REM, contributing to false-alarm. We ought to find a more adequate REM detector that would not react to such features as REM (using source selection), or to do careful decision adaptation as to reduce the activation of the REM detector in proximity of short periods of Wakefulness.

4.2.2 First REM period detection issue

The detection of the first REM period is critical for studies where REM latency matters (reminder: it is the time elapsed between sleep onset and the beginning of the first REM period). REM periods are often shorter in the beginning of a recorded night and get longer by the end of the recording. Here again we illustrate first REM period detection through two examples:

- In Figure (4.19), the first period of REM is too short (a few epochs at most) to be detected; while features and the associated detector are sensitive to this first REM, the decision statistic does not take values higher than the decision threshold, which, if lowered would result in a higher false-alarm rate. This is also partly due to filtering, which helps capture the sleep process (as observations are linked to one another through time, and are therefore non- independent).
- In Figure (4.20), we are in a similar - yet clearer - situation as in Figure (4.18): REM detector activates at time period $\in [1h10;\ 1h20]$, most-likely due to the two short awakenings (yellow line). In this example, REM at this time period would fit very well with the cyclicity of sleep.

The first example corresponds to a non-detection issue; if REM latency is critical, it may be detected through other means (rupture in signal patterns detection, among others). Sleep latency is not a robust metric on which to optimise a system; in this manuscript and for detectors optimisation, we consider the average working point on the validation dataset, which is way more robust and practical for optimising our methods.

4.2.3 REM Prior shift and class mixture

Here we consider two similar situations with different outcomes:

- In both cases (illustrated in Figure (4.21 and 4.22)), there is a lot of prior shift as the proportion of Wakefulness is high.
- In the situation displayed in Figure (4.21), the REM detector would yet achieve acceptable classification performances, as it takes higher values in REM sleep and lower values in NREM sleep.
- In the second situation, displayed in Figure (4.22), performances would be poorer as there is not only prior shift, but also an increased class-mixture.

Using the reduced set of features (allowing 3D representations and used to compute the kernel matrices in this chapter), there are situations with high-mixture between REM and NREM observations, resulting in poor detection performances. Such situations cannot be solved using transfer learning, but may probably be solved using source selection (and in future work, features selection).
(Uninformed) transfer learning will be useful in situations of covariate shift and of prior shift only.

4.3 Additional illustrations of data shift

In the previous Section of this Chapter, we presented a few observations on REM detection and of some of its challenges. In the end, it results in any combination of the three types of data shifts presented in Section (1.3.3). In this third and last Section, we give two additional proofs of data shift.

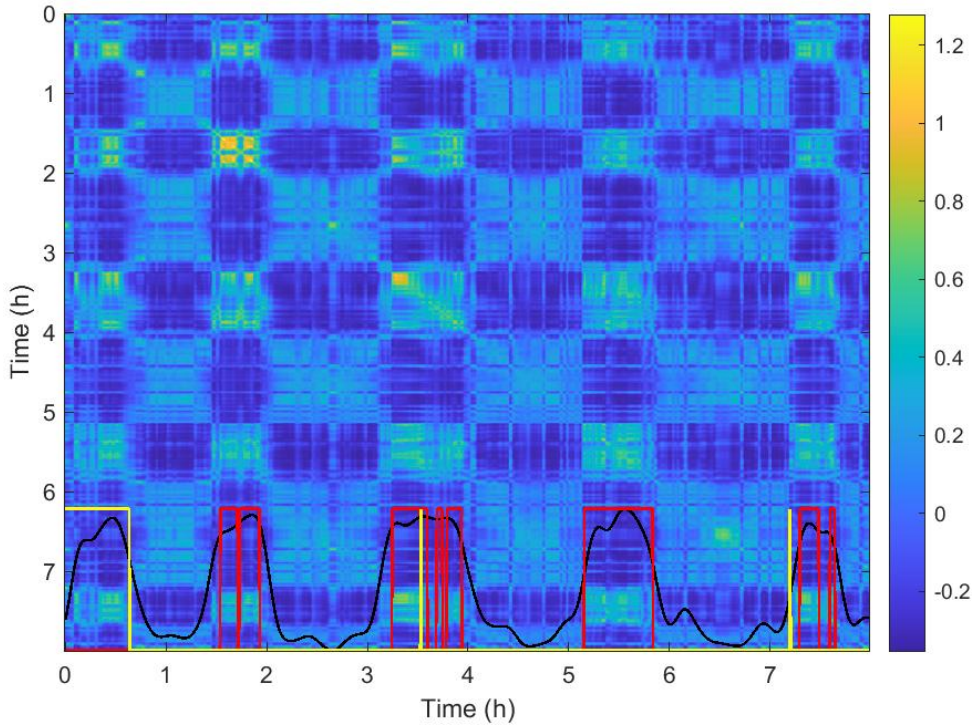


Figure 4.17: Illustration of REM resemblance to Wakefulness for the used features and corresponding Gram matrix. Red line and yellow line correspond to the ground-truth REM and Wakefulness, respectively. Black line is the output of a REM detector applied to this recording.

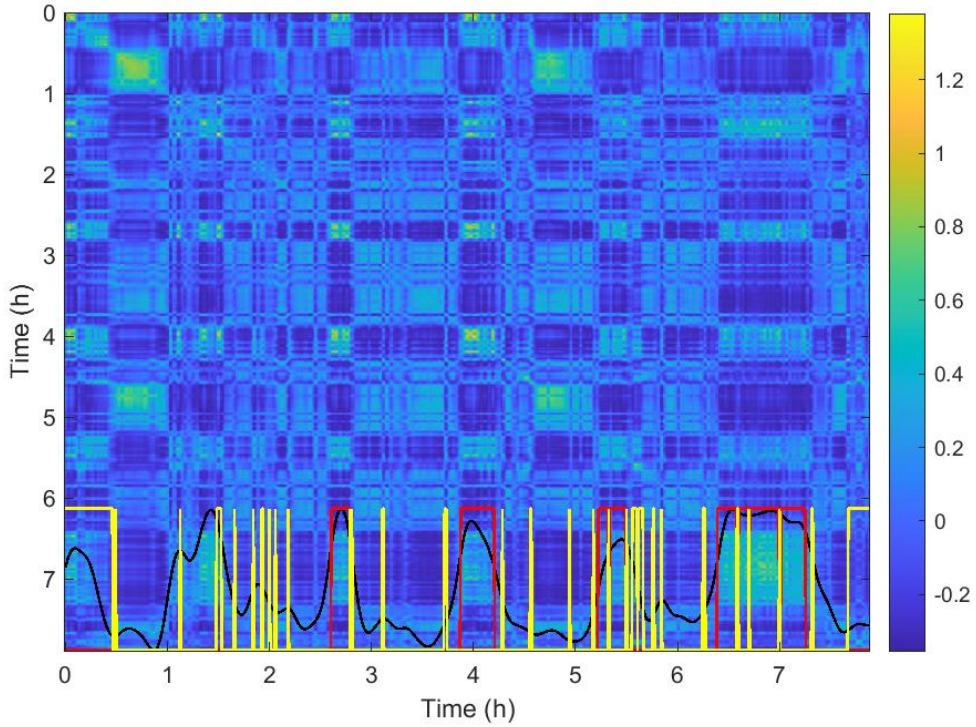


Figure 4.18: Additional illustration using the same representations as in Figure (4.17). There are more short transitions to wakefulness in this second recording, which may result in REM detection false-alarms.

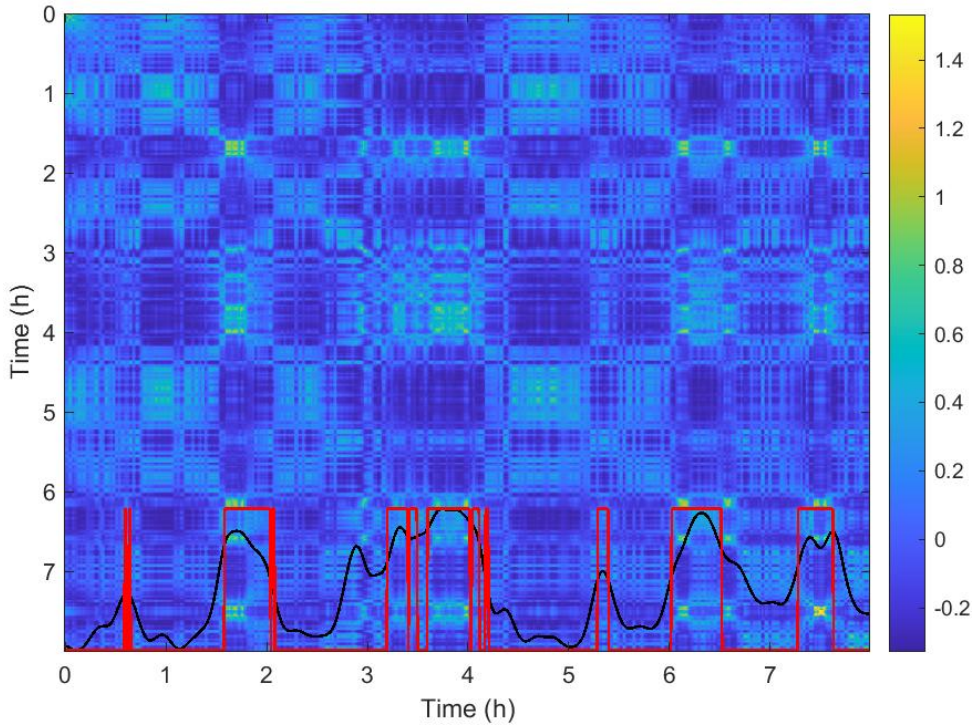


Figure 4.19: In this first example, the first REM period is too short and would be missed by the REM detector.

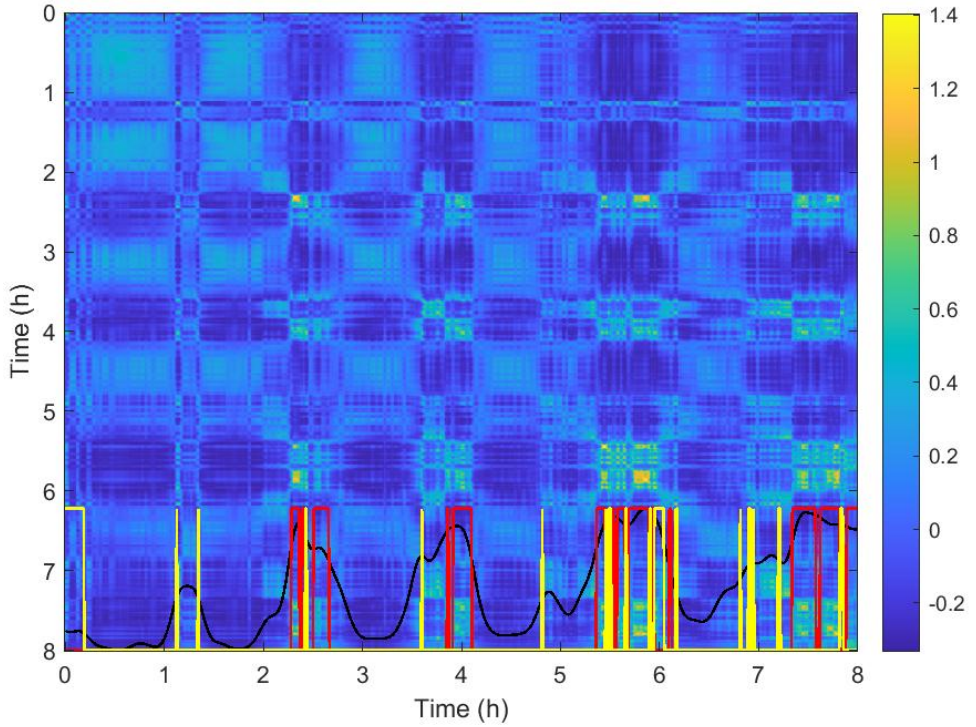


Figure 4.20: In this second example, there are two consecutive short transitions to Wakefulness just after $\approx 1h10$, which result in higher values in the detector decisions statistics. This may result in false-alarm.

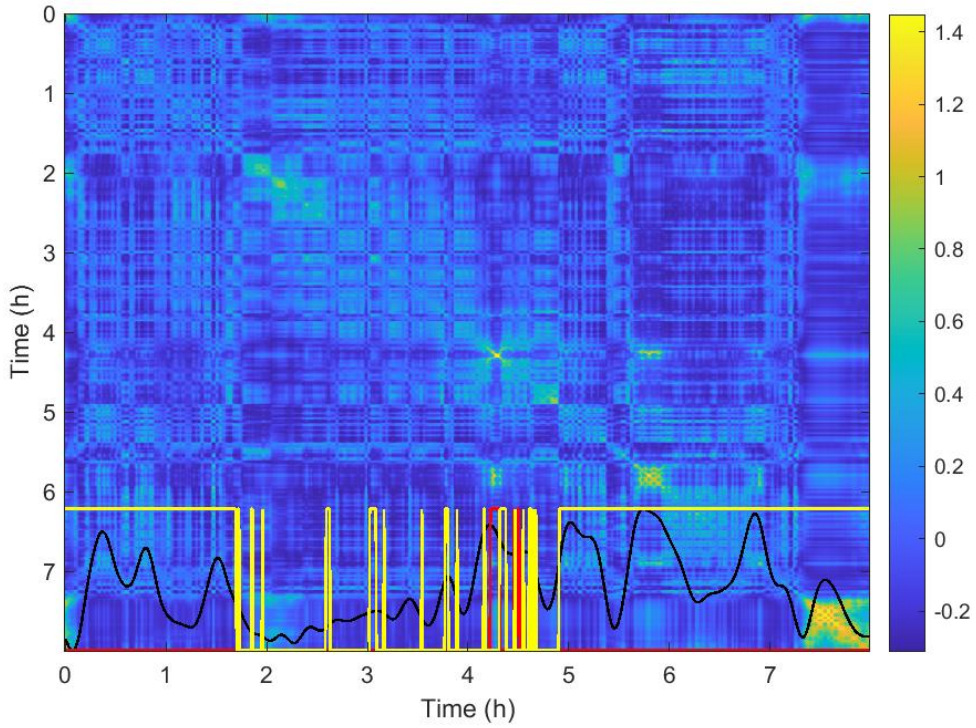


Figure 4.21: Illustration of a night with high prior shift, but where the REM detector would still achieve acceptable classification performances.

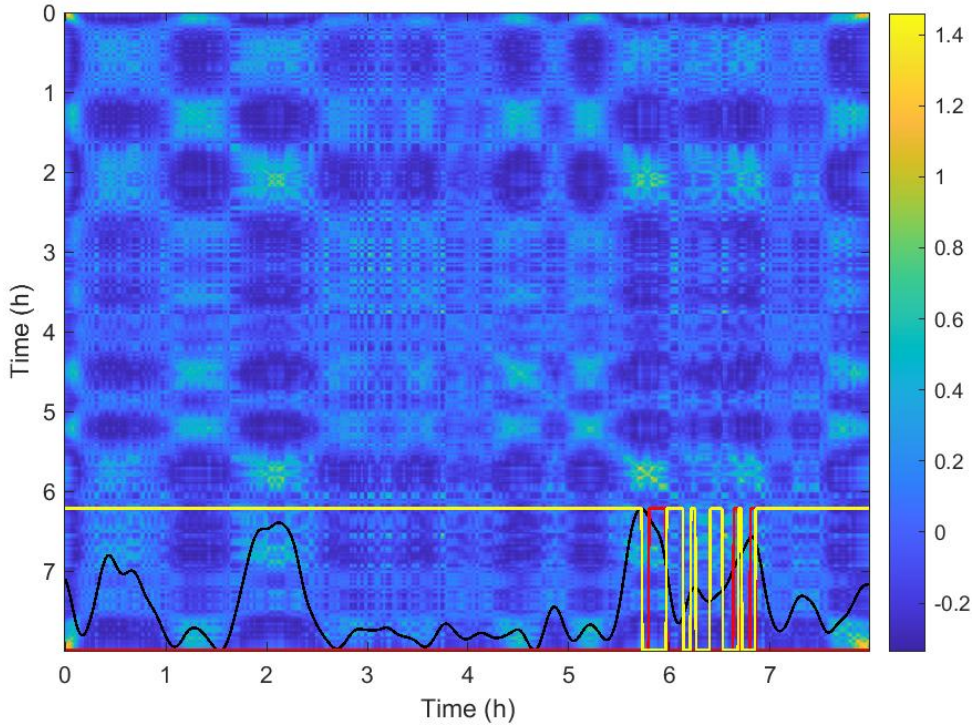


Figure 4.22: Illustration of a second situation similar to the one in Figure (4.21), but with a more problematic outcome, due to higher class-mixture in the features.

4.3.1 Through cross-kernel matrices

In Figure (4.24) we represent the cross-Gram matrix computed between the two datasets of Figures (1.7 and 1.8). As can be seen, intersections of the source and target REMs have colder colours than their surroundings, which is an indicator of class data shift.
In Figure (4.23) we represent the kernel matrix of the recording given as example of intra-individual variability: as can be seen, first and second REM cycles (represented in resp. blue and green in upper panel) resemble each other − hotter colours in the kernel matrix and closer proximity in the 3D representations, as do third and fourth cycles (resp. red and yellow).

4.3.2 Observation selection using Elastic Net and SVMs

In [57, 58], authors consider equivalences between the Least Absolute Shrinkage and Selection Operator (LASSO) method and SVMs. This inspired us to do *traditional* training set selection (as opposed to training source selection, which we propose in Chapter 3)). Indeed: On one hand, we have SVMs dual optimisation problem, that is:

$$\begin{aligned} \min_{\boldsymbol{\alpha},\, b_{SVM}} \quad & \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_i \alpha_i \\ & s.t.\ 0 \leq \alpha_i \leq C \ \ (\forall i) \\ & \sum_i \alpha_i y_i = 0 \end{aligned} \tag{4.6}$$

where C is SVMs' regularisation parameter allowing to manage the compromise between the margin width and the error term, $k(\cdot,\ \cdot)$ is a kernel function and $(\forall i),\ \mathbf{x}_i$ is a (source) training observation. The optimiser is of the form:

$$f_{SVM}(\mathbf{x}) = \sum_{i|\alpha_i \neq 0} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b_{SVM} \tag{4.7}$$

On the other hand, we here use the Elastic Net (EN) [59] method directly on the source-Gram matrix, to minimise w.r.t. $\boldsymbol{\beta}$ and b_{EN}:

$$||Y - (K\boldsymbol{\beta} + b\mathbf{1}_{N,\,1})||_2^2 + \lambda\big(\alpha||\boldsymbol{\beta}||_1 + (1-\alpha)||\boldsymbol{\beta}||_2^2\big) \tag{4.8}$$

where N is the number of observations, $\lambda \geq 0$ modulates the complexity penalisation and $\alpha \in [0,\ 1]$ how costs weigh. These are determined, for instance, through cross-validation. Such optimisation problem is referred to as:

- Ridge regression for $\alpha = 0$.
- LASSO for $\alpha = 1$.

In all cases, such quantity is minimised by a model of the form:

$$f_{EN}(\mathbf{x}) = \sum_{j|\beta_j \neq 0} \beta_j k(\mathbf{x}_j, \mathbf{x}) + b_{EN} \tag{4.9}$$

Usage as an additional proof of data shift

Both f_{SVM} and f_{EN} models are similar, only they select observations differently: SVMs only select Support Vectors, which are close to the zone of between-class mixture and/or ill-classified observations, while EN tends on selecting easily-classified observations (this we observed experimentally through multiple toy examples) when applied on kernel-target matrices (for which columns and corresponding lines correspond to given observations).
On sleep data, we applied both sparse models on each of our source recordings and kept the support vectors and the observations retained by EN, respectively. These are represented in Figure 4.25: black asterisks and green triangles are the observations selected by EN and corresponding to NREM and REM, respectively; blue pluses and red squares to the observations selected by SVM, and here again to resp. NREM and REM. As can be seen, the mixture area (SVM selection) is very large compared to the areas selected by EN; moreover, this large mixture area is to be compared to the one of each individual night, which is way smaller, as can be seen for example on Figures (1.7 and 1.8); mixture on a typical recording usually constitutes around 10% of a given night. This is another good indicator of the data shift problem lying in sleep data.

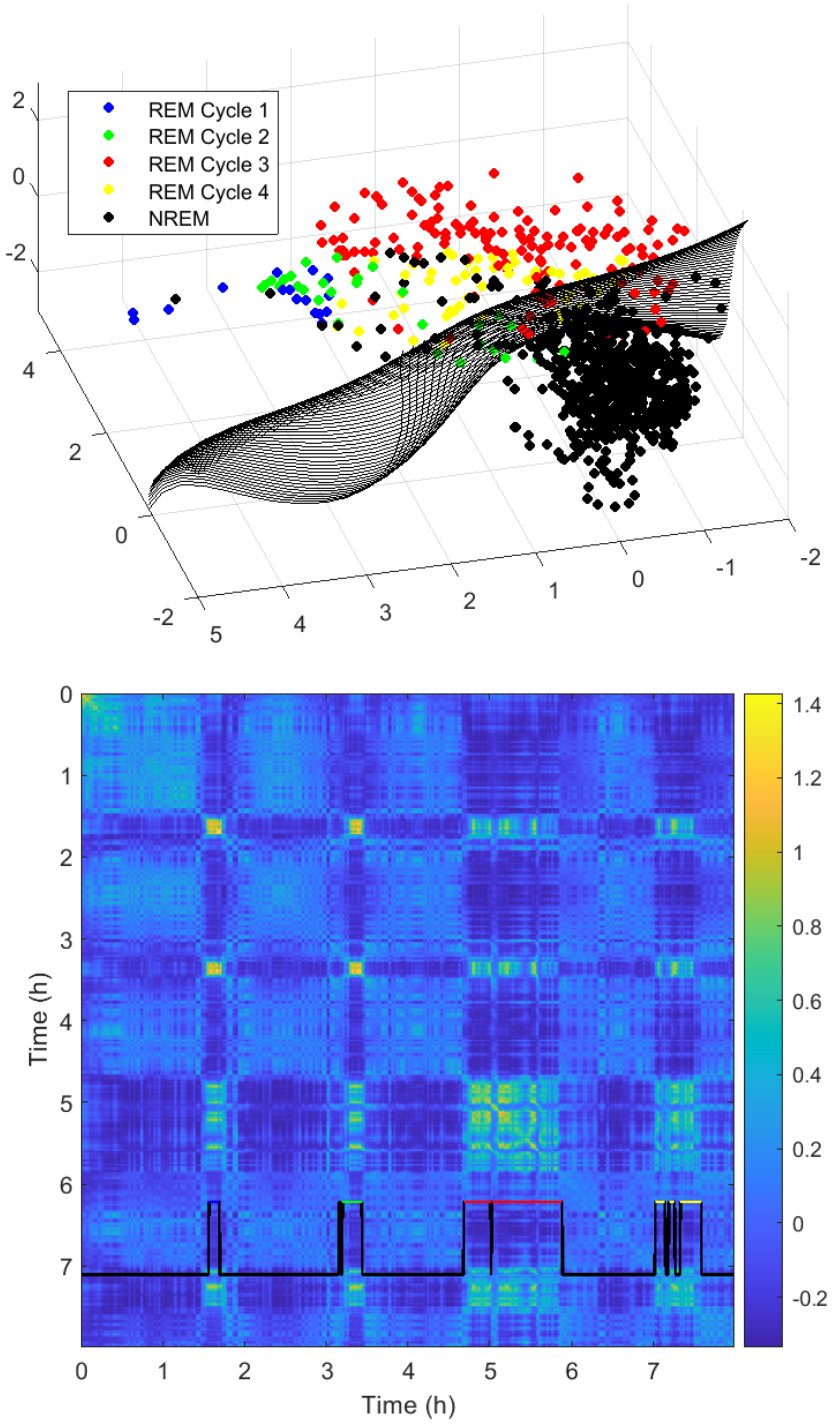


Figure 4.23: Evidence of data shift throughout a given night, as seen through the kernel matrix represented in the lower panel.

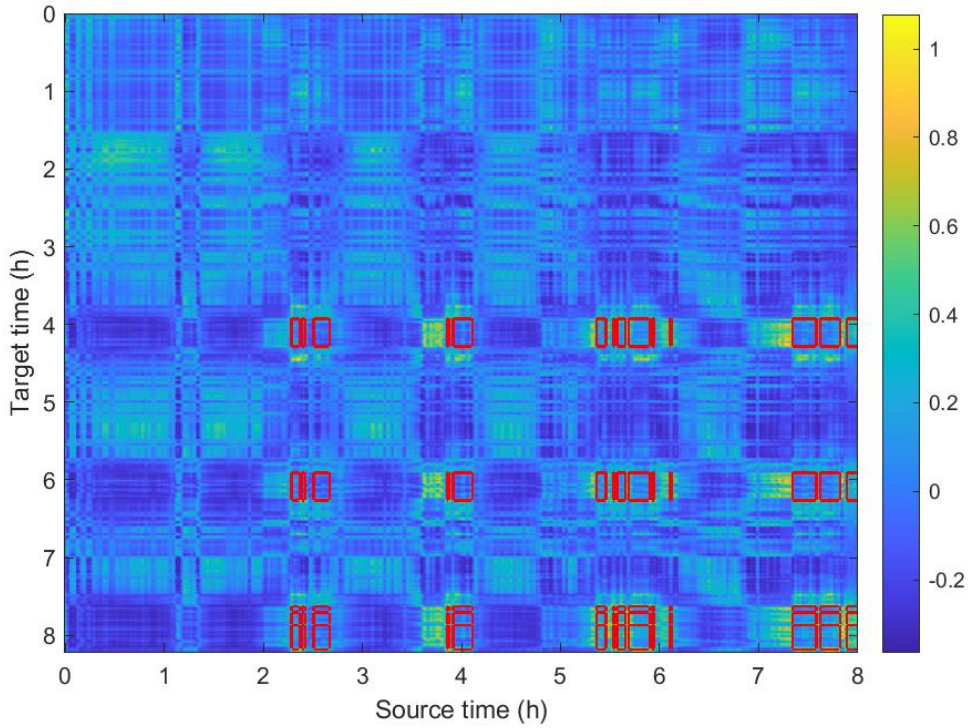


Figure 4.24: Illustration of REM data shift, using the same datasets as in Figures (1.7 and 1.8), this time through the scope of their corresponding cross-Gram matrix. Red squares correspond to REM in both the source and the target.

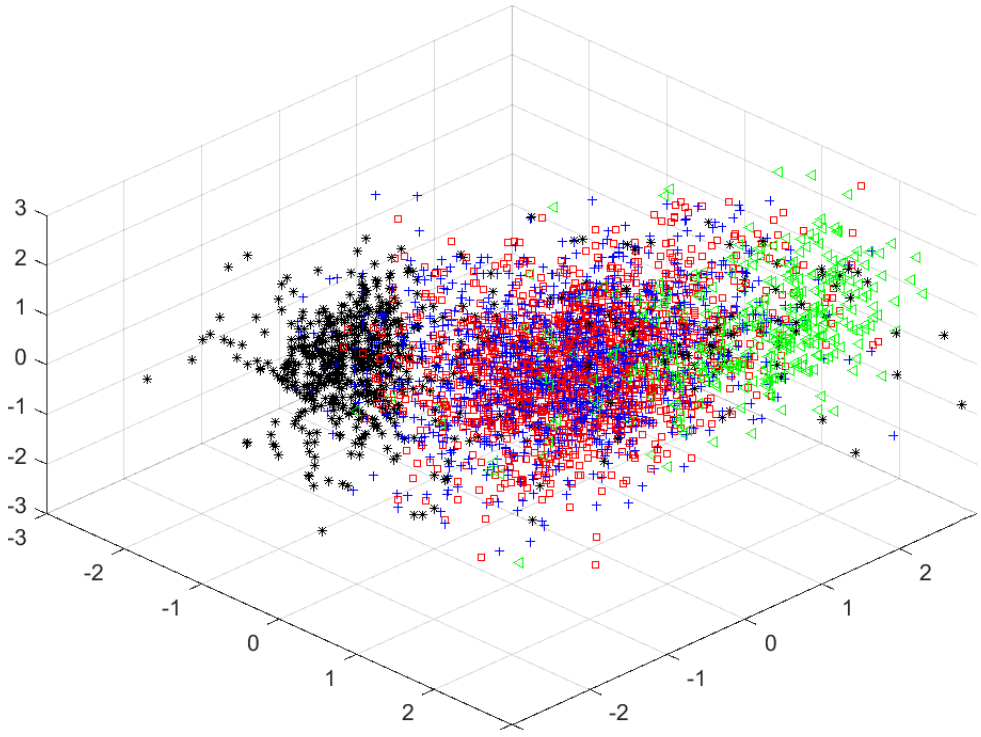


Figure 4.25: Merging on each sleep recording individually of support vectors (blue pluses and red squares) selected by SVM and observations selected by EN on the kernel matrix (black asterisks and green triangles). Selection from SVM is large (larger than the one from EN), which tends to indicate a data shift problem between sleep recordings.

Chapter 5

Operational setup of our methods

Contents

In the previous two chapters, we presented our methods and introduced a few REM detection issues using kernel-based representations. In this chapter, we apply our methods on REM detection through qualitative illustrations; this leads to advised operational setup for each method. We begin by introducing our measures of performances and the (source, target) example in Section (5.1) and thereafter present in three separate sections the direct transfer methods (Section (5.2)), the iterative optimisation methods w.r.t. the kernel centres (Section (5.3)) and finally our predictors for training source selection (Section (5.4)).
We point out that this Chapter's main purpose is to illustrate our methods and their associated parameters effects; while the resulting operating setup is adequate in many situations, there are situations where it will underperform. We present the retained setup in Chapter (6) and the resulting performances on the whole validation dataset.

5.1 Performances assessment

In this section, we introduce our metrics to assess how well a given detector performs on a given target. We thereafter give a few specificities on how our metrics are computed in practice. Finally, we present our (source, target) couple of example recordings, which helps us illustrate our methods applied to REM detection.

5.1.1 Operating points

Quantitative results are given in the form of triplet of values, respectively representing:

- The current operating point (OP) computed from the unmodified decision statistic outputted by a given method: false-alarm rate (FAR) and associated (DR).
- The detection rate associated to the desired operating point, for which the false-alarm rate has been set to 7%. Such operating point would require, in practice, to be able to select the correct threshold for each (source, target) couple, which is difficult without knowledge of the target labels. This only helps to assess our methods "optimal" performances.

Through this chapter, results are therefore presented in the form Current FAR/Current DR, DR at FAR set to 7%. Results should be interpreted as follows:

- The higher the DR at FAR=7%, the better the decision statistic of the considered source applied to the given target.
- The current operating point also informs on how well a given method performs, and gives further indications of the performance dispersion: the closer the current false-alarm rate to the desired one (7%), the less dispersed the results.

We ought to select the method setup maximising the DR at FAR=7% and second minimising the current FAR difference with the desired one. Such setups are highlighted in bold throughout the chapter.
Such performance metrics are illustrated in Figure (5.2): current operating points are represented by the oversized markers, while detection rate at false-alarm rate set to 7% are obtained at intersections of each curve with the vertical black bar.

5.1.2 Example of a target with shifted REM

We mainly use the source and the target illustrated in Figures ((1.7, 1.8) and 4.24) as this is exactly the type of data situation we expect to be able to solve for: there is only little class-mixture, meaning that classes may be separated; the REM class data shifts, meaning that there is a need for decision/domain adaptation. This is a good candidate for qualitatively studying our methods.
In Figure (5.1) we illustrate each method output on this target, starting with the configuration:

- QLTL (green line with stars): K_{TT} computed for $\sigma = 2$ and mean-centred; $\lambda_{rel} = 0.35$; input estimated target labels are the direct output of the SVC trained on the source and applied on the target (as displayed by the black lines in Figure 1.8)).
- KTATL (blue line with triangles): same kernel matrix K_{TT} and input estimated target labels as QLTL.
- KCATL (red line with pluses): K_{TS} computed for $\sigma = 2$ and mean-centred; input source labels obtained by self-application of the source detector on its own data.

This yields the following results:

- Reference performances (SVC output = black line with asterisks): operating point 6.68% FAR/7.28% DR; detection rate at desired false-alarm rate: 11.26%.
- QLTL performances: $7.69\%/33.77\%, 23.84\%$.
- KCATL performances: $21.05\%/100.00\%, 51.66\%$.
- KTATL performances: $15.79\%/100.00\%, 100.00\%$.

Meaning that in these current set-ups, only KTATL results in excellent classification performances (although the current FAR is a bit high).
The remainder of this Chapter organises around this example of (source, target) couple with REM data shift: in the following section, we fine-tune our parameters; next, we consider KCATL with co-optimised centring applied to this couple; in the final section, we consider the target and study our performances predictors using a subset of four sources to select upon. Conclusions based on this example have been confirmed on the remainder of the validation dataset, which will be the subject of Chapter (6).

5.2 Direct decision adaptation

In this Section, we study our transfer methods (QLTL, KCATL, KTATL) sensitivity to their (hyper-)parameters.

5.2.1 Compromise parameter λ_{rel} (QLTL only)

The greater the compromise parameter, the more important the choice of σ; indeed, for $\lambda_{rel} = 0$, the unsupervised information of K_{TT} is not taken into account, meaning that σ has no influence on QLTL results in this situation. The other way around, for $\lambda_{rel} \to 1$, the choice of σ is critical as it directly affects the Gram matrix eigenvectors, that is the non-supervised part of the solution.
In Figure (5.3), we illustrate QLTL outputs for λ_{rel} ranging from 0 to 0.9 by steps of 0.15. Other parameters are unchanged and correspond to the ones of Subsection (5.1.2). Results are (resp. current operating point and detection rate at false-alarm rate fixed at 7%):

- $\lambda_{rel} = 0$, results of the reference detector (i.e. no transfer).
- $\lambda_{rel} = 0.15$: 6.48%/11.26%, 18.54%
- $\lambda_{rel} = 0.30$: 6.28%/20.53%, 33.77%
- $\lambda_{rel} = 0.45$: 7.09%/57.62%, 57.62%; the unsupervised information starts to be taken into account, resulting in improved performances.
- $\lambda_{rel} = 0.60$: 6.68%/100.00%, 100.00%
- $\lambda_{rel} = 0.75$: 6.07%/100.00%, 100.00%;
- $\lambda_{rel} = 0.90$: 7.29%/100.00%, 100.00%; this corresponds to the overall best performances among the selected experiments, while also being arguably equivalent to the performances of $\lambda_{rel} = 0.60$ and $\lambda_{rel} = 0.75$. We highlight the corresponding decision statistic as a black line in Figure (5.3).

This example clearly shows how important the unsupervised information contained in the target kernel matrix is to correct the supervised information given by the source, which miss-classifies all REM observations.

5.2.2 Kernel parameter influence

For the three transfer methods, we consider the decision statistics outputted for $\sigma = 0.1, 0.2, 0.5, 1, 2, 5$ and 10; this is illustrated in Figure (5.4) and corresponding measures of performances are given in Table (5.1). We once again highlight in black the best results obtained for each method: $\sigma = 2$ for KCATL, $\sigma = 10$ for KTATL and QLTL.
We conclude the following:

- Both KTATL and QLTL efficiently take advantage of the unsupervised information contained in K_{TT}. In Subsection (4.1.1), we express that a balanced choice of σ results in a good dynamic for K_{TT}: smooth enough to ignore small changes in the signal, but sensitive enough to detect transitions from REM and to REM; this seems to be confirmed here, as both methods gave almost ideal results in for $\sigma \geq 3$: only the current operating point gets better for greater values of σ.
- KCATL performs poorly whatever the value of σ. Intuitively, if the source is too far from the target in terms of data distribution, K_{TS} may not suffice to bridge for data shift. This last statement requires further investigation.
- While KCATL performs poorly for this (source, target) couple, we observe interesting global results in application on the whole target dataset. This is presented in the following Chapter.

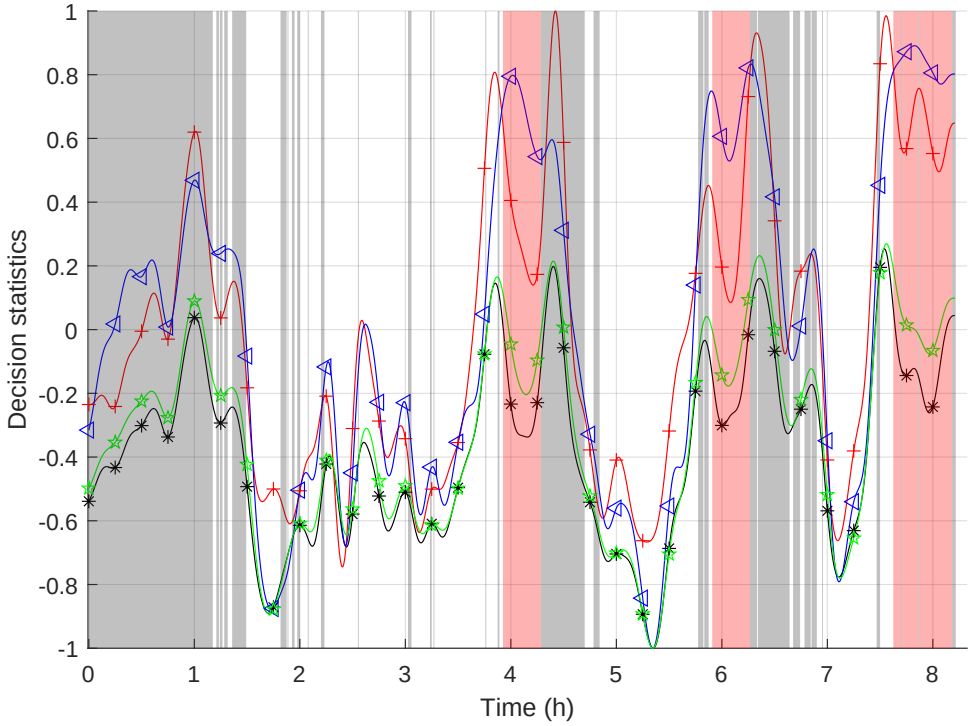


Figure 5.1: Reference output (black line with asterisks), QLTL output (green line with stars), KCATL output (red line with pluses) and KTATL output (blue line with triangles) for the target displayed in Figure (1.8) and using source detector of Figure (1.7). Red areas correspond to REM ground-truth, grey areas are not considered in performance assessment.

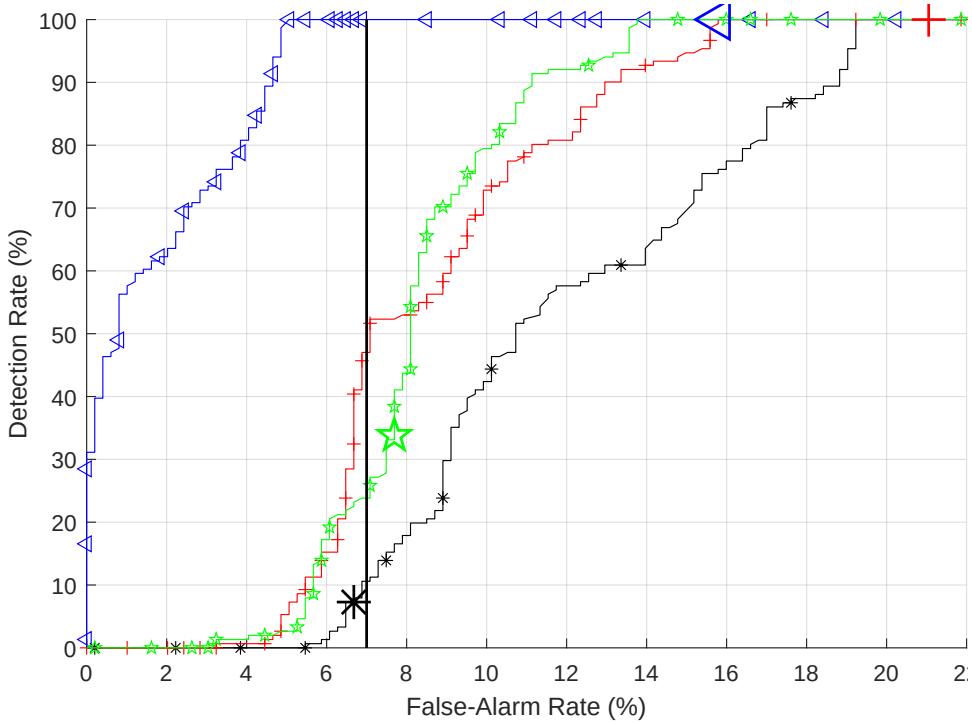


Figure 5.2: Resulting Receiving Operator Characteristics (ROC) Curves of detectors in Figure (5.1): oversized markers represent the current Operating Point (OP) of each method, while vertical bar at FAR$=7\%$ correspond to the desired OP.

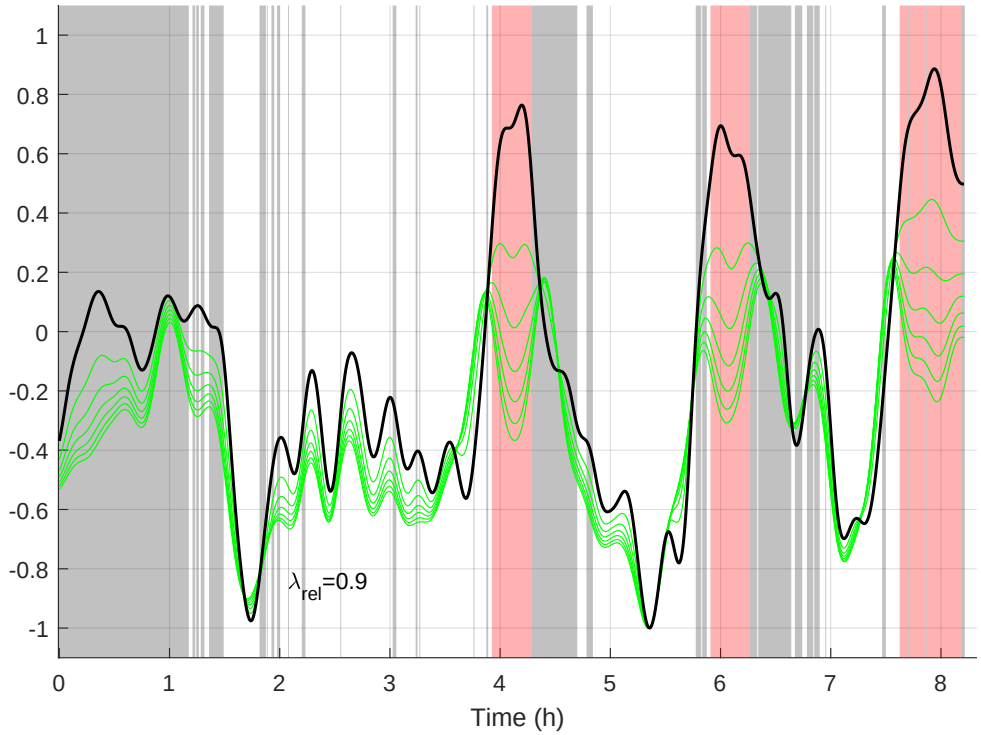


Figure 5.3: Decision statistics outputted by QLTL for $\lambda_{rel} = 0,\ 0.15,\ 0.3,\ 0.45,\ 0.6,\ 0.75$ and 0.9. The best curve is highlighted in black and corresponds to $\lambda_{rel} = 0.90$. Red areas correspond to REM ground-truth, grey areas are not considered in performance assessment.

Kernel par.	KCATL	KTATL	QLTL
$\sigma = 0.1$	35.22%/100.00%, 26.49%	24.70%/100.00%, 1.99%	11.74%/9.93%, 0.00%
$\sigma = 0.2$	33.81%/100.00%, 29.14%	24.90%/100.00%, 0.66%	10.12%/0.00%, 0.00%
$\sigma = 0.5$	26.52%/100.00%, 23.18%	22.67%/100.00%, 3.97%	21.26%/1.99%, 0.00%
$\sigma = 1$	21.86%/100.00%, 37.75%	19.23%/100.00%, 58.94%	22.27%/62.91%, 0.00%
$\sigma = 2$	**21.05%/100.00%, 51.66%**	15.79%/100.00%, 100.00%	12.55%/100.00%, 100.00%
$\sigma = 5$	22.87%/77.48%, 42.38%	13.16%/100.00%, 100.00%	8.50%/100.00%, 100.00%
$\sigma = 10$	23.08%/75.05%, 39.07%	**12.75%/100.00%, 100.00%**	**7.89%/100.00%, 100.00%**

Table 5.1: Current operating point (FAR/DR) and DR at FAR$= 7\%$ for multiple values of sigma and for each transfer method. Best results are highlighted in bold.

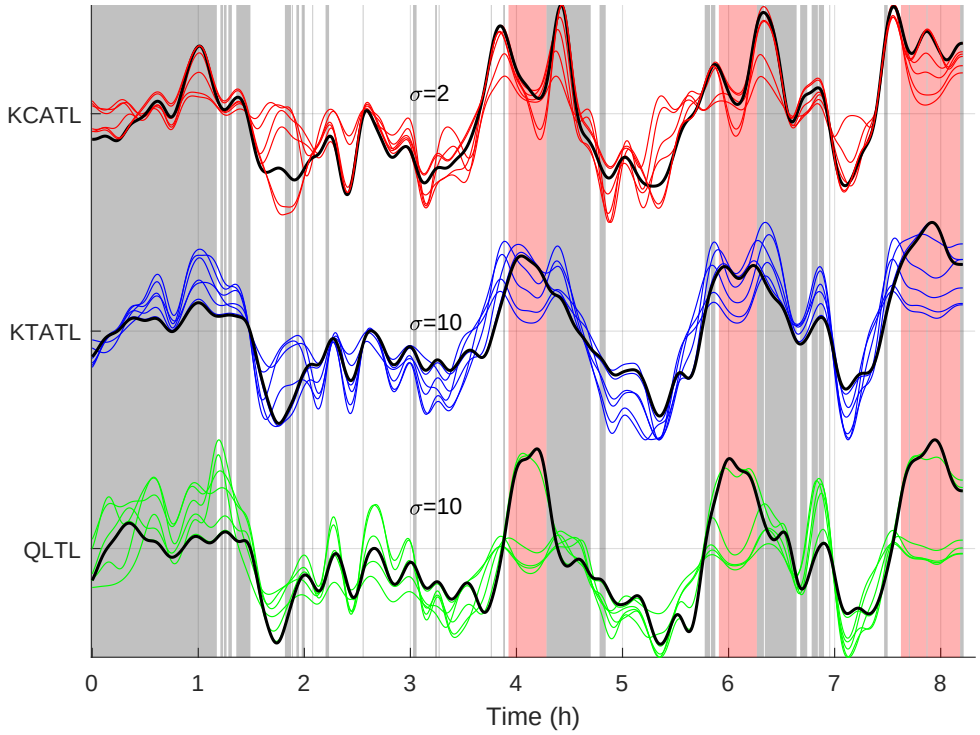


Figure 5.4: Output of each transfer method (specified in y-axis caption) for different values of the standard Gaussian kernel parameter σ (seven curves corresponding to $\sigma = 0.1, 0.2, 0.5, 1, 2, 5$ and 10). QLTL balancing parameter λ_{rel} set to 0.95 so to maximise λ effect (as explained in Subsection (5.2.2)). Best results are highlighted in black and the corresponding parameter value is displayed for each curve.

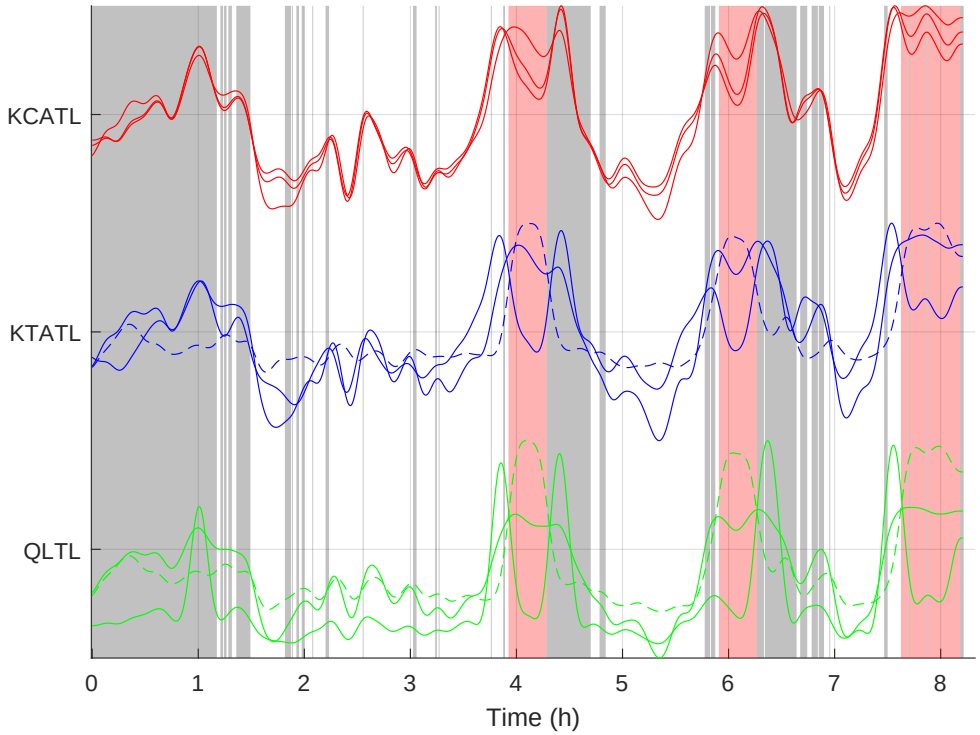


Figure 5.5: Output of each transfer method (specified in y-axis caption) for different codings of the input supervised information (source labels in the case of KCATL, estimated target labels in the case of KTATL). As the target true labels are unknown, we display the corresponding results as discontinuous lines.

5.2.3 Comparison to an up to date joint density optimal transport method

We give here a quick presentation of a very efficient transfer method from the literature (JDOT, [32]), to which we thereafter compare our methods. JDOT relies on the joint optimisation of both the density transport of the observations, but also of the labels, w.r.t. both a transport matrix $\Gamma = (\gamma_{ij}) \ \forall (i \in \{1, \cdots, n_S\}, j \in \{1, \cdots, n_T\})$ and the classification model f:

$$\begin{aligned} \min_{\Gamma, f} \sum_{i,j} \gamma_{ij} \Big(\beta d(\mathbf{x}_i^S, \mathbf{x}_j^T) + (y_i^S - f(\mathbf{x}_j^T))^2 \Big) + \lambda \Omega(f) \\ s.t. \ \sum_{j=1}^{n_T} \gamma_{ij} = \frac{1}{n_S} \ \forall i \in \{1, \cdots, n_S\} \\ \sum_{i=1}^{n_S} \gamma_{ij} = \frac{1}{n_T} \ \forall j \in \{1, \cdots, n_T\} \end{aligned} \tag{5.1}$$

where d denotes the Euclidean distance, $\mathbf{x}_i^S$ and $\mathbf{x}_j^T$ source and target observations, and y_i^S the source labels. $f \in \mathcal{H}$ ($\mathcal{H}$ is a RKHS) is the sought classification model to assess target labels, Ω is a complexity penalisation function, and n_S and n_T denote the source and target number of observations, respectively. λ and β are compromise parameters. We consider both minimisation problems; tackling the one w.r.t. Γ while f is fixed leads to minimising the distance cost between each source observation and each target one (transport minimisation), this is solved by any simplex algorithm, among others.
Tackling the minimisation w.r.t. the classification model f while Γ if fixed is developed below for $\Omega(f) = ||f||_{\mathcal{H}}^2$:

$$\min_f \sum_{i,j} \gamma_{ij} (y_i^S - f(\mathbf{x}_j^T))^2 + \lambda ||f||_{\mathcal{H}}^2 \tag{5.2}$$

Considering the previous expression, and using the representer theorem, we have:

$$f(\star) = \sum_{i=1}^{n_T} \alpha_i k(\star, \mathbf{x}_i^T) \tag{5.3}$$

so that:

$$||f||_{\mathcal{H}}^2 = \langle \sum_{i=1}^{n_T} \alpha_i k(\star, \mathbf{x}_i^T), \sum_{j=1}^{n_T} \alpha_j k(\star, \mathbf{x}_j^T) \rangle_{\mathcal{H}} = \boldsymbol{\alpha}' K \boldsymbol{\alpha} \tag{5.4}$$

where $\boldsymbol{\alpha}$ is the column vector containing α_i values, $\forall i \in \{1, \cdots, n_T\}$. Thanks to the quadratic nature of $\sum_{i,j} \gamma_{ij} (y_i^S - f(\mathbf{x}_j^T))^2$, we obtain this equivalent minimisation problem:

$$\min_{f \approx \boldsymbol{\alpha}} \ \frac{1}{n_T} \sum_{j=1}^{n_S} (\hat{y}_j^T - f(\mathbf{x}_j^T))^2 + \lambda \boldsymbol{\alpha}' K \boldsymbol{\alpha} \tag{5.5}$$

where $(\hat{y}_j^T)_{j \in \{1, \cdots, n_T\}}$ is the j^{th} component of $\hat{Y}^T = n_T \Gamma' Y^S$, Y^S being the source labels column vector. In matrix form, it is shown that such problem is equivalent to:

$$\approx \min_{\boldsymbol{\alpha}} \ \frac{1}{n_T} ||\hat{Y}^T - K\boldsymbol{\alpha}||^2 + \lambda \boldsymbol{\alpha}' K \boldsymbol{\alpha} \tag{5.6}$$

finally leading to the solution:

$$\boldsymbol{\alpha} = (K + \lambda n_T \mathcal{I}_{n_T})^{-1} \hat{Y}^T \tag{5.7}$$

The nature of the optimisation problem allows the use of an alternate optimisation scheme on Γ and $\boldsymbol{\alpha}$, leading in a few iterations to the global optimal solution (considering the criterion and the constraints, existence of a single optimum is demonstrated in [32]).
We illustrate JDOT results on the same source and target pair as in Figure (5.1) and subsequent. To use this method, we had to decimate the data with a ratio (1:100); we use tradeoff parameters $\beta = 0.2$ and $\lambda = 0.1$. The

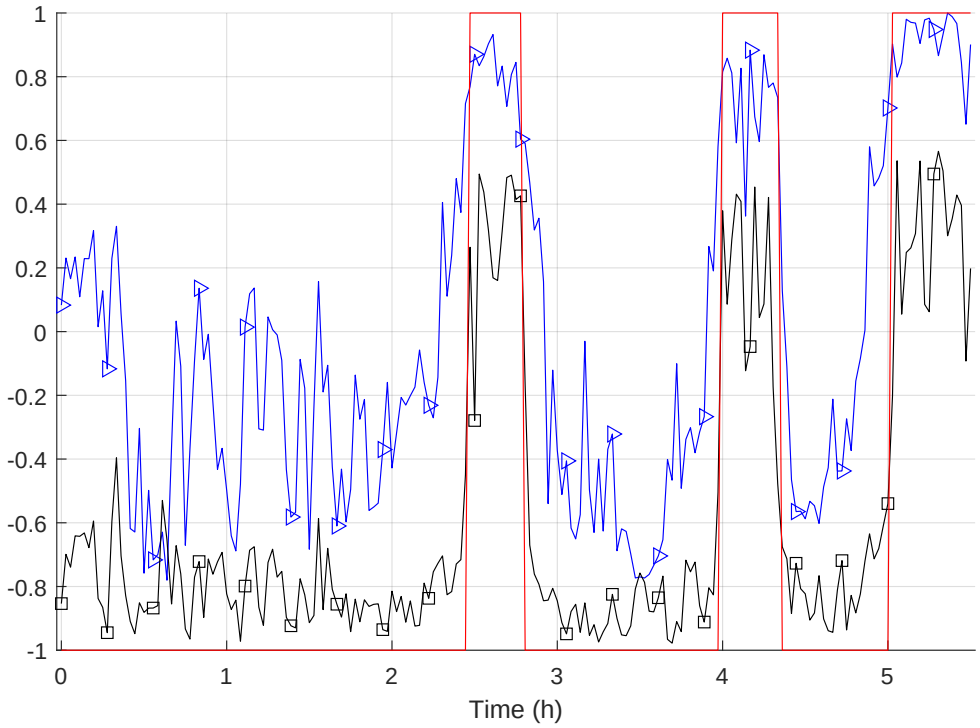


Figure 5.6: Output of the JDOT algorithm [32] - black line with squares - to be compared to the output of KTATL - blue line with triangles. The red line represents the ground-truth.

computation time was 8s to be compared with 0.01s using KTATL. Computation time of JDOT increases a lot with larger numbers of observations to classify, more than linearly, so we cannot use this method for our application. As can be seen in Figure (5.6), results look similar (but we have to carefully select the threshold using KTATL) at the price of a computation time almost 800 times larger. This clearly illustrates our need for non iterative transfer methods (such as KxATL). The similarity between KTATL and JDOT outputs has been verified on other couples of recordings.

5.2.4 Labels coding effect

We consider three ways of inputting supervised information:

- The non-binarised decision statistic outputted by the source detector applied to the target: Y_T^S for KTATL, Y_S^S for KCATL.
- The binarised decision statistic: $sign(Y_T^S)$ for KTATL, $sign(Y_S^S)$ for KCATL.
- The ground-truth labels, which are only available for the source, and therefore only usable in KCATL: Y_S.

We represent each method output for each proposed label coding in Figure (5.5) and corresponding results in Table (5.2). We make the following conclusions:

- If target labels were known, there would be nothing to transfer (more than that, there would be nothing to detect); QLTL and KTATL decision would be almost perfect, as illustrated by the dashed-lines. This only serves an illustrative purpose, and is unusable in practice.
- The best configuration for inputting supervision is to use the non-binarised decision statistic outputted by the source detector on itself - KCATL - or on the target - QLTL, KTATL.

Labels coding	KCATL	KTATL	QLTL
Non-binarised decision statistic	16.19%/100.00%, 100.00%	15.79%/100.00%, 100.00%	7.49%/100.00%, 100.00%
Binarised decision statistic	19.43%/100.00%, 54.30%	25.51%/66.89%, 0.66%	7.09%/12.58%, 12.58%
Ground-truth labels	21.05%/100.00%, 51.66%	4.45%/100.00%, 100.00%	3.04%/100.00%, 100.00%

Table 5.2: Current operating point (FAR/DR) and DR at FAR$= 7\%$ for multiple ways of inputting knowledge and for each transfer method.

- Such a result may be explained by an improved coherence between the unsupervised information and the supervised one: non-binarised decision statistic is similar to the eigenvectors of K_{TT} or to the left-singular vectors of K_{TS}.
- This is especially true in the case of KCATL, where we obtain acceptable performances, which was not the case whatever the value of σ using the ground-truth labels.

5.2.5 Conclusion on Transfer Learning methods

As aforementioned, while this Chapter focuses only around one (source, target) couple of recordings, it is representative of many of our cases of data shift. Resulting choices of parameters are adequate for most situations. Therefore, following what has been presented in Subsections (5.1.2 to 5.2.4), we advise:

- To input supervised information by using directly the decision statistic outputted by the source detector applied to the target (QLTL, KTATL) or to the source training data (KCATL). This ensures better coherence between the content of the Gram matrix (unsupervised information) and supervised information.
- To select a value for σ that will ensure optimal unsupervised information. This should be done in a validation step.
- For QLTL, to select a value for λ that will adequately compromise between supervision and non-supervision. This, too, should be done in a validation step.

However, optimal parameters depend on the (source, target) couple. While our recommendations are valid for most situations, they might be inadequate in specific set-ups. Methods to select the right parameters for any (source, target) couple are still under investigation.
In this Section, we also omitted to analyse the influence of kernel-centring: without some knowledge of the labels, we can only choose either not to centre, or to centre using the average value in the RKHS/in input space. The latter systematically outperforms the former. As a conclusion, so far, QLTL and KTATL work the best and it is quite easy and direct to take advantage of the unsupervised information contained in K_{TT}. The case of KCATL seems more complicated, and requires a specific set-up to give satisfying results: the use of non-binarised, self-estimated source labels, and careful centring. In the following Section, we tackle the centring of K_{TS} for KCATL, by maximising KCA (or KCP) w.r.t. the centres (as presented in Section (3.3)).

5.3 Iterative centres optimisation

In this section, we apply our work of [54] − succinctly developed in Annex B − to the application, using once again the (source, target) couple displaying an obvious REM data shift. We consider six centre optimisation problems:

- Maximisation of KCA w.r.t. both source and target centres in the RKHS. (Figures (5.7 and 5.8))
- Maximisation of KCA w.r.t. the source centre only in the RKHS. (Figures (5.9 and 5.10))
- Maximisation of KCA w.r.t. both source and target centres in input space. (Figures (5.11 and 5.12))

- Maximisation of KCA w.r.t. the source centre only in input space. (Figures (5.13 and 5.14))
- Maximisation of KCP w.r.t. both source and target centres in input space. (Figures (5.15 and 5.16))
- Maximisation of KCP w.r.t. the source centre only in input space. (Figures (5.17 and 5.18))

Corresponding results are presented in Table (5.3).

5.3.1 Decision boundaries representation

In order to compute the representations displayed in Figures (5.8, 5.10, 5.12, 5.14, 5.16 and 5.18), we have replaced the target dataset by values in a 3D grid. We then apply KCATL between the source and the grid using the centres obtained by optimisation on the true target dataset. Finally, to output the decision boundaries of the corresponding KCATL detector, we form the contours at each value of the z-axis for the decision output set to the true detector decision threshold.
For the sake of clarity, we choose not to add boundaries for other values of decision threshold, as it would surcharge the Figures.

5.3.2 Operational setup

Following what has been presented in the previous Section:

- The input source labels are re-estimated using the source detector.
- The cross-Gram matrix K_{TS} is computed using $\sigma = 2$.

Regarding the optimisation of the centres and of the target labels:

- We initialise the target labels as the output of the source detector applied to the target data.
- Input-space centring: we initialise the source and target centres as the average observation in the source dataset and the target dataset, respectively.
- Centring in the RKHS: we initialise the source and target centres as the average observations in the RKHS for the source observations and the target ones, respectively.
- Maximising KCA: in this case, for the optimisation algorithm, we use gradient step parameter $\eta = 10^{-2}$, and end the algorithm when KCA improves by less than $\Delta_S = 10^{-3}$, where Δ_S is the optimised value evolution between two iterations.
- Maximising KCP: in this second case, $\eta = 10^{-6}$ and $\Delta_S = 10^{1}$ (as KCP might take large values, contrary to KCA).

Such choice of parameters ensures relatively quick convergence on our examples.

5.3.3 Application and results

We comment the results displayed in Table (5.3):

- Resulting KCP is greater when optimising w.r.t. both centres (94874) than when maximising w.r.t. the source centre only (24107): two degrees of liberty allow to reach higher values.
- Similarly, input space optimised KCA is greater when considering both centres (0.86) than just the source one (0.73).
- In the case of optimising in the RKHS, results between dual-optimisation and single-optimisation are really close, both for the maximised measure and for the classification performances. We further comment on that below.

Regarding the resulting classifications:

Method	η, Δ_S	KCA (KCP)	Performances
SVC (Ref)		0.17 (10341)	6.68%/7.28%, 11.26%
KCP, Input, Dual	$10^{-6}, 10^{1}$	0.69 (94874)	0.00%/0.00%, 100.00%
KCP, Input, Single	$10^{-6}, 10^{1}$	0.71 (24107)	25.71.%/100.00%, 100.00%
KCA, Input, Dual	$10^{-2}, 10^{-3}$	0.86 (30948)	0.20%/2.65%, 78.81%
KCA, Input, Single	$10^{-2}, 10^{-3}$	0.73 (15772)	15.99%/100.00%, 100.00%
KCA, RKHS, Dual	$10^{-2}, 10^{-3}$	0.77 (17686)	16.19%/100.00%, 100.00%
KCA, RKHS, Single	$10^{-2}, 10^{-3}$	0.77 (17524)	16.19%/100.00%, 100.00%

Table 5.3: Centring methods results to be compared to the reference method without transfer (SVC); η denotes the gradient step size parameter and Δ_S the stopping criterion, both for optimisation algorithms. In the method column, KCP/KCA denotes the quantity we maximise, Input/RKHS denotes the way we centre and Dual/Single denote whether we optimise w.r.t both centres, or w.r.t the source centre only. Best method and associated results are highlighted in bold.

- All methods, except maximisation of KCA w.r.t. both centres in input space, result in a perfect detection rate of 100% at FAR=7%.
- However, there is a clear decision threshold issue, as the current false-alarm rates are quite far from 7%. For instance, in the case of maximised KCP w.r.t. both centres in input space, it results in full non-detection.

Regarding the selected centres:

- Usually, efficient centres are located in the mixture-between-class area of a dataset. This is expected (refer to the explanations in Subsection (2.4.5) and Figure (2.2)).
- However, in the case of the target centre, this optimal location is unknown as target labels are unknown.
- Part of the optimisation problems set-ups lead to diverging centres. This may lead to poor performances and may require addition of constraints to the problem.
- KCP maximisation w.r.t. centre(s) in the RKHS is not presented here: this is due to experimental results selecting one observation in each dataset as centres for the transfer method. This may be an idea for rapid centre selection in the future.

To conclude on this family of methods, they lead to quite interesting results but may also result in negative transfer, and are highly sensitive to both the initialisation and the gradient step parameters. They currently still require fine-tuning and better understanding.

5.4 Predictors for TSS

5.4.1 Subset of sources

In this section, we consider the same target as before, and denote by S_1 the source detector used in the two previous sections and displayed in Figures (1.7 and 1.8). We introduce three new sources S_2, S_3 and S_4, similarly represented in Figure (5.19 to 5.24). These sources are selected as to result in various performances on the target, that is:

- S_1 results in very poor performances due to the REM data shift.
- S_2 results in excellent performances and would adequately classify target observations.
- S_3 and S_4 both lead to fair performances, but display clear false-alarm and non-detections (as can be seen on Figures (5.22 and 5.24)).

The objective is to study how well our different predictors will perform in determining which source detector to choose for the target, and with which confidence.

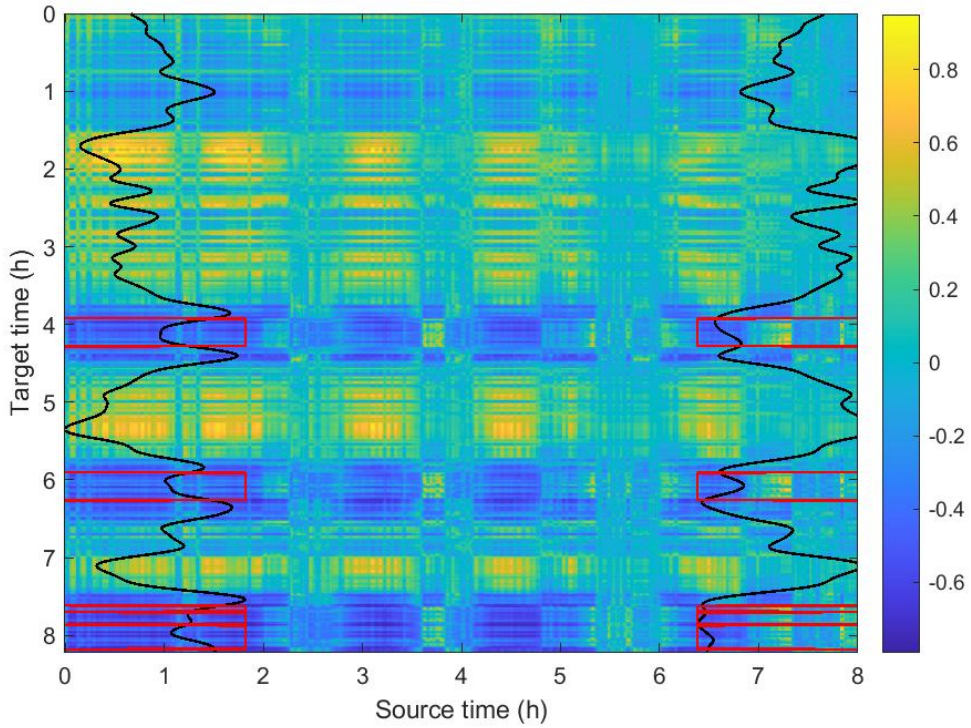


Figure 5.7: Kernel matrix with optimised source and target centres (expressed in the RKHS and maximising KCA); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

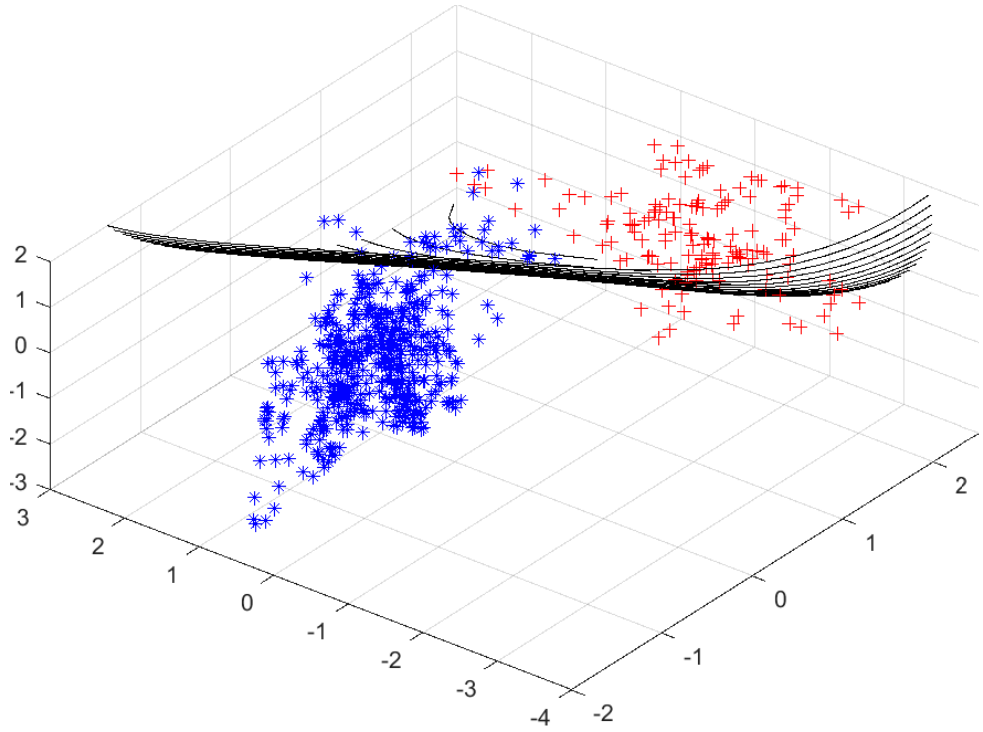


Figure 5.8: Representation of the decision boundary for KCATL output using centres described in Figure (5.7): maximisation of KCA w.r.t. both source and target centres in the RKHS.

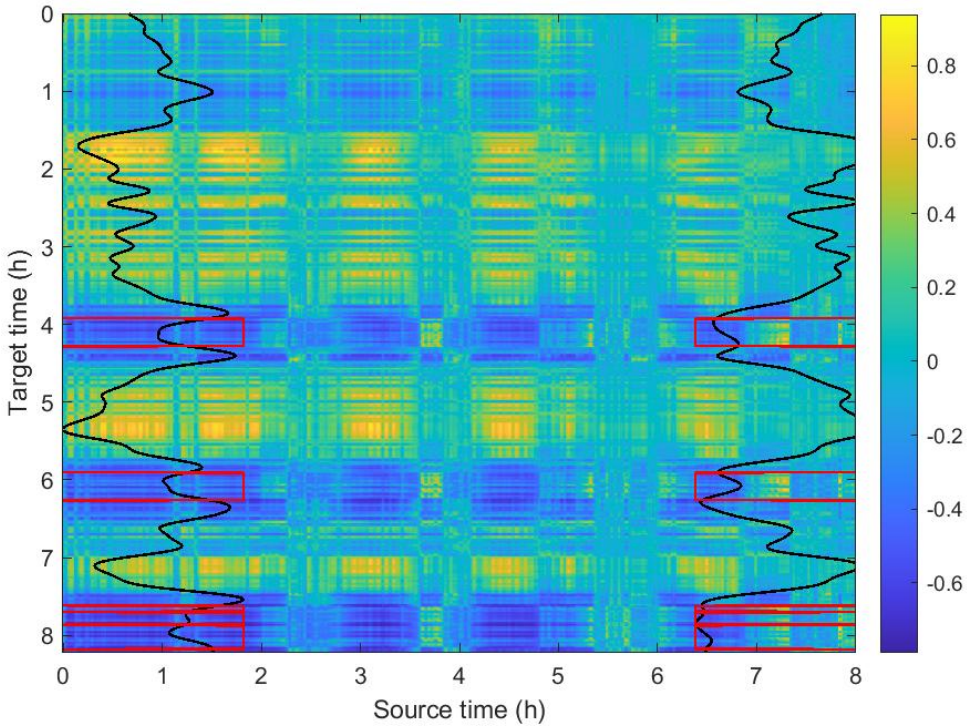


Figure 5.9: Kernel matrix with optimised source centre (expressed in the RKHS and maximising KCA); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

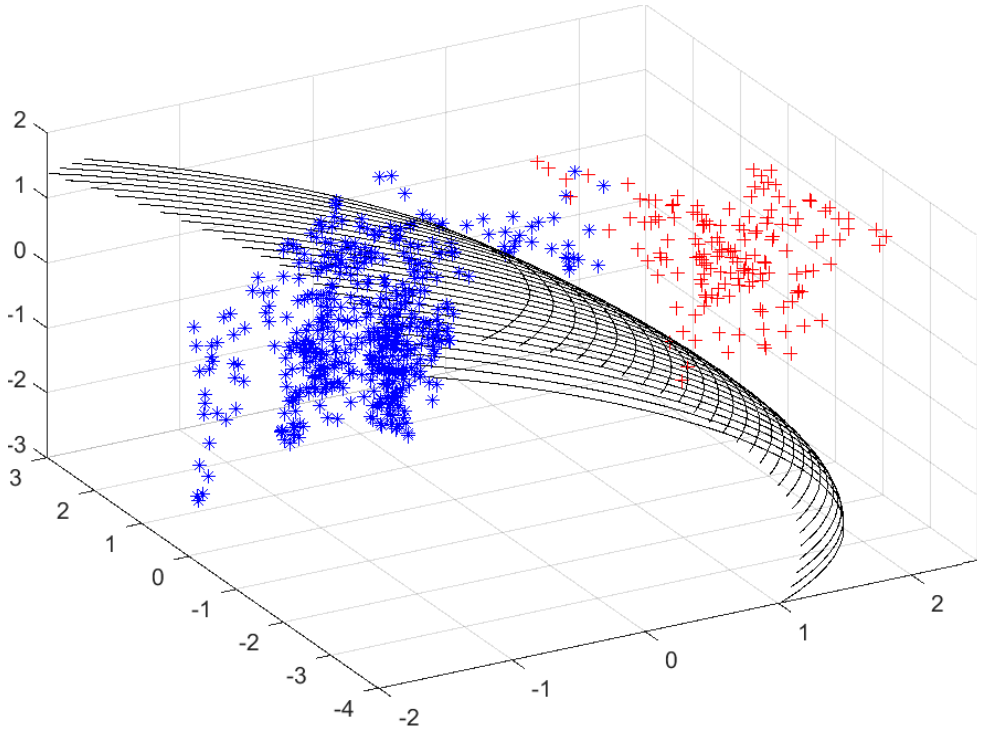


Figure 5.10: Representation of the decision boundary for KCATL output using centre described in Figure (5.9): maximisation of KCA w.r.t. solely the source centre in the RKHS.

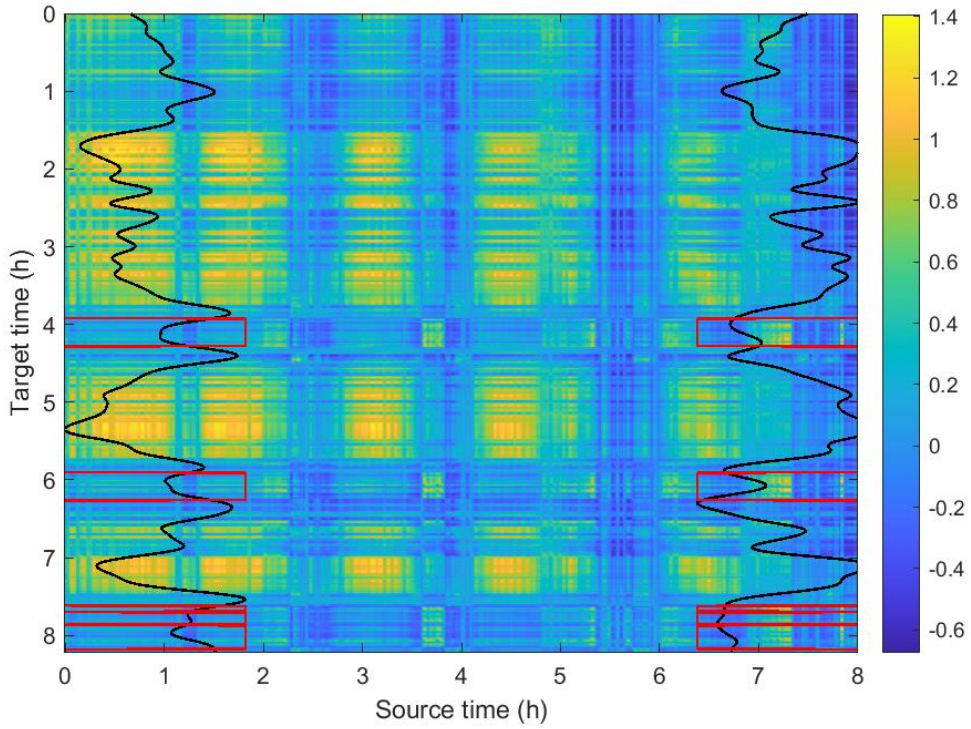


Figure 5.11: Kernel matrix with optimised source and target centres (expressed in input space and maximising KCA); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

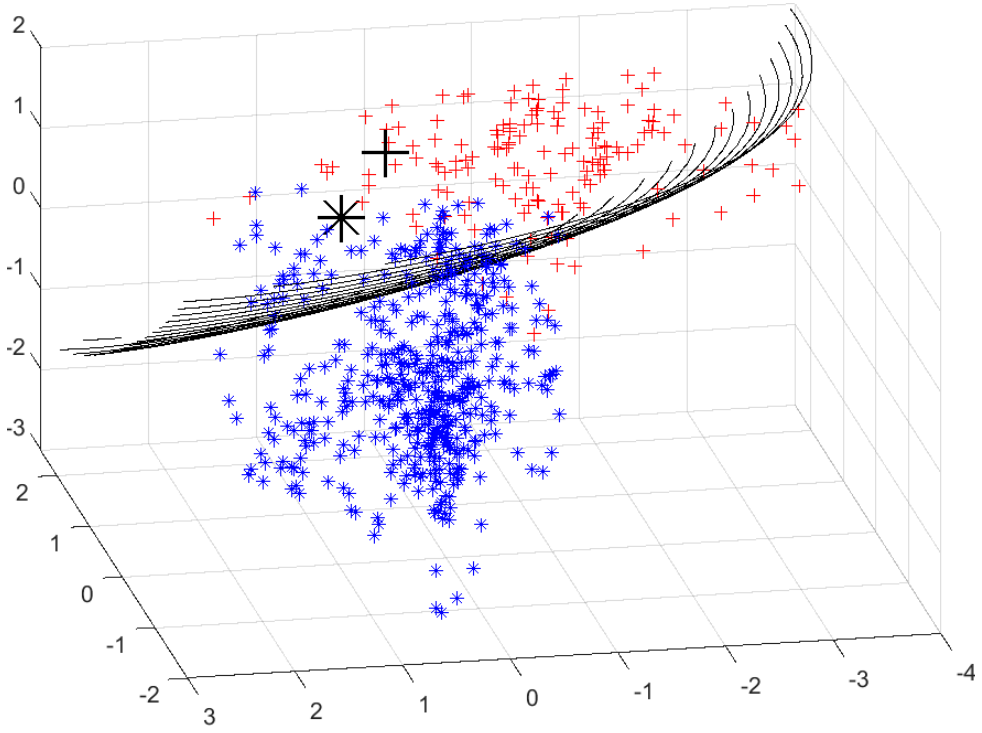


Figure 5.12: Representation of the decision boundary for KCATL output using centres described and represented in Figure (5.11): maximisation of KCA w.r.t. both source and target centres in input space. These are represented by the bold black cross and bold black asterisk, respectively.

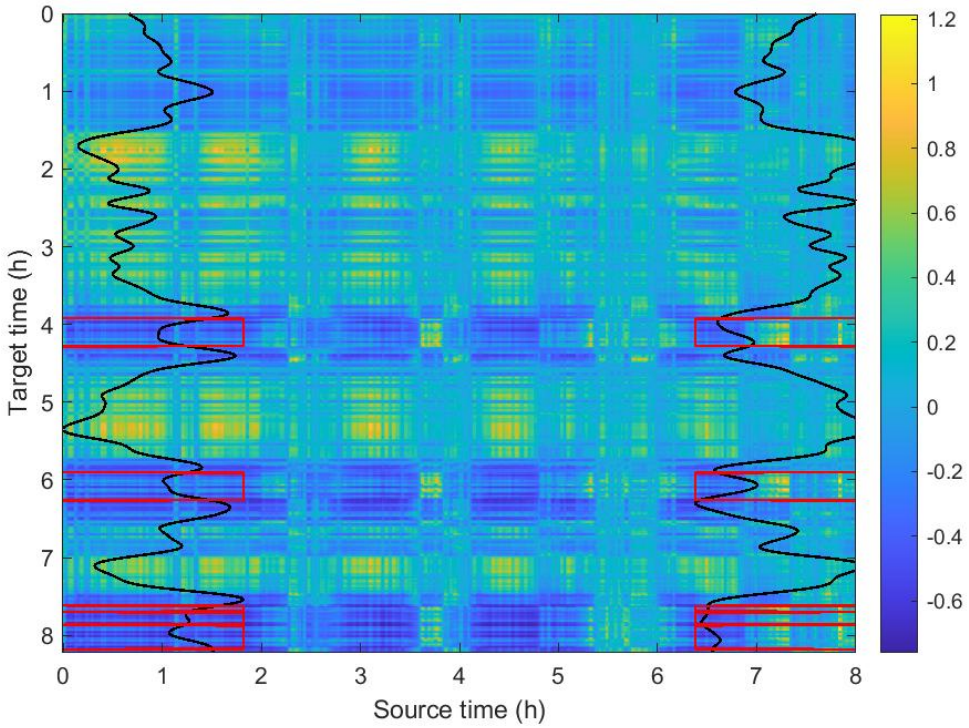


Figure 5.13: Kernel matrix with optimised source centre (expressed in input space and maximising KCA); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

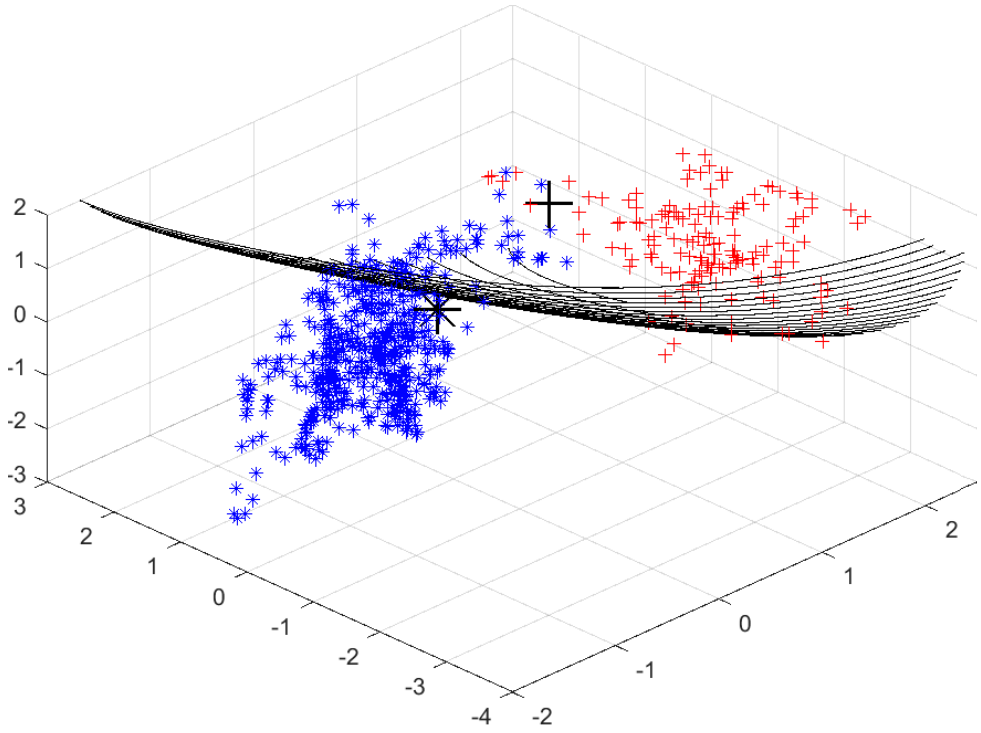


Figure 5.14: Representation of the decision boundary for KCATL output using centres described and represented in Figure (5.13): maximisation of KCA w.r.t. solely the source centre in input space (bold black cross). The bold black asterisk represents the target centre, which is fixed to the average value of all observations.

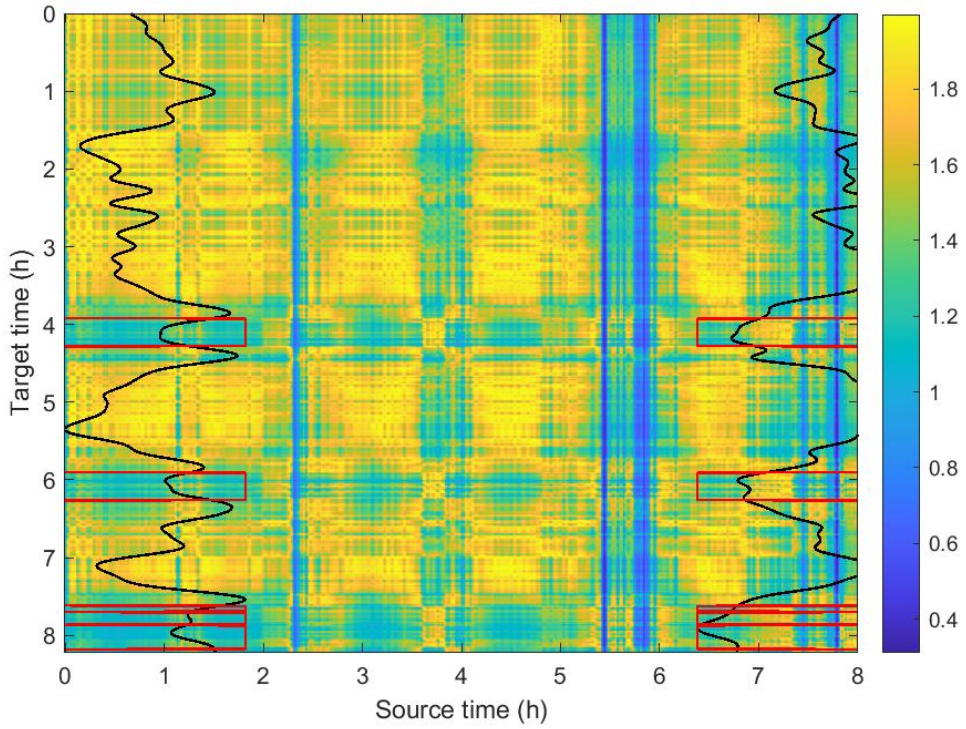


Figure 5.15: Kernel matrix with optimised source and target centres (expressed in input space and maximising KCP); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

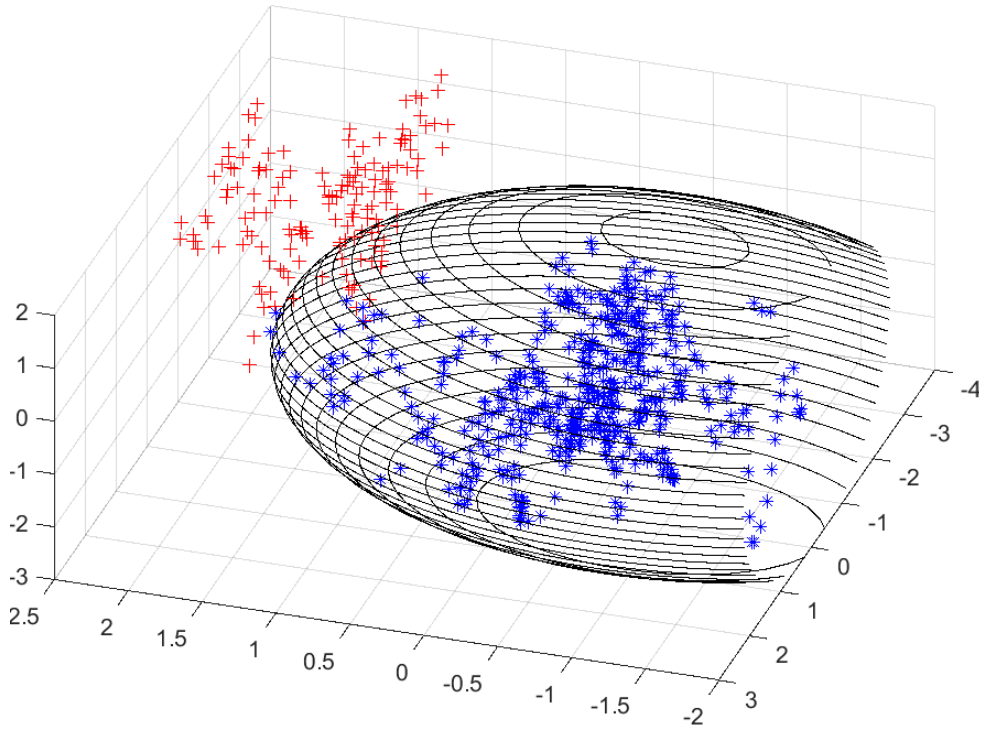


Figure 5.16: Representation of the decision boundary for KCATL output using centres described and represented in Figure (5.15): maximisation of KCP w.r.t. both source and target centres in input space. These are not represented here, as they are out of the representation boundaries.

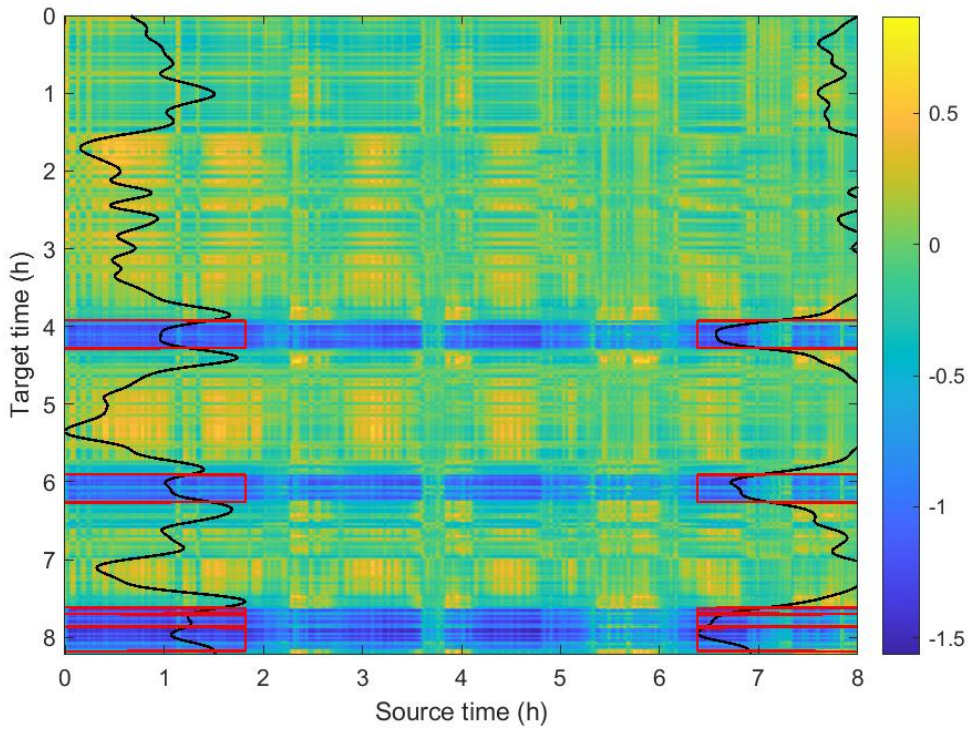


Figure 5.17: Kernel matrix with optimised source centre (expressed in input space and maximising KCP); corresponding KCATL output (right black line), reference output (left black line) and ground-truth REM (red lines).

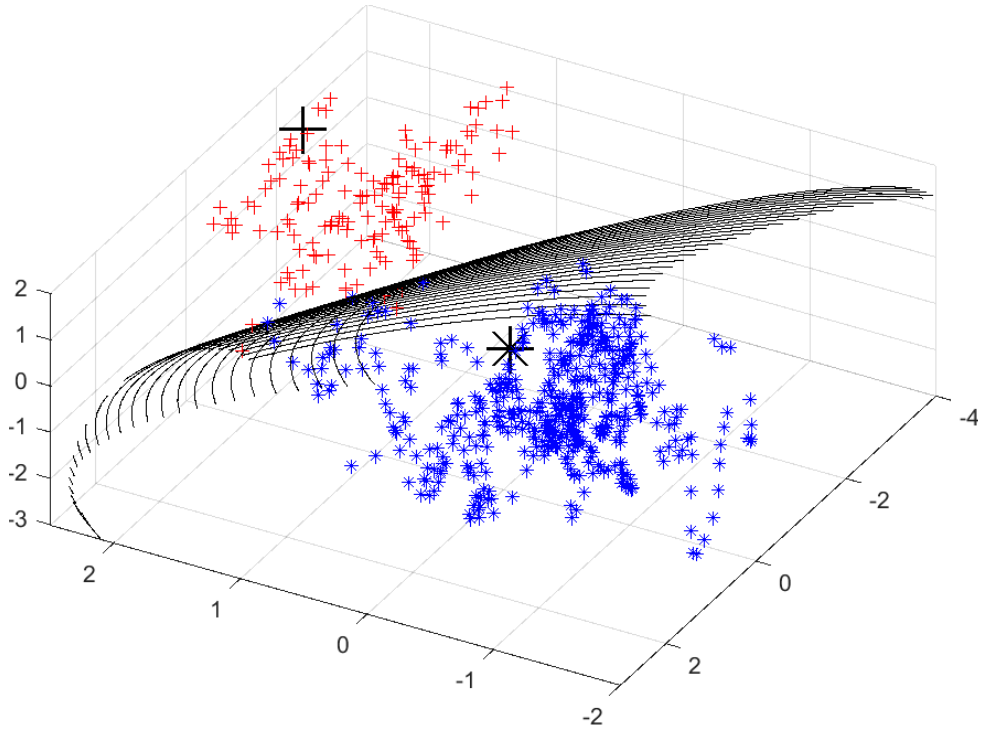


Figure 5.18: Representation of the decision boundary for KCATL output using centres described and represented in Figure (5.17): maximisation of KCP w.r.t. solely the source centre in input space (bold black cross). The bold black asterisk represents the target centre, which is fixed to the average value of all observations.

Metric	Source S_1	Source S_2	Source S_3	Source S_4
Obs. KCA	0.27	**0.37**	0.26	0.32
Obs. KTA	0.15	**0.44**	0.21	0.29
Max. KCA	**0.49**	0.45	0.42	0.47
Max. KTA	0.27	**0.53**	0.36	0.41
Correl. KCA	0.68	**0.81**	0.69	0.77
Correl. KTA	0.56	**0.76**	0.68	0.71
Current OP	6.68%/7.28%	5.26%/100.00%	3.44%/58.94%	5.87%/77.48%
DR at FAR=7%	11.26%	100.00%	100.00%	85.43%

Table 5.4: Performance predictors (upper part of the table) and actual performances (lower part of the table) for four considered sources directly applied to the target taken as example in this chapter.

5.4.2 Accuracy of prediction

Table (5.4) displays six of our predictors values for each of the four sources we consider to be applied to T. We observe the following:

- Five out of six predictors would select the correct source for this target.
- For most predictors, prediction value is well correlated to the actual performances (ideal detection rate): maximal values are for S_2, in second place S_3, third place S_4 and last place S_1.
- Maximised KCA seems inadequate in predicting performances: this may be due to the aforementioned results of KCATL, which is required to compute maximised KCA.
- Overall, the best predictors in this example are KTA-based.

TSS is more thoroughly presented in the next Chapter, as considering more sources and all targets lead to additional observations.

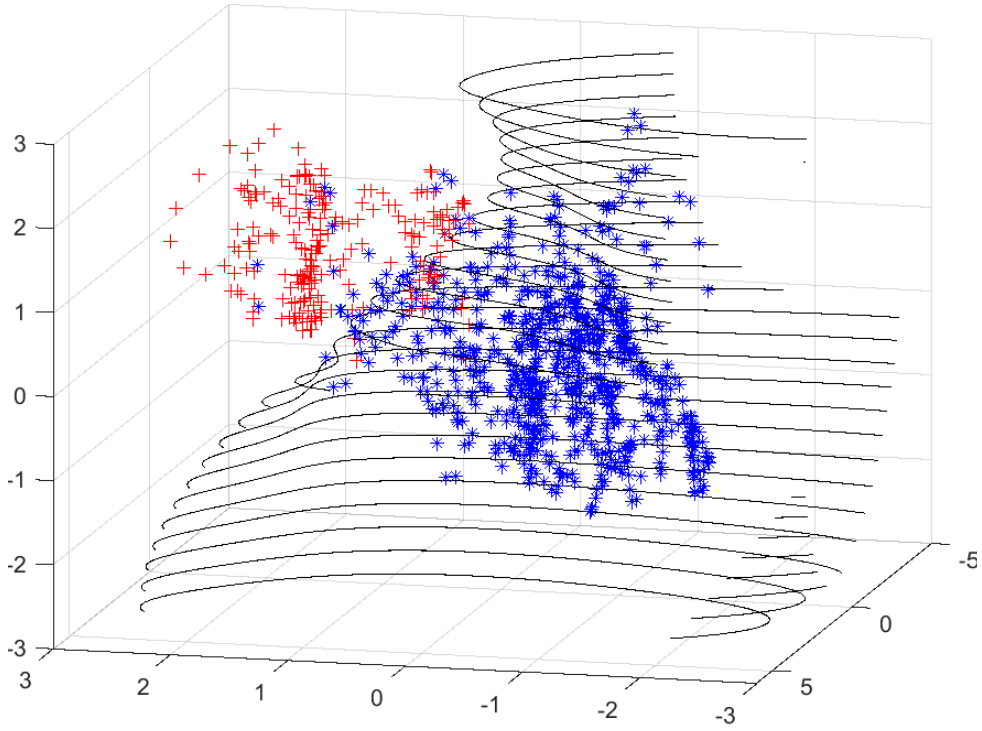


Figure 5.19: Source S_2 observations (blue asterisks = NREM, red crosses = REM) and associated source detector decision boundary (black lines).

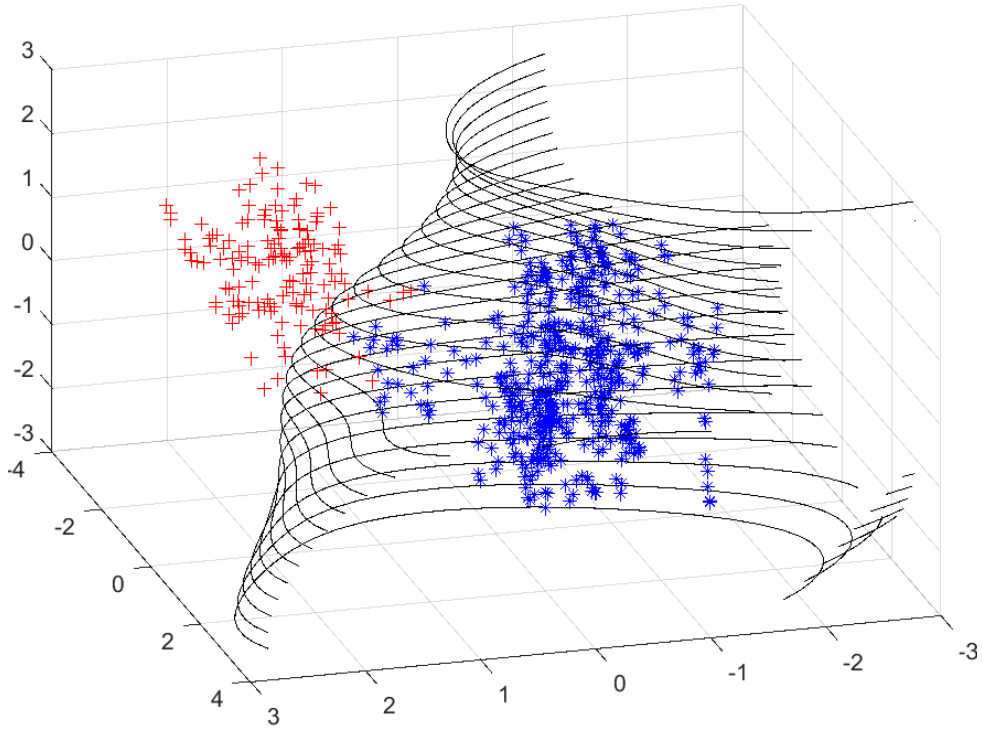


Figure 5.20: Target data observations (blue asterisks = NREM, red crosses = REM) in relation with source detector S_2 decision boundary (black lines).

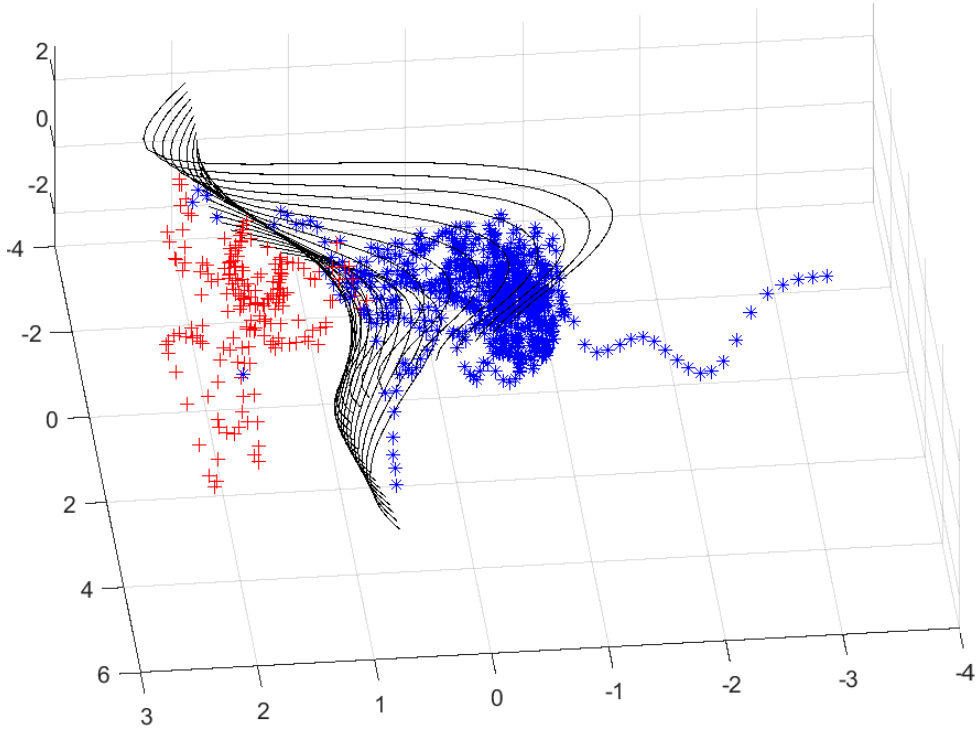


Figure 5.21: Source S_3 observations (blue asterisks = NREM, red crosses = REM) and associated source detector decision boundary (black lines).

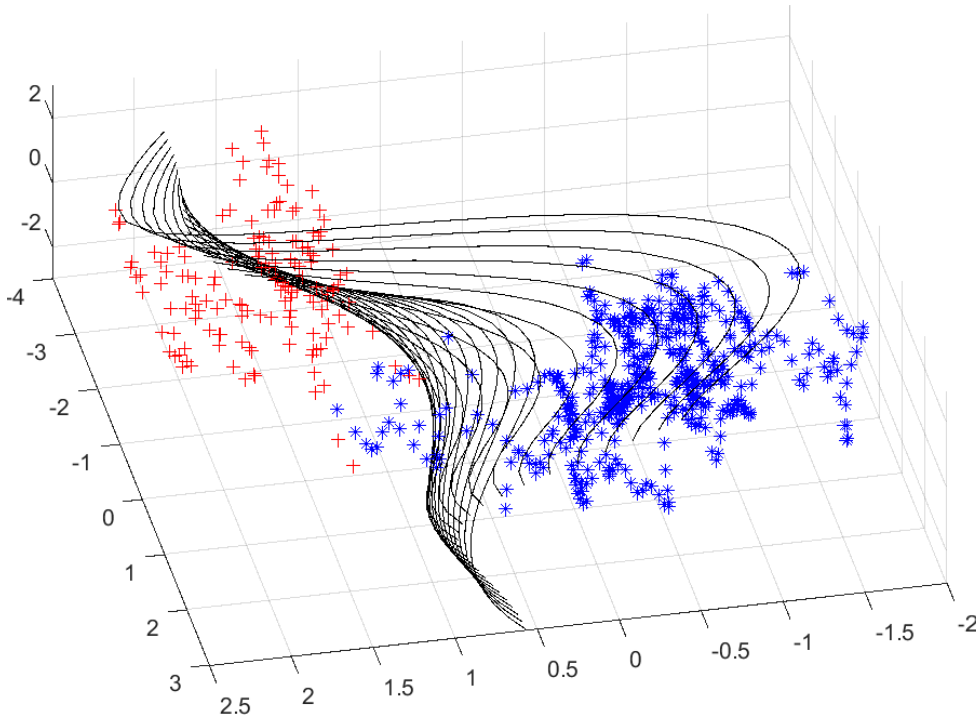


Figure 5.22: Target data observations (blue asterisks = NREM, red crosses = REM) in relation with source detector S_3 decision boundary (black lines).

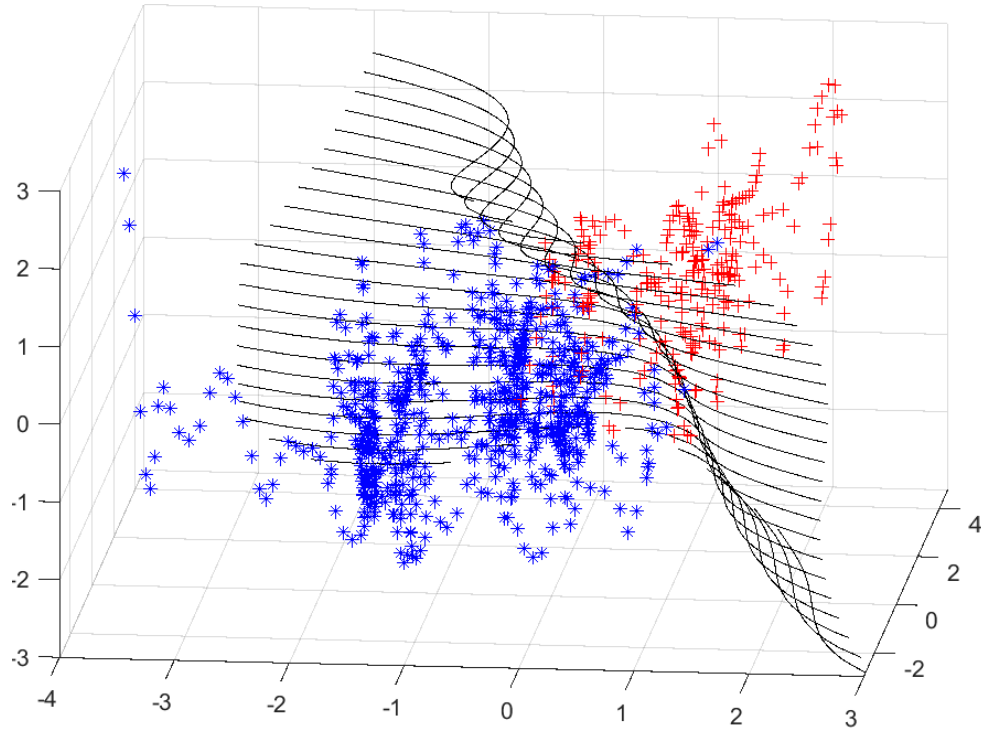


Figure 5.23: Source S_4 observations (blue asterisks = NREM, red crosses = REM) and associated source detector decision boundary (black lines).

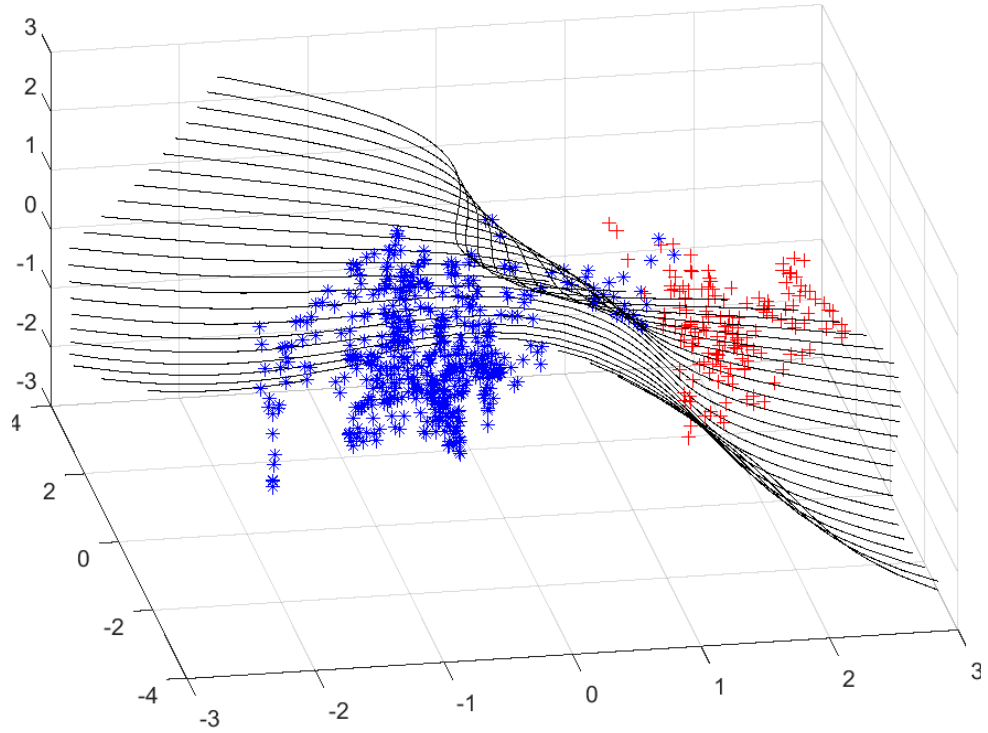


Figure 5.24: Target data observations (blue asterisks = NREM, red crosses = REM) in relation with source detector S_4 decision boundary (black lines).

Chapter 6

Results on the validation/target dataset

Contents

The previous chapter focuses on studying our methods results on a few recordings of interest (presenting solvable situations of data shift). It served mainly an illustrative purpose, although the resulting operational setup is adequate for most recordings.
In this last chapter, we consider the application of all methods on the validation dataset, to conclude on how effective our methods are on the application. We compare these results to the ones of a reference detector, giving the highest performances without transfer or source selection.
This final chapter organises as follows:

- In Section (6.1) we present the results of our non-iterative transfer methods applied to the whole target dataset. We also select our single-best detector without transfer as the reference detector, which helps us determine each method's effects on the application.
- Section (6.2) gives the results of optimised centring using the reference detector as a unique source. We did not consider other sources as computation-time is great, this method being iterative.
- Finally, Section (6.3) further illustrates (Training) Source Selection (TSS); we compare our predictors to the actual performances for each (source, target) couple, and comment on the observed results.

6.1 Immediate transfer learning results

In this section we present the results of all three methods of transferring knowledge (QLTL, KCATL and KTATL) in application on the whole RV/target dataset.
All computations are done using the following set-up:

- Gaussian Kernel parameter set to $\sigma = 2$ and resulting kernel matrix mean-centred for all methods: K_{TT_C} for QLTL and KTATL, K_{TS_C} for KCATL.

- (QLTL only) Compromise parameter set to $\lambda = 0.35$.

- Supervised information inputted as the non-binarised decision statistic outputted by the considered source S: on the target data Y_T^S for QLTL and KTATL, on its own data Y_S^S for KCATL.

Measures of performances are once again established in detection of REM against the N2 and N3 sleep stages; in this Chapter, we mainly consider the average detection rate at average false-alarm rate set to 7%, and will also consider each (source, target) couple performances to assess individual evolution of performances. We develop this latter point farther below.

6.1.1 Average results for each source detector

First, we considered each source recording as a single source, trained a detector on it and applied the three transfer methods (QLTL, KCATL and KTATL). We compute the average detection rate, while the average false-alarm rate has been fixed to 7%; as aforementioned, such a value is ensured during the validation step and for methods comparison purpose: the threshold of decision of each individual detector (without transfer and for each transfer method) has been selected so that the average false-alarm rate on RV is set to the same value, allowing performances comparison.
In Figure (6.1), results are sorted by increasing base detector performances (black asterisks). As can be seen, QLTL systematically results in a slight improvement (green stars), KCATL results in varying improvements while there is also a case of negative transfer (red pluses - see index 41 for a case of negative transfer) and KTATL improves results globally: the worse the base detector, the better the improvement (blue triangles).
On the average$\pm$standard-deviation on each source applied to every target, our methods resulted in an improvement of $5.28 \pm 4.88\%$ average detection rate for QLTL, $10.24 \pm 11.87\%$ for KCATL and $13.31 \pm 12.07\%$ for KTATL.

6.1.2 Reference performance

As to conclude on our method's performance and interest on our application, we require the consideration of a *reference detector without transfer*. To do so, we select the best performing source without transfer, which corresponds to the right-most asterisk in Figure (6.1). This is therefore considered the best we can achieve using a single source without the use of transfer/source selection on our application.
In Figure (6.2), we represent such reference detector ROC curve (black line) and the ones of each transfer method using this detector to input supervised information (three other lines). Visually, KCATL (red line with pluses) is slightly worse than the reference method, QLTL (green line with stars) is slightly better and KTATL (blue line with triangles) is better.
Now, quantitatively, the corresponding average operating points are:

- $7.00 \pm 5.71\%$ false-alarm rate for $81.73 \pm 15.75\%$ detection rate for the reference detector.

- $7.02 \pm 5.66\%$ FAR for $82.92 \pm 15.59\%$ DR using QLTL.

- $7.01 \pm 5.03\%$ FAR for $82.02 \pm 15.15\%$ DR using KCATL.

- $7.00 \pm 6.13\%$ FAR for $83.26 \pm 13.07\%$ DR using KTATL.

Such operating points are obtained by averaging over all individual operating points (application of a source detector to each given target): this is illustrated by the black asterisks and the bold one in Figure (6.2).
To conclude on the reference detector, all methods of transfer result in an improvement in average detection rate ($+1.19\%, +0.29\%$ and $+1.53\%$ for resp. QLTL, KCATL and QLTL); however, some other sources may result in yet better performances using transfer (e.g. see KTATL at indexes 23 and 46 in Figure(6.1)). Additionally, our objective is ultimately to ensure good classification performances on every target, which might not be the case even with acceptable average performances.

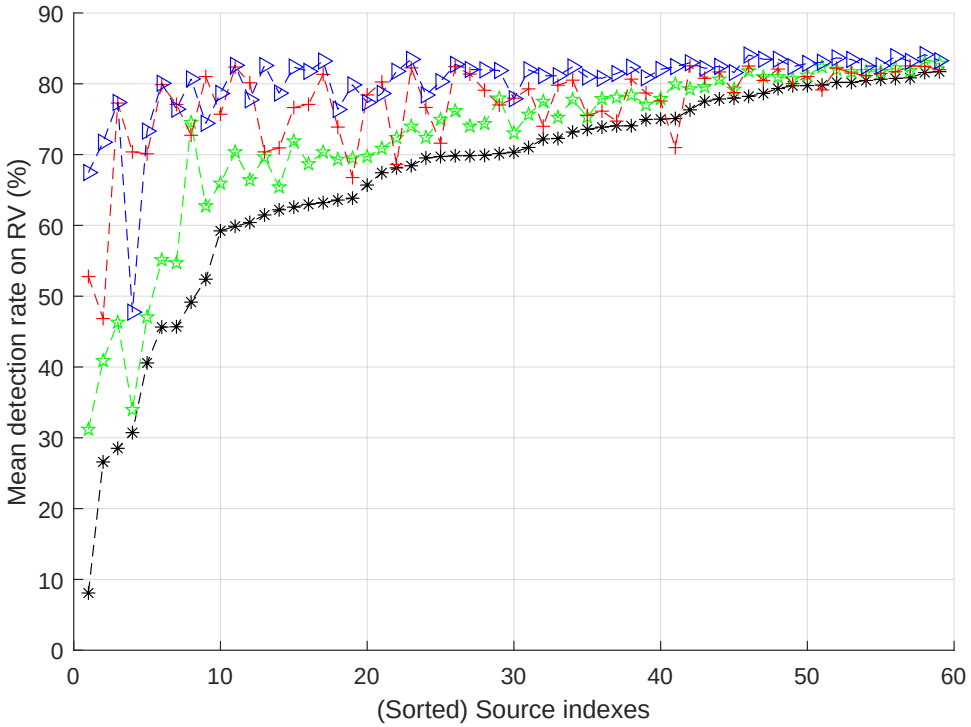


Figure 6.1: Mean detection rate for an average false-alarm rate fixed at 7% on the entire RV dataset; source indexes sorted by increasing reference performances (detector without transfer - black asterisks). Transfer method: QLTL (green stars); KCATL (red pluses); KTATL (blue triangles).

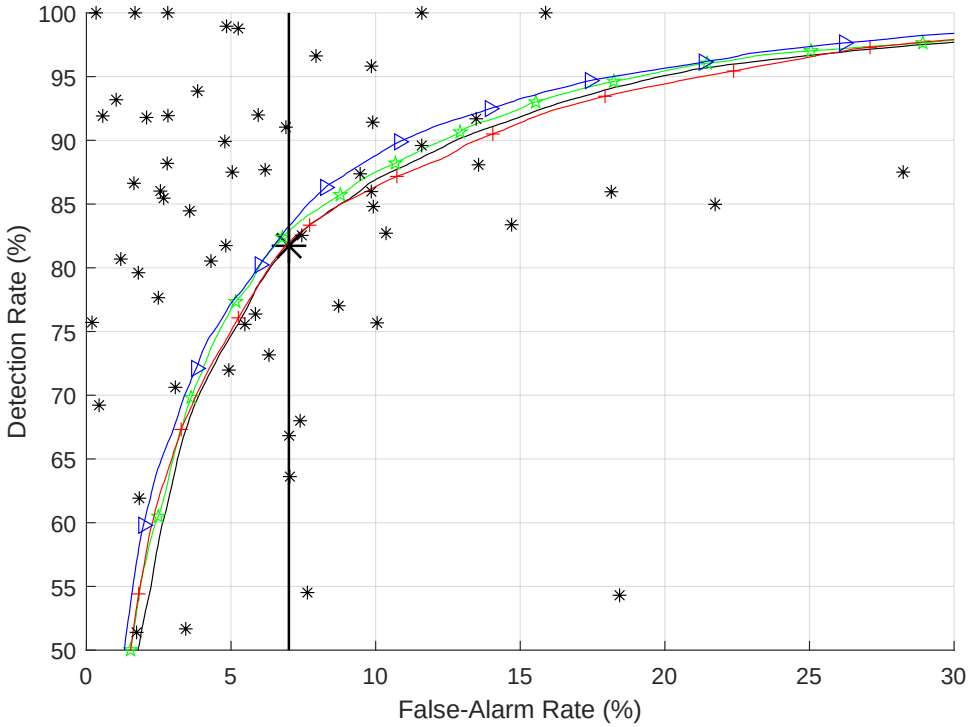


Figure 6.2: Selection of the best source without transfer as reference detector (right-most black asterisk in Figure (6.1).) Corresponding ROC curve (black line) and results of each transfer method (QLTL: green curve with stars, KCATL: red curve with pluses, KTATL: blue curve with triangles). Black asterisks: operating point for each target; bold one: average operating point on all targets.

6.1.3 Reliability of transfer

While average results give an idea of the global behaviour of each method, we would rather have reliable results that will perform similarly on any target, than highly variable/unpredictable performances.
To assess that, let us have a look at Figures (6.3 and 6.4), where for a given source recording, and for each method - base SVM detector and each of the three transfer methods, we represent the detection rate at false-alarm rate set to 7% and the distance with the *perfect* operating point, respectively.
The *perfect* operating point is the one of coordinates $[0, 1]$, which corresponds to perfect detection with no false-alarm. By assessing the distance between this point and the actual operating points for each target, we have a measure of performance that takes into account the actual performances; in other words:

- Figure (6.3) displays non-actual performances that could only be attained if we knew how to select the decision threshold adequately for each target individually. It mainly helps to assess how well a given method transfers knowledge.
- Figure (6.4) gives a representation closer to the actual performances. It also gives an idea on how a given method affects the input decision statistic.

Target indexes have been sorted in both figures, to allow to consider both metrics for each target. We make the following commentaries and interpretations:

- For most targets, the transfer has little impact on the DR at FAR=7%, and only slightly affects the distance to the perfect OP.
- In some situations, KTATL positively transfers knowledge (Figure (6.3)see indexes 6, 8, 9, 11, 12, 26, among others). For some, this also leads to an improved actual operating point (Figure (6.4): e.g. indexes 8, 9, 12, 18, 26).
- There are a few cases of negative transfer for all transfer methods, e.g. for target indexes 10 and 33. This is arguably due to the kernel-matrices eigenvectors/singular-vectors being non-discriminative of the REM sleep stage, for these recordings.
- Indexes 1 and 2 correspond to problematic recordings whose REM may not be discriminated using the proposed features. However, transferring knowledge helps a little in the case of index 2, and helps a lot in the case of index 3, leading in the latter case to acceptable performances.

6.1.4 Partial conclusion

Following the previous presentations, we conclude on our direct transfer methods:

- All methods result, in most cases, on improved performances on the average.
- Considering the single best - reference - source detector, transferring knowledge leads to little effects in most cases, and slight-to-good improvements for a few recordings. Cases of negative transfer are rare.
- Some improvements may not be immediate, and due to modified decision statistics, may require the selection of an adequate decision threshold for some targets. This statement requires further investigations.
- Overall, KTATL performs the best over KCATL, which may lead to unreliable results, and over QLTL, which leads to smaller improvements.

6.2 Optimised centring

We consider KCATL with optimised centring using:

- the reference detector/recording as sole input source (these iterative methods are quite time-consuming).
- all six centring methods (as presented and developed in Section (5.3)).

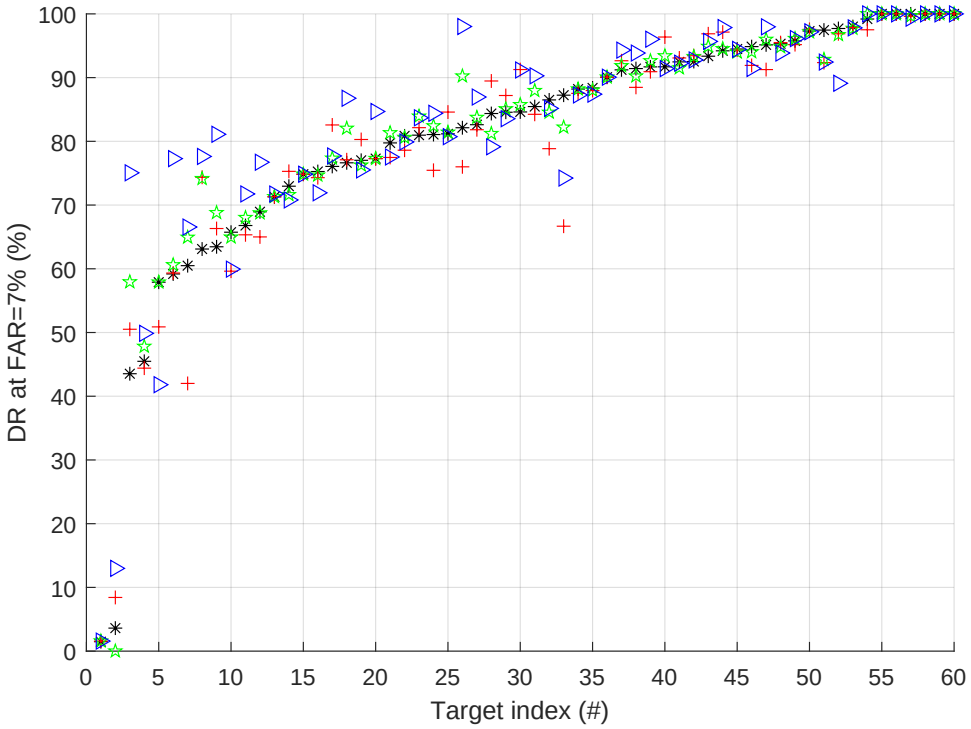


Figure 6.3: DR at FAR set to 7% for each target using the reference detector. Target indexes sorted by increasing reference performances. Black asterisks: performances without transfer; green stars, red pluses, blue triangles: results of QLTL, KCATL and KTATL, respectively.

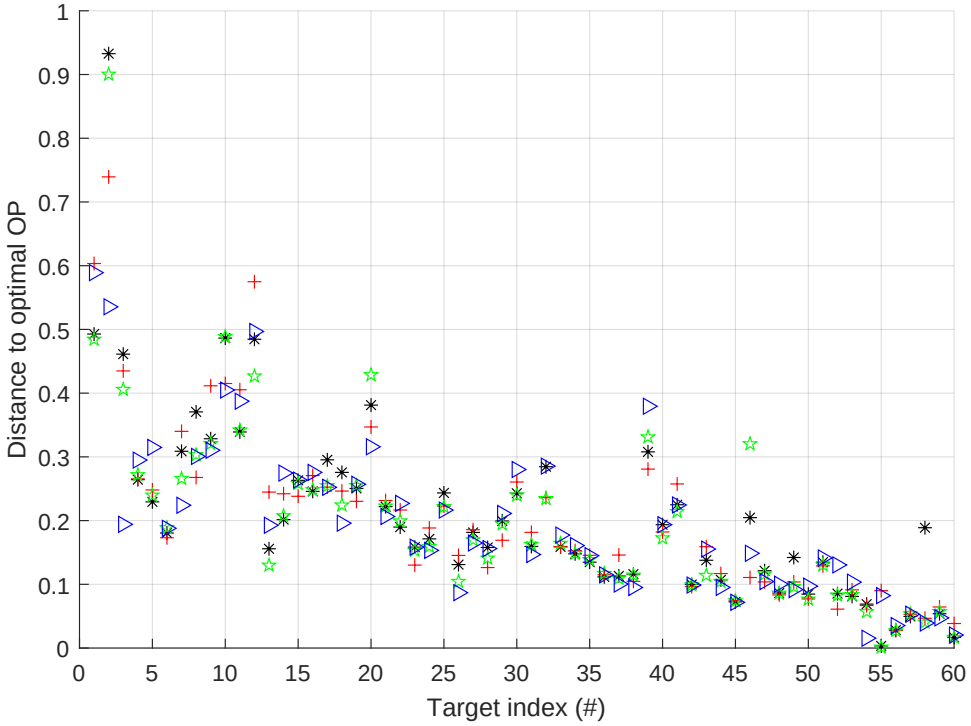


Figure 6.4: Distance with the *perfect* operating point for each actual OP. Target indexes sorted as in Figure (6.3). Black asterisks: performances without transfer; green stars, red pluses, blue triangles: results of QLTL, KCATL and KTATL, respectively.

- the same operational setup as previously: $\sigma = 2$, initialisation as mean centres (be it in the RKHS or in input space), $\eta = 10^{-2}, \Delta_S = 10^{-3}$ when maximising KCA and $\eta = 10^{-6}, \Delta_S = 10^{1}$ when maximising KCP.

The resulting ROC curves are represented in Figure (6.5). The corresponding operating points are as follows:

- $7.00 \pm 5.58\%$ false-alarm rate for $72.89 \pm 25.16\%$ detection rate for maximising KCA w.r.t. both centres in input space.
- $7.03 \pm 5.51\%$ FAR for $77.71 \pm 19.21\%$ DR for maximising KCA w.r.t. the source centre in input space.
- $7.02 \pm 5.11\%$ FAR for $82.01 \pm 15.14\%$ DR for maximising KCA w.r.t. both centres in the RKHS.
- $7.04 \pm 5.03\%$ FAR for $82.07 \pm 15.16\%$ DR for maximising KCA w.r.t. the source centre in the RKHS.
- $7.03 \pm 4.53\%$ FAR for $77.45 \pm 20.14\%$ DR for maximising KCP w.r.t. both centres in input space.
- $7.02 \pm 6.10\%$ FAR for $82.47 \pm 15.48\%$ DR for maximising KCP w.r.t. the source centre in input space.

We observe the following:

- On the considered configuration, only the maximisation of KCP w.r.t. the source centre in input space leads to a marginal improvement of the ROC curve/performances.
- All methods lead to an increased dispersion in the detection rate.
- Maximising KCA w.r.t both centres or the source centre only in the RKHS lead to almost identical results. This was observed in the previous chapter as well.

We therefore conclude that, due to the non-iterative nature of this method, to the sub-adequate results we observe and to the increased dispersion in the performances, KCATL with optimised centring is inadequate for our application as it is.
However, as it leads to improved transfer of knowledge in a few situations, it may be pursued in the future; there are yet some aspects of the methods to be investigated and improved.

6.3 Training source selection results

Training source selection does not come with this threshold of decision issue, as the output of a given (source, target) couple is directly the one of a reference detector. For each (source RL, target RV) couple, we compute our most efficient predictors.
In Figures (6.6 and 6.7) we represent two of these metrics, respectively:

- The correlation between the decision outputted by the source and the target $sign(Y_T^S)$, and the one obtained when multiplying the decision by the kernel-target matrix $K_{TT_C} sign(Y_T^S)$.
- The observed kernel-target alignment, computed using K_{TT_C} and the non-binarised output of each source detector Y_T^S.

All predictors - including these two - have been computed using mean-centred kernel-matrices of parameter $\sigma = 2$, and when possible, non-binarised outputs of the source detector. Then after, in Figures (6.8 and 6.9), we represent respectively for each (source, target) couple:

- The detection rate at FAR=7% (as aforementioned, this would require to adapt the decision threshold for each couple).
- The distance of the current operating point to the one of perfect detection (0% FAR, 100% DR).

Through analysis and comparison of these four figures, we notice the following:

- There are underperforming sources which achieve poor performances in application on any target recording. These are easily recognisable as outlying horizontal patterns in Figures (6.8 and 6.9). These include sources at indexes 10, 30, 44 and 59, among others.

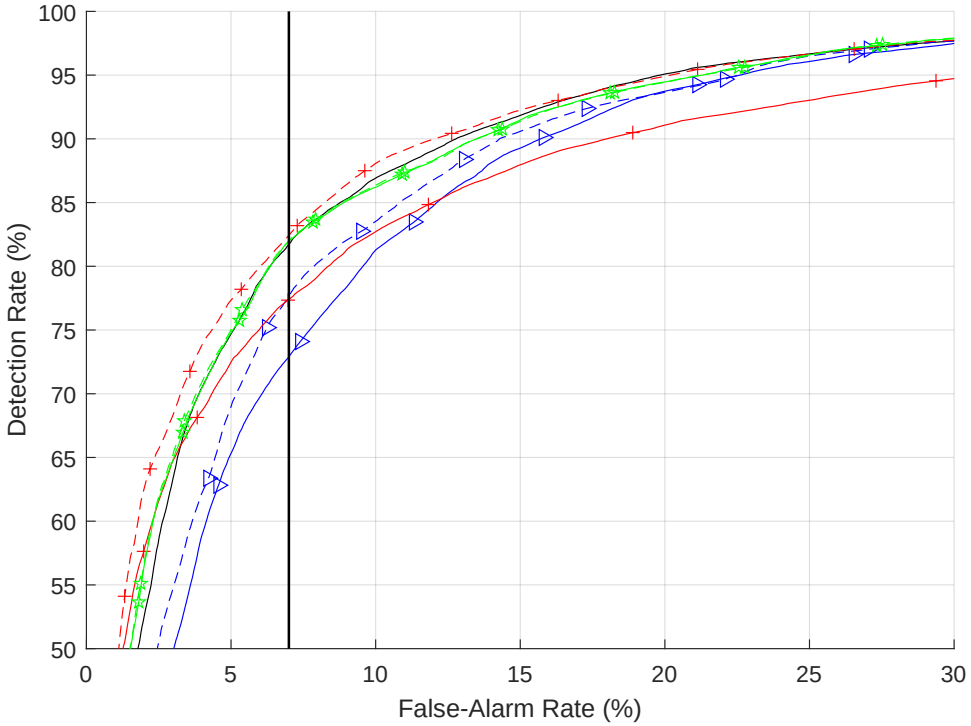


Figure 6.5: ROC Curves of KCATL with optimised centres; the black curve is the reference detector without transfer. Blue curves with triangles, red curves with pluses, green curves with stars correspond to the optimisation of KCA in input space, of KCP in input space and of KCA in the RKHS, respectively. Dotted curves correspond to maximisation w.r.t. the source centre only while plain curves to the one w.r.t. both centres.

- There are difficult targets on which most (if not all) sources perform poorly. Likewise, these may be recognised by their vertical outlying pattern. See, for instance, targets of indexes 20 and 24.

- There is a good correlation between the predictors and the actual performances, as can be observed by comparing Figures (6.6 and 6.7) and Figures (6.8 and 6.9). The higher the correlation, the better the selection is expected to work.

- Underperforming sources are well-detected using the observed correlation as predictor (colder lines at indexes 10, 30, 44, 59). It may perform better than the observed KTA, which fails to assess these sources as underperforming.

In the remainder of this final section, we first display the results of training source selection in selecting one source among all available sources; in a second subsection, we consider an improved sub-selection of sources to choose amongst; in a third subsection, we consider multiple sources instead of just one; in a final subsection, we give our conclusions on TSS.

6.3.1 Selecting one source among all sources

In this subsection, we select one source for each target using each of our predictors; the selected source is the one maximising the considered predictor. Average results are:

- $7.02 \pm 4.93\%$ false-alarm rate for $80.06 \pm 16.12\%$ detection rate using the observed KTA as predictor.

- $7.03 \pm 5.24\%$ FAR for $81.84 \pm 15.22\%$ DR using QLTL using the observed KCA.

- $7.04 \pm 5.47\%$ FAR for $82.73 \pm 13.68\%$ DR using the observed correlation with and without KTATL.

- $7.01 \pm 5.39\%$ FAR for $80.15 \pm 15.00\%$ DR using the observed correlation with and without KCATL.

In this configuration, only the observed correlation between decision without transfer and decision using KCATL leads to improved performances over the reference detector.
In Figure (6.10), we represent the frequency at which each source is selected for each of the predictors. As can be seen, observed KCA and observed KTA have low diversity in their selections (selecting mainly sources of indexes 29 and 14, respectively), while their correlation counterparts result in more diverse selections. This piece of information may help to constitute an adequate ensemble of sources for each predictor, and its exploitation is under investigation.

6.3.2 Fusion of multiple selections

In some situations, the selected detector may be inadequate for the given target. This is especially true when dealing with prior shifts, to which our predictors are insensitive. In order to improve reliability in our selection, we combine the output of the k detectors that maximise the considered predictor.
We simply sum the output of the k detectors, weighed by the value of the predictor, and normalise by the sum of the predictor values; here are the results for $k = 5$:

- $7.02 \pm 5.41\%$ false-alarm rate for $81.37 \pm 15.06\%$ detection rate using the observed KTA as predictor.

- $7.02 \pm 5.33\%$ FAR for $82.01 \pm 15.29\%$ DR using QLTL using the observed KCA.

- $7.02 \pm 5.80\%$ FAR for $83.03 \pm 13.93\%$ DR using the observed correlation with and without KTATL.

- $7.01 \pm 5.62\%$ FAR for $81.73 \pm 14.87\%$ DR using the observed correlation with and without KCATL.

When comparing with the results of the previous subsection, for which $k = 1$, this leads to a slight improvement in detection performances. The choice of an optimal value for k can be done through validation.

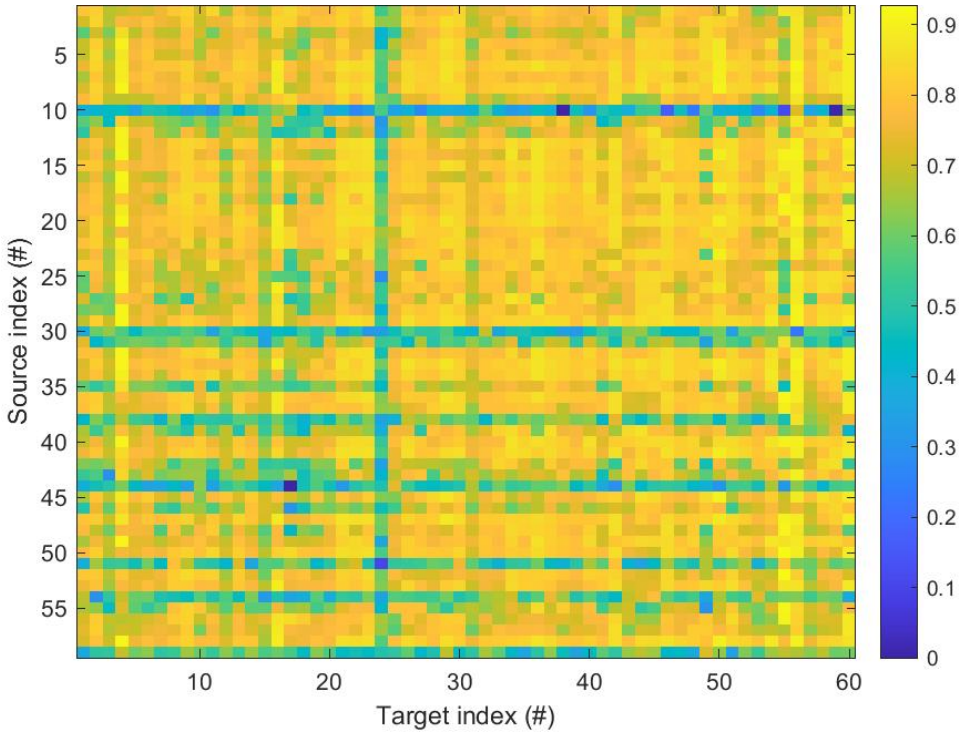


Figure 6.6: Prediction value for each (source, target) couple of recordings. The performance predictor is here the correlation between source detector output and KTATL.

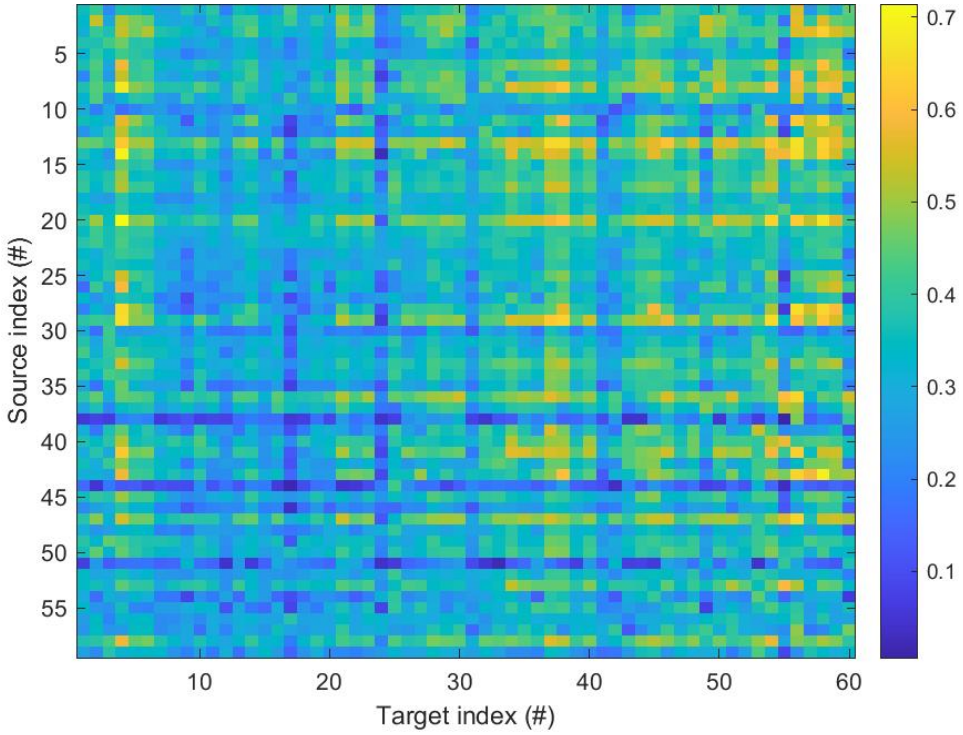


Figure 6.7: Prediction value for each (source, target) couple of recordings. The performance predictor is here the observed kernel-target alignment.

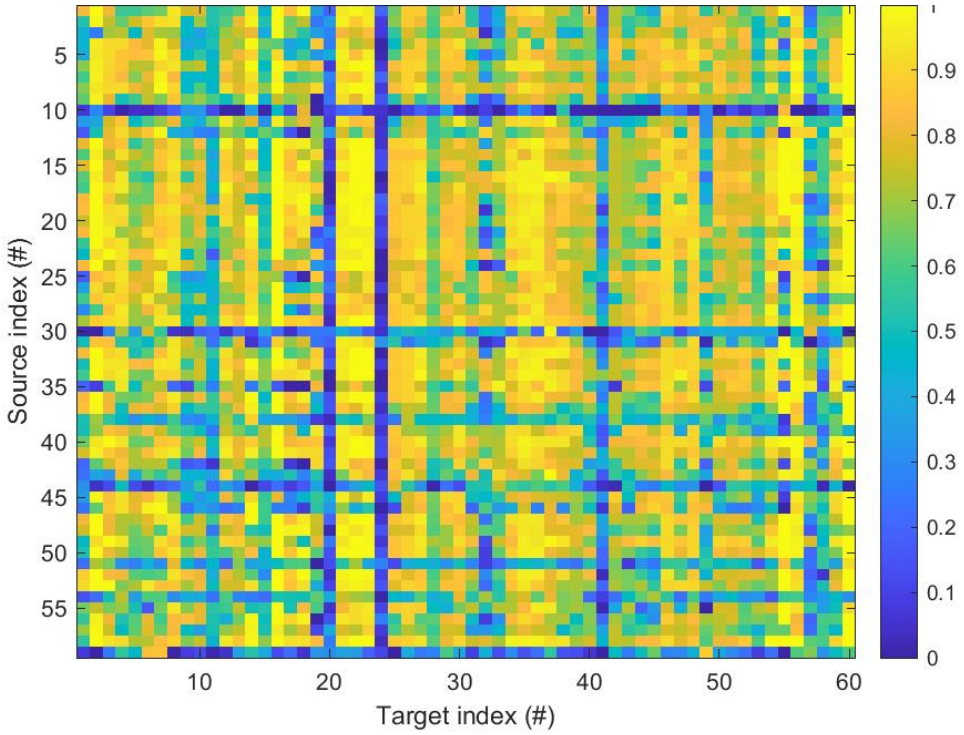


Figure 6.8: Actual performances in application of each source to each target; the measure represented here is the detection rate at false-alarm rate set to 7%. The hotter the better.

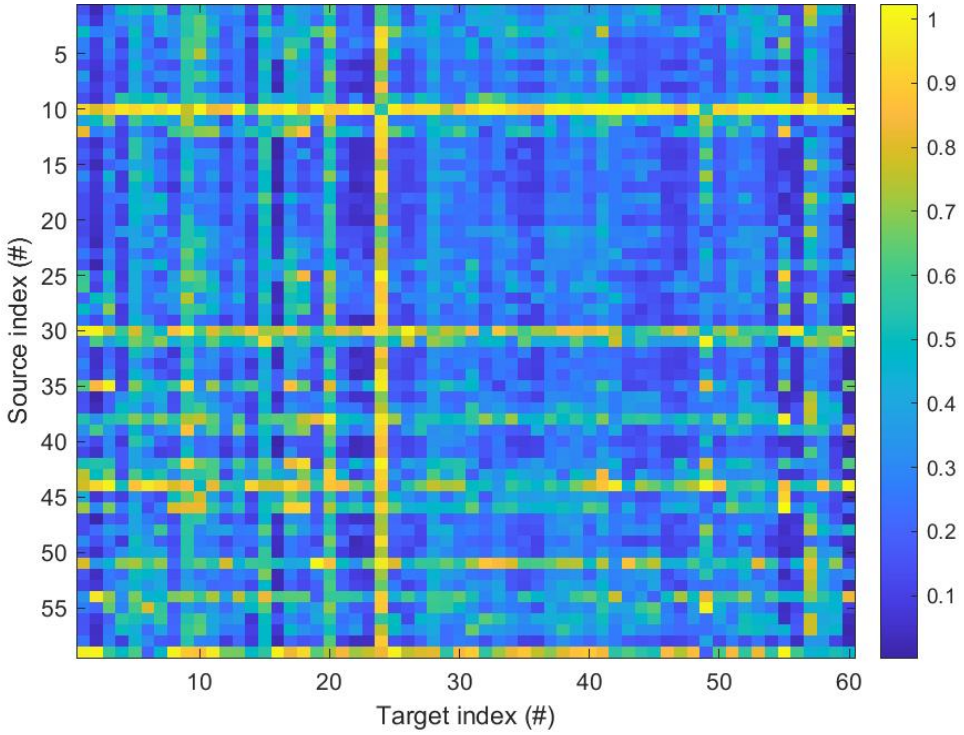


Figure 6.9: Actual performances in application of each source to each target; the measure represented here is the distance between the actual operating point and the one corresponding to perfect detection. The colder the better.

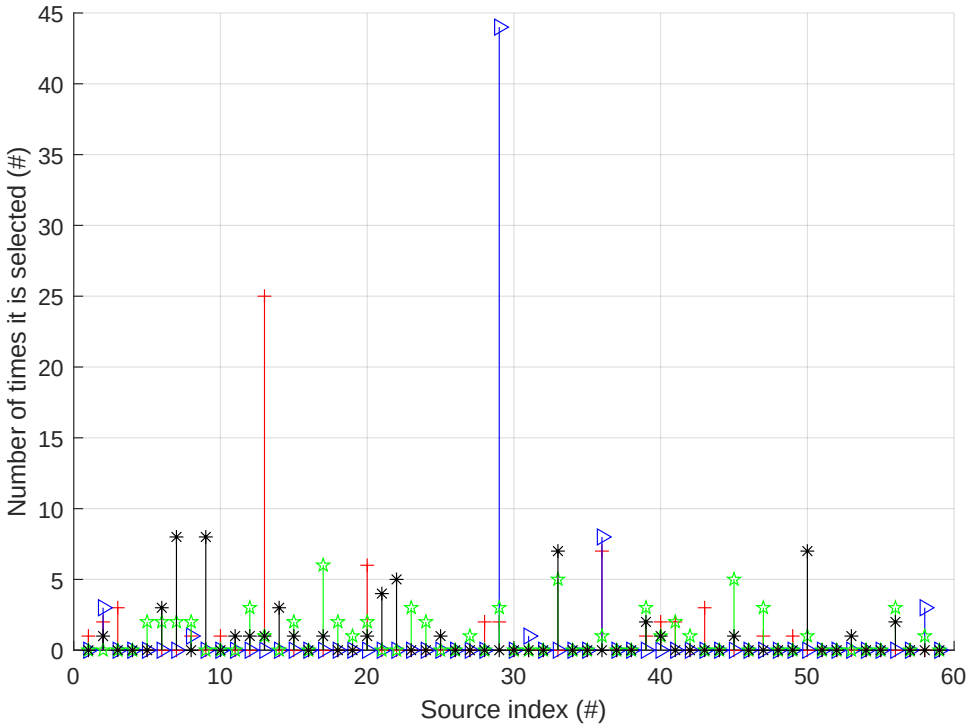


Figure 6.10: Number of times each source gets selected over all targets. Black asterisks, green stars, blue triangles, and red pluses respectively correspond to correlation w/wo KCATL, correlation w/wo KTATL, observed KCA and observed KTA predictors.

6.3.3 Improvement by sources sub-selection

One of the key elements of TSS is the constitution of a good ensemble of sources from which to choose from. In our application/example, it is ill-advised to consider all sources for selection, as some sources are systematically underperforming and as there is a certain amount of redundancy in the sources.
Here we restrain our ensemble to the top ten performing sources, and use again $k = 5$:

- $7.03 \pm 5.42\%$ false-alarm rate for $82.28 \pm 16.11\%$ detection rate using the observed KTA as predictor.
- $7.02 \pm 5.27\%$ FAR for $82.24 \pm 15.93\%$ DR using QLTL using the observed KCA.
- $7.02 \pm 5.60\%$ FAR for $82.71 \pm 15.73\%$ DR using the observed correlation with and without KTATL.
- $7.02 \pm 5.49\%$ FAR for $82.92 \pm 15.20\%$ DR using the observed correlation with and without KCATL.

Such restrained ensemble of sources improves the results of all predictors except the observed correlation w/wo KTATL, which is slightly worsened.
We considered other methods of selecting sources for an efficient ensemble; for instance, to consider for each source how its predictors correlate to the actual performances. In the end, the method resulting in the best selections is the Sequential Floating Forward Selection (SFFS): we select sources sequentially by retaining at each step the one maximising performances (in TSS) using the ones previously selected. Corresponding results are:

- $7.01 \pm 5.69\%$ false-alarm rate for $81.99 \pm 14.69\%$ detection rate using the observed KTA as predictor.
- $7.03 \pm 5.41\%$ FAR for $82.55 \pm 14.17\%$ DR using QLTL using the observed KCA.
- $7.02 \pm 5.94\%$ FAR for $83.43 \pm 13.77\%$ DR using the observed correlation with and without KTATL.
- $7.02 \pm 5.80\%$ FAR for $82.98 \pm 14.08\%$ DR using the observed correlation with and without KCATL.

This method indeed creates a good selection of sources, but is most likely not optimal.

6.3.4 Partial conclusion

Training source selection seems effective in selecting adequate source detectors for each target, and critically depends on:

- The performances predictor used for the selection. In our experiments, the observed correlation with and without application of KTATL seem to correlate the most with the actual performances, and overall results in the best performances in selection.
- The ensemble of sources among which to select from. This is a delicate part of the setup, as an optimal selection of sources should both be diversified, as to make the selection interesting; but should also contain predictable sources to ensure an adequate source may be selected.
- The way the selection is done, that is for instance, our proposed merging of the output of the k sources maximising the considered predictor. Such merging is naive, but yet improves the performances on the application. Alternatives have been considered and are still under investigation.

There is a good amount of perspectives to source selection, which may require further researches:

- An extension to a heterogeneous ensemble of detectors, that is, established using different sets of features. Technically, our predictors only use the output of detectors, which may be of any kind and based on any set of features; also, any similarity matrix may replace the kernel-matrix in our metrics computations. The issue is that we do not know yet how such modifications may affect the reliability of the predictions.
- Further, it may be interesting to use source selection to also determine whether or not to use transfer learning for a given (source, target) couple.

Conclusions and perspectives

This doctoral project objective is the improvement of sleep staging classifiers through palliation of the observed data shifts that may exist between two given recordings. We give a quick summary of this manuscript content below:

- In Chapter (1) we have introduced the application, the associated data and have illustrated the data shift issue using two sleep recordings examples. We explained how data shift negatively affects our classifiers performances, therefore introducing the subject of our research: how to palliate variability-induced data shift.
- In Chapter (2) we have presented the methodological background to our researches: transfer learning, domain/decision adaptation, and kernel methods.
- In Chapter (3), we explained and developed on the different, innovative, methods that resulted from our researches: three non-iterative methods of transferring knowledge (QLTL, KCATL, KTATL), one iterative method of kernel-centring (KCATL through a centred cross-Gram matrix, where the centres have been optimised to maximise KCP/KCA), and multiple performances predictors so to enable source selection (TSS).
- In Chapter (4, we gave further illustrations of some recurring issues in REM/NREM detection, through the analysis of kernel matrices. We also illustrated the discriminative information contained in these matrices, the key component of our transfer methods.
- Chapter (5) focused on the illustration of our methods in application on a few recordings of interest; these helped to better choose our methods parameters. While such recordings may not be representative of all recordings, they constitute good examples of the type of shift situation we ought to solve for.
- Finally, in Chapter (6), we considered our methods application on the whole validation/target dataset. This allowed us to compare performances with the ones of the current detector (without transfer), and to better assess the methods benefits and limitations.

After such presentations, we are now able to conclude on the results of this doctoral project, which may be seen through different points of view:

Methodological contributions

Regarding our methodological contributions to transfer learning:

- QLTL and KTATL both give consistent results and usually improve a detector output. This is due to the unsupervised information contained in the target kernel matrix, which plays a crucial role in efficiently transferring knowledge.
- KTATL is more direct than QLTL, and usually performs better, while also requiring the consideration of one less parameter (λ, QLTL's compromise parameter). While this is true for our application and toy examples, there may be unconsidered situations where QLTL is more adequate.
- KCATL is less consistent, as it highly depends on the source data distribution properties, being used in the computation of the cross-Gram matrix and therefore in KCATL output.

Type	Title	Conf./Journal	Status
Conference	QLTL: a Simple yet Efficient Algorithm for Semi-Supervised Transfer Learning	ICPRS2019	Published(2019)
Conference	Influence of Data Centring in Kernel-Cross Alignment: Application to Transfer Learning	ICPRS2021	Published(2021)
Article	Cross-Gram Matrices and their use in Transfer Learning: Application to Automatic REM Detection using Heart-Rate	Computer Methods and Programs in Biomedicine (CMPB)	Published(2021)
Patent	A computer-implemented method of selecting a preferred training dataset from a plurality of training datasets for a target dataset	European Patent Office (EPO)	Filed

Table 1: List of scientific communications achieved during the doctoral project.

- The current main limitation of the three methods is the selection of an adequate decision threshold: Gram matrices dynamic directly affects the methods decision statistics. This may lead to more dispersed classification performances.

Regarding our work around the optimal centring of the cross-Gram matrices, it led to interesting results: the strong improvement of KCATL classification performances in numerous setups. However, in many other cases, it resulted in negative transfer, and sometimes led to absurd choices of centres. Current results are too unpredictable to consider optimised centring KCATL as a good method of transfer; methods to improve the problem, by addition of constraints and by the improvement of initial conditions, are under investigation.
Finally, source selection is the most interesting contribution we suggest: contrary to our transfer methods, it gives a preview of the expected performances of the application of a source detector to a target, with high reliability. It also has the advantage of not requiring adaptation of the decision threshold, the one obtained through training and validation being adequate in most cases. This last method yet requires fine-tuning, i.e. selection of an optimal subset of sources from which to select from, improvement of our predictors prediction capabilities, and so on, but primary results are more than encouraging.
Additionally, one may want to consider the situation where no source is adequate for a given target: we may want to combine training source selection and transfer learning, allowing not only to predict performances, but also to transfer knowledge when needed. This last proposed classification framework requires further research and is also being investigated.
The main perspective of our work is to further investigate the causes of negative transfer for all methods (TL-based and TSS), for the few situations when it happens. Moreover, for each of our methods, we ought to investigate and characterise which data properties allow an efficient transfer of knowledge/a successful selection. Such association would help better understand how data properties relate to our methods performance. Regarding QLTL, perspectives also include the pursuit of Bregman divergences, and how the asymmetry property affects its results.

Scientific communications

The doctoral project resulted in two presentations in international conferences (International Conference on Pattern Recognition Systems - ICPRS), in one submission to an international journal (Computer Methods and Programs in Biomedicine) and in the filing of one patent at the European Patent Office. These are summarised in Table (1). The second conference presentation has been awarded the "best student paper" award.
The two presentations focused on our transfer methods and their operational setup, the journal paper focuses on their application in REM detection, and the patent protects the source selection method.

Gains on the application

We consider our single best source classifier as a reference detector without transfer and without selection; such detector achieves an average operating point of $7.00 \pm 5.71\%$ false-alarm rate for $81.73 \pm 15.75\%$ detector rate on the target dataset.
Using our best transfer method (KTATL) in an optimised operational setup (best source for knowledged transfer and

associated parameters), we achieve an average operating point of $7.01 \pm 5.86\%$ false-alarm rate for $84.14 \pm 12.56\%$ detection rate.
Finally, using our best setup for source selection, we achieve: $7.02 \pm 5.94\%$ FAR for $83.43 \pm 13.77\%$ DR.
Both the transfer and the source selection methods resulted in improvements on the application.

Appendix A

Résumé substantiel en français

Introduction

Ce projet doctoral a pour objet l'amélioration d'un outil de classification automatique des stades du sommeil par la prise en compte des variabilités inter-et-intra-individuelles des données. Ces dernières engendrent des changements dans la loi jointe des données et de leurs étiquettes associées $p(X, Y)$ − nous les qualifierons de *décalages des données* − ce qui résulte en des performances de classification réduites sur certains enregistrements.
Ce résumé reprend les six chapitres du document principal: une présentation de l'application, ses enjeux et les limitations observées; une introduction aux concepts essentiels de l'état-de-l'art, principalement basés sur le "transfert d'apprentissage" et les méthodes à noyau; le développement de nos contributions méthodologiques, deux méthodes de transfert d'apprentissage (QLTL, KxATL), un algorithme d'optimisation du centrage de matrices noyau et une méthode de sélection de "sources"; quelques observations réalisées sur les données sommeil d'éléments pouvant compliquer la tâche de détection du sommeil paradoxal; des résultats en application sur quelques enregistrements servant d'exemples, puis finalement les résultats en application sur l'ensemble complet de validation, permettant de juger des apports de nos méthodes à l'application.
En guise de conclusion, nous récapitulerons les résultats et perspectives de nos travaux.

A.1 L'application

Dans cette première section, nous introduisons succinctement l'analyse du sommeil, les alternatives à l'examen de référence et Somno-Art, l'application développée par PPRS, l'entreprise d'accueil. Nous présentons finalement les données utilisées au sein de cette étude, et illustrons le problème de décalage des données au travers d'un exemple.

A.1.1 L'étude du sommeil

La classification automatique des stades de sommeil se fait usuellement au travers d'une polysomnographie (PSG) [1], un examen lourd et coûteux généralement réalisé en milieu hospitalier ou dans un centre dédié d'étude du sommeil. Une PSG est l'acquisition simultanée:

- De multiples dérivations électroencéphalographiques (EEGs), servant au suivi de l'activité du cortex cérébral
- D'un ou deux électrooculogrammes (EOGs), au suivi des mouvements oculaires
- D'un électromyogramme (EMG) pour le tonus musculaire
- D'un électrocardiogramme (ECG)
- De mesures supplémentaires éventuelles

Un expert du sommeil ou personnel entrainé peut alors annoter la PSG en découpant des segments de trente secondes d'enregistrement et en y associant l'un des états suivants:

- L'éveil, dénoté W, pour *Wakefulness*.
- Les stades de sommeil N1 - un stade transitoire - N2 - stade de sommeil léger - et N3 - stade de sommeil profond dit "à ondes lentes".
- Le stade de sommeil paradoxal REM (pour *Rapid-Eye Movement*), état caractérisé par des mouvements oculaires rapides.

L'examen résulte en l'établissement d'un hypnogramme, représentation temporelle des stades de sommeil, duquel sont extraits des paramètres sommeil d'intérêt permettant l'établissement éventuel d'un diagnostique: latence d'endormissement, durée cumulée de chaque stade de sommeil, etc.

A.1.2 Les alternatives à la PSG et la Solution Somno-Art

De nombreuses recherches ont étudié le lien entre Système Nerveux Central (CNS), sur lequel se base la polysomnographie, et Système Nerveux Autonome (ANS). L'ANS pilote la balance sympathique/parasympathique, ou sympathovagale. Cette dernière présente l'avantage de pouvoir être estimée à partir du rythme cardiaque [6], dont la mesure est facile: patch au torse, mesure photoplethysmographiques (PPG)[7, 8], estimation par ballistocardiographie (BCG) [9], etc.
De ce fait, de nombreuses alternatives à la PSG ont été proposées pour leur praticité, réduisant l'examen au simple port d'un bandeau, d'un brassard, ou encore la mise en place de l'appareil de mesure sous le matelas; outre la réduction des contraintes liées à la PSG (nombreuses électrodes), cela permet également de réaliser l'examen hors milieu médical spécialisé.
La Solution Somno-Art [12] s'inscrit dans ces alternatives à la PSG, étant composée d'un dispositif (Somno-Art Device - un brassard qu'il suffit de porter à l'avant-bras) mesurant le pouls par PPG et les mouvements par accélérométrie, et d'un algorithme d'analyse (Somno-Art Software), basé principalement sur le traitement du signal et l'apprentissage automatique, permettant d'associer automatiquement et rapidement les mesures observées aux stades de sommeil (étiquettes recherchées).
A titre illustratif, nous représentons dans la Figure (A.1) quelques features extraits du rythme cardiaque et permettant ici de discriminer le sommeil paradoxal du reste.

A.1.3 Les données de l'application

Nous travaillons avec:

- Un ensemble d'apprentissage composé de 59 enregistrements (ou "nuits") effectués sur 36 sujets. Dans la suite, nous considérons chacun de ces enregistrements comme des "sources" au sens du transfert d'apprentissage (TL).
- Un ensemble de validation composé de 60 nuits provenant de 32 sujets. Ce second ensemble constitue nos "cibles", encore une fois au sens du TL.
- Un ensemble de test composé de 56 enregistrements venant de 30 sujets. Ce dernier ensemble n'est considéré que pour l'évaluation finale des performances, et n'est donc pas utilisé au cours de nos développements.

Ces nuits d'enregistrement proviennent de sujets aux caractéristiques variées (age, sexe, pathologie éventuelle, etc.), assurant une grande variabilité inter-individuelle. La plupart de ces sujets ont été suivis au cours de plusieurs nuits, assurant également une variabilité intra-individuelle élevée (effets éventuels de la fatigue, du stress, de la prise de médicaments, etc.).
Bien que la majorité des enregistrements présente des propriétés statistiques similaires, permettant l'établissement de classificateurs des stades de sommeil efficaces en moyenne, ces variabilités engendrent des décalages des données qui entraînent, pour certaines nuits, des performances de classification réduites.
Nous illustrons ce propos au travers de l'exemple d'un couple composé d'une source - Figure (A.2) - et d'une cible - Figure (A.3) - présentant un tel décalage des données. En effet, les observations correspondant au sommeil paradoxal (carrés rouges) ont bougé et le détecteur appris sur la source ne sera pas efficace en détection sur cette cible.

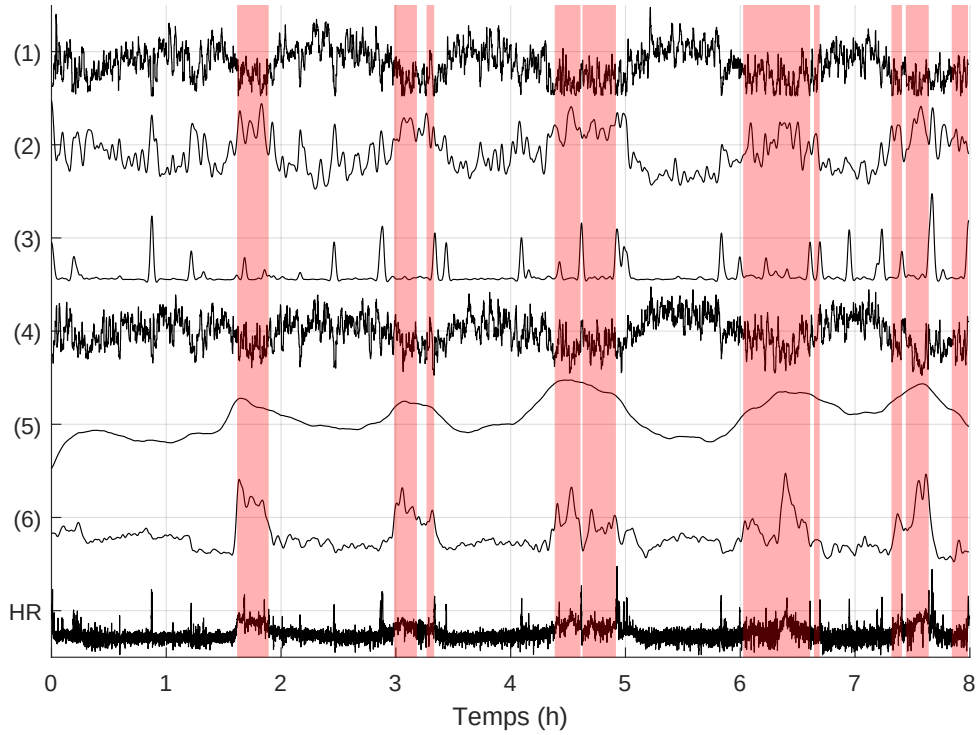


Figure A.1: Six features permettant de discriminer le sommeil paradoxal, ainsi que le rythme cardiaque (signal le plus en bas). Les zones en rouge correspondent au sommeil paradoxal. Abscisse: temps, en heures.

A.1.4 Le projet doctoral

Ce projet doctoral a pour objectif l'établissement de solutions permettant de pallier le problème de décalage des données illustré au travers des Figures (A.2 et A.3). Pour ce faire, nous avons dans un premier temps focalisé nos recherches sur les méthodes de transfert d'apprentissage, et sur celles de sélection de sources dans un second temps.
Cependant, au vu du nombre important de données à classifier, il est capital de ne considérer que des solutions permettant une classification en un temps raisonnable; cela exclut presque immédiatement toute solution itérative. Nous avons également focalisé notre attention sur les méthodes à noyau, tirant avantage de l'astuce du noyau et des bonnes propriétés de ces méthodes.
L'algorithme Somno-Art étant composé de nombreuses parties, nous nous focalisons ici sur l'amélioration du détecteur de sommeil paradoxal (REM). Les méthodes développées pourront être appliquées sur les autres parties de l'algorithme.

A.2 État de l'art

Dans cette section, nous introduisons les définitions principales du transfert d'apprentissage et en présentons les fondamentaux; cela sert à situer précisément quelles méthodes sont adéquates à notre problématique. Dans un second temps, nous présentons rapidement les principales propriétés des méthodes à noyau, et comparons succinctement nos contributions méthodologiques à celles de la littérature.

A.2.1 Transfert d'apprentissage: principales définitions

Les techniques "traditionnelles" d'apprentissage machine reposent sur l'hypothèse suivante: la tâche de classification est unique et les données d'apprentissage et d'application suivent la même distribution statistique et le même lien aux étiquettes recherchées. Dès lors que cette hypothèse n'est plus vérifiée, les performances de classification

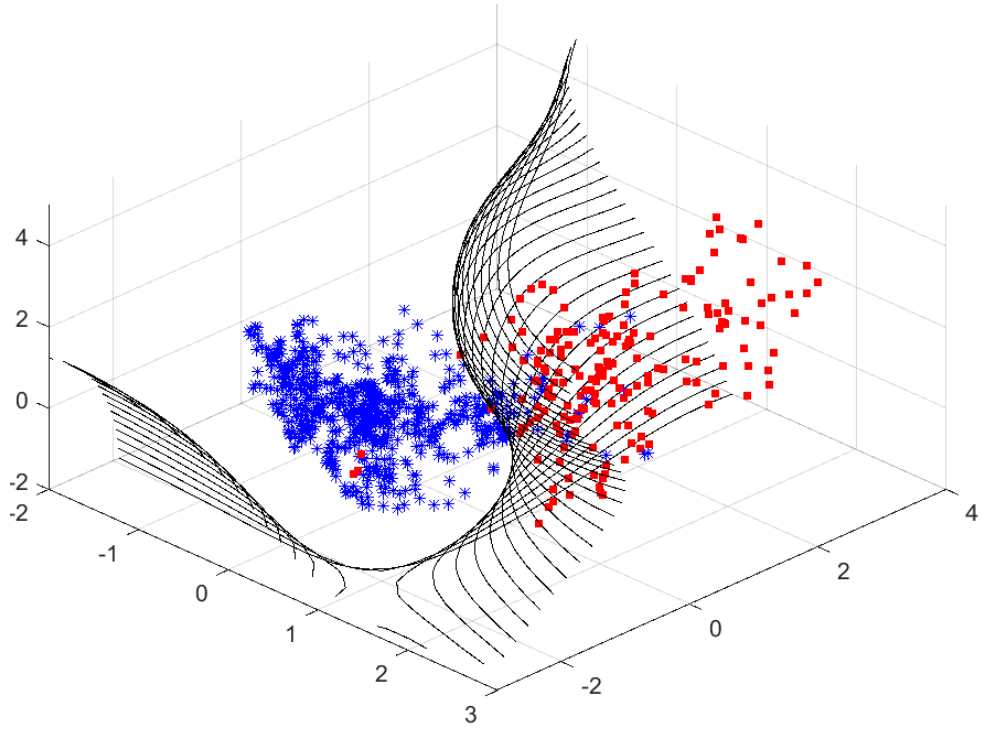


Figure A.2: Apprentissage sur un enregistrement "source": les astérisques bleus représentent le N2 et le N3, les carrées rouges le REM. Les lignes noires délimitent la frontière de décision d'un SVM entraîné sur ces données.

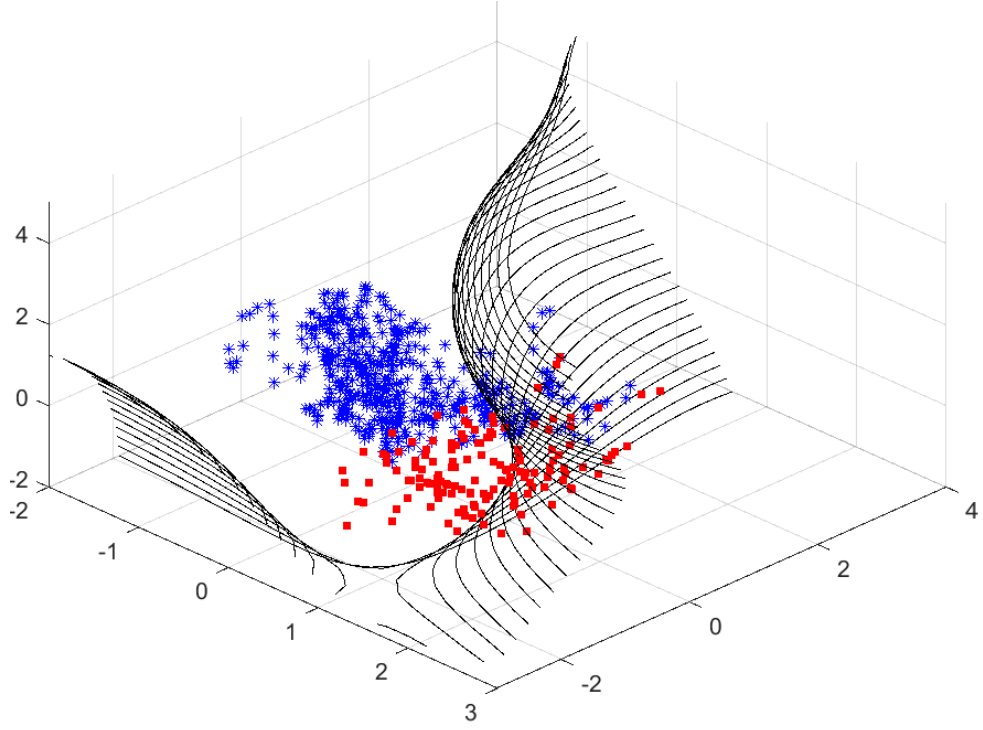


Figure A.3: Mise en évidence du décalage des données: sur cet enregistrement également, les astérisques bleus représentent le N2 et le N3, les carrées rouges le REM. Il apparait clairement que le détecteur représenté sur la Figure (A.2) classifierait à tort nombre d'observations de REM en Non-REM.

sont réduites. Le transfert d'apprentissage [19, 20, 21] permet de pallier ce genre de limitations, et ce de diverses manières et pour divers problèmes.
Avant de poursuivre, il est nécessaire de définir précisément quels types de différences il peut exister entre un jeu de données d'apprentissage, que l'on appellera *source* dans la suite, et un jeu de données aux étiquettes inconnues, *cible* [19]:

- **Domaine:** Un domaine $\mathcal{D}$ se compose d'un ensemble d'observations $X = \mathbf{x}_{i,i\in\{1,...,N\}} \in \mathcal{X}$, $\mathcal{X}$ étant l'espace de représentation, et $P(X)$ étant la loi de probabilité marginale associée. Nous abrégeons le domaine $\mathcal{D} = \{\mathcal{X}, P(X)\}$.

- **Tâche:** Une tâche $\mathcal{T}$ est associée à un domaine $\mathcal{D}$, et est composée des étiquettes $y_{i,i\in\{1,...,N\}} \in \mathcal{Y}$, $\mathcal{Y}$ étant l'espace des étiquettes, et leur fonction de prédiction associée $f : \mathcal{X} \to \mathcal{Y}$, qui correspond la majorité du temps, d'un point de vue statistique, à $f(\mathbf{x}_i) = p(y_i|\mathbf{x}_i)$. Nous abrégeons la tâche $\mathcal{T} = (\mathcal{Y}, f(.))$

- **Source:** Une source S est composée d'un ensemble d'observations et de leurs étiquettes associées. Nous dénotons par $\mathcal{D}_S = \{(\mathbf{x}_i, y_i)\}_{i\in\{1,...,N_S\}} \in \mathcal{X}_S \times \mathcal{Y}_S$ le *domaine des données source*. Généralement, les étiquettes source sont connues et N_S, le nombre d'observations source, est grand.

- **Cible:** Une cible T est composée d'un ensemble d'observations dont les étiquettes associées sont généralement inconnues - et sont par ailleurs ce que l'on cherche à déterminer. Nous dénotons le *domaine des données cible* par $\mathcal{D}_T = \{(\mathbf{x}_i, y_i)\}_{i\in\{1,...,N_T\}} \in \mathcal{X}_T \times \mathcal{Y}_T$. Il correspond à l'ensemble d'application sur lequel nous cherchons à assurer des bonnes performances en classification.

Suivant ces définitions, il apparait clairement que l'apprentissage machine "traditionnel" fonctionne lorsque $\mathcal{D}_S = \mathcal{D}_T$ et $\mathcal{T}_S = \mathcal{T}_T$. Nous pouvons à présent donner une définition précise du transfert d'apprentissage:

Definition 5 (Transfert d'apprentissage) *Soient un domaine source $\mathcal{D}_S$ et une tâche d'apprentissage source $\mathcal{T}_S$, un domaine cible $\mathcal{D}_T$ et une tâche d'application cible $\mathcal{T}_T$, le **transfert d'apprentissage** tend à améliorer les capacités prédictives d'une fonction de classification cible $f_T(.)$ en $\mathcal{D}_T$ en tirant avantage des connaissances apprises en $\mathcal{D}_S$ et $\mathcal{T}_S$, où $\mathcal{D}_S \neq \mathcal{D}_T$, et/où $\mathcal{T}_S \neq \mathcal{T}_T$.*

Il existe différentes sous-catégories de transferts d'apprentissage, dépendant des différences précises entre domaines et tâches de la source et de la cible; dépendant des connaissances éventuelles que l'on a des étiquettes cibles; dépendant finalement de la méthode utilisée pour effectuer le transfert.
Dans le premier cas:

- Le transfert est dit **transductif** s'il s'agit d'une différence de domaine entre source et cible: $\mathcal{D}_S \neq \mathcal{D}_T$. Dans ce cas, soit il s'agit d'une différence d'espace de représentation, $\mathcal{X}_S \neq \mathcal{X}_T$ et le transfert est alors dit **hétérogène**; soit cet espace est le même - **homogène** - et la différence réside dans la loi de probabilité marginale des observations: $P(X_S) \neq P(X_T)$.

- Le transfert est dit **inductif** si les tâches sont différentes: $\mathcal{T}_S \neq \mathcal{T}_T$. Dans ce cas, soit les étiquettes recherchées diffèrent $\mathcal{Y}_S \neq \mathcal{Y}_T$, soit elles sont les mêmes et c'est leurs lois conditionnelles aux observations qui diffèrent: $p(Y_S|X_S) \neq p(Y_T|X_T)$.

- Finalement, il est dit **non-supervisé** ou encore **implicite** si ni le domaine ni la tâche ne sont les mêmes: $\mathcal{D}_S \neq \mathcal{D}_T$ et $\mathcal{T}_S \neq \mathcal{T}_T$.

Pour le second point, on parle de transfert **informé** si une partie des étiquettes cibles est connue et peut être utilisée pour améliorer le transfert, et **non-informé** dans le cas contraire.
Enfin, nous décrivons ci-dessous quatre des principales manières de transférer de la connaissance entre source et cible:

- Transfert de **feature/représentation**: l'idée est de rechercher un nouvel espace de représentation où source et cible sont plus proches [24] (selon des critères d'optimisation tels que des distances ou divergences [16, 22, 23], généralement).

- Transfert d'**instance**: il s'agit ici de sélectionner et pondérer des observations de la source pour réutilisation au sein de la cible.

- Transfert de **paramètres**: recherche de paramètres communs à la source S et à la cible T pour optimiser le transfert [25, 26, 27].
- Transfert de **relation/connaissance**: assez particulier et généralement utilisé dans les réseaux relationnels, recherche d'une carte des relations liant S et T.

Il est également important d'évoquer le transfert **négatif**, qui correspond aux situations où la connaissance transférée a un effet détrimentaire sur la classification des observations cibles.

A.2.2 L'adaptation de domaine

Dans le cadre de notre application:

- L'espace des représentations est le même entre source et cible $\mathcal{X}_S = \mathcal{X}_T$ et il en va de même pour l'espace des étiquettes $\mathcal{Y}_S = \mathcal{Y}_T$. Nous considérerons donc des méthodes de transfert homogène, bien que des méthodes hétérogènes puissent être considérées également dans de futures recherches.
- Nous n'avons aucune information sur les étiquettes cibles, le transfert est donc non-informé.
- La différence entre source et cible réside in fine dans leurs domaines $\mathcal{D}_S \neq \mathcal{D}_T$; nous nous focaliserons donc sur les méthodes de transfert transductif.

De manière plus détaillée, nous distinguerons trois types de décalages affectant les tâches de classification:

- Le décalage co-varié, une différence des lois marginales des observations: $p(X_S) \neq p(X_T)$.
- Le décalage des probabilités a priori des classes: $p(Y_S) \neq p(Y_T)$.
- Le décalage de concept, une différence dans les lois conditionnelles des étiquettes aux observations: $p(Y_S|X_S) \neq p(Y_T|X_T)$.

Nous nous focaliserons au travers de cette thèse sur l'établissement de méthodes d'adaptation de la décision (une idée similaire à l'adaptation de domaine), afin de pallier aux effets néfastes de ces décalages. Au vu du nombre important de données à traiter, l'efficacité computationnelle est une caractéristique importante des méthodes que nous avons développées; cela constitue une différence importante avec une bonne partie des méthodes de la littérature, qui reposent généralement sur la résolution de problèmes d'optimisation sous contraintes résultant souvent en des méthodes itératives.
Additionnellement, au vu du nombre important d'enregistrements dont nous disposons, nous proposons une méthode de sélection de source, le principe étant, pour une cible donnée, de choisir la (ou les) meilleure(s) source(s) assurant une classification efficace.

A.2.3 Les méthodes à noyau

Nos méthodes sont principalement basées sur les noyaux, de par leur praticité, leur rapidité d'exécution et les avantages qu'apporte l'astuce du noyau.
Deux aspects des noyaux sont particulièrement importants dans nos travaux: l'alignement noyau-cible (*kernel target alignment* - KTA), une mesure de similarité entre information portée par les observations et celle portée par les étiquettes; et le centrage des matrices noyau, qui influera significativement sur la mesure de KTA et sur nos méthodes de transfert/de sélection de source(s).
Le KTA [46] est calculé comme suivi:

$$\begin{aligned} KTA(K, YY') &= \frac{\langle K, YY' \rangle_{\mathcal{F}}}{\sqrt{\langle K, K \rangle_{\mathcal{F}} \langle YY', YY' \rangle_{\mathcal{F}}}} \\ &= \frac{Y'KY}{||K||_F ||Y||^2} \end{aligned} \tag{A.1}$$

où K dénote la matrice de Gram calculée au sein d'un jeu d'observations, Y les étiquettes en vecteur colonne, $\langle \star, \star \rangle_F$ le produit scalaire de Frobenius, $|| \star ||_F$ la norme de Frobenius et $|| \star ||$ la norme $\mathcal{L}_2$. Il prend valeur entre

-1 et 1, selon le degré de similarité entre les deux matrices. Traditionnellement, le KTA est utilisé pour faire de l'optimisation/de l'ingénierie de noyau [47, 48]: les étiquettes Y sont connues, et le KTA est maximisé selon le noyau et ses paramètres. Dans nos méthodes, nous faisons l'inverse: le noyau et ses paramètres ont été fixés lors de l'apprentissage et la validation, et la maximisation du KTA sera faite sur les étiquettes recherchées; cela sera développé au travers de la section suivante.
Traitons à présent la question du centrage; un élément de la matrice centrée K_C a pour expression générale:

$$K_{C_{ij}} = \langle \phi(\mathbf{x}_i) - C,\ \phi(\mathbf{x}_j) - C \rangle_{\mathcal{H}} \tag{A.2}$$

où $\mathbf{x}_i$ et $\mathbf{x}_j$ sont deux observations, C est le centre considéré, et $\langle \star, \star \rangle_{\mathcal{H}}$ dénote le produit scalaire dans l'espace de Hilbert à noyau reproduisant (RKHS) $\mathcal{H}$.
Nous considérons alors deux manières de centrer: soit directement dans l'espace des observations $C = \phi(\mathbf{x}_s)$ pour un centre $\mathbf{x}_s \in \mathcal{X}$, soit dans le RKHS $\mathcal{H}$ en utilisant une combinaison linéaire des observations $C = \sum_{i=1} \alpha_i \phi(\mathbf{x}_i)$ où le centre est déterminé par le choix du vecteur des pondérations $\boldsymbol{\alpha}$.
Un centrage efficace au sens de la maximisation de l'alignement (ou du transfert d'apprentissage) nécessite la connaissance des étiquettes [51, 49], qui sont supposées inconnues dans le cas de la cible. Nous utilisons donc principalement le centrage *moyen* dans le RKHS, pour lequel tous les poids sont fixés à $\alpha_i = \frac{1}{N_T}$ où N_T est le nombre d'observations cible. Cela entraîne une expression pratique pour le calcul de la matrice centrée:

$$K_C = (\mathcal{I}_N - \frac{1}{N}\mathcal{J}_N) K (\mathcal{I}_N - \frac{1}{N}\mathcal{J}_N) \tag{A.3}$$

où K est la matrice de Gram non centrée, $\mathcal{I}_N$ est la matrice identité de taille N et $\mathcal{J}_N$ la matrice dont tous les éléments valent 1.
Nous représentons dans la Figure A.4 l'effet d'un choix de centre adéquat: le KTA mesuré entre la matrice des étiquettes (illustrée en haut à gauche) et la matrice noyau centrée sur C_2 (en bas à droite) est plus grand que celui mesuré en centrant sur C_1 (en bas à gauche), traduisant sa plus grande capacité de discrimination des classes.

A.3 Contributions méthodologiques

Nos contributions méthodologiques comprennent deux méthodes de transfert d'apprentissage, une méthode d'optimisation du centrage des matrices noyau (élément important de nos méthodes de transfert), et la méthode de sélection de source(s). Cette partie présente les éléments essentiels de chaque méthode.

A.3.1 Quadratic Loss Transfer Learning

Cette première méthode [52] repose sur le compromis entre une partie entièrement supervisée, la minimisation du coût quadratique entre les étiquettes cherchées et les étiquettes estimées par un ensemble de détecteurs, et une seconde partie entièrement non-supervisée, la maximisation de la polarisation noyau-cible (KTP - le numérateur du KTA et donc son équivalent non-normalisé). Le problème d'optimisation résultant s'exprime donc:

$$\hat{Y_T} = \underset{Y_T}{argmin} \sum_{i=1}^{N_S} \left(||Y_T - Y_T^{\{S_i\}}||^2 \right) - \lambda \langle K_{TT}, Y_T Y_T' \rangle_F \tag{A.4}$$

où Y_T dénote les étiquettes cibles que l'on cherche à déterminer, $Y_T^{\{S_i\}}$ est l'estimation de Y_T fournie par le détecteur associé à la source S_i, N_S est le nombre de sources considérées, K_{TT} est la matrice noyau calculée au sein des observations cibles et $\lambda \geq 0$ est un paramètre de compromis permettant de moduler entre supervision et non-supervision.
L'étude de la matrice Hessienne nous permet d'assurer la convexité du problème, nécessitant la contrainte additionnelle $\lambda < \frac{N_S}{\mu_{max}}$, où μ_{max} dénote la plus grande valeur propre de K_{TT}. La solution du QLTL est alors fournie par:

$$\hat{Y_T} = (N_S \mathbb{I}_N - \lambda K_{TT})^{-1} \left(\sum_{i=1}^{N_S} Y_T^{\{S_i\}} \right) \tag{A.5}$$

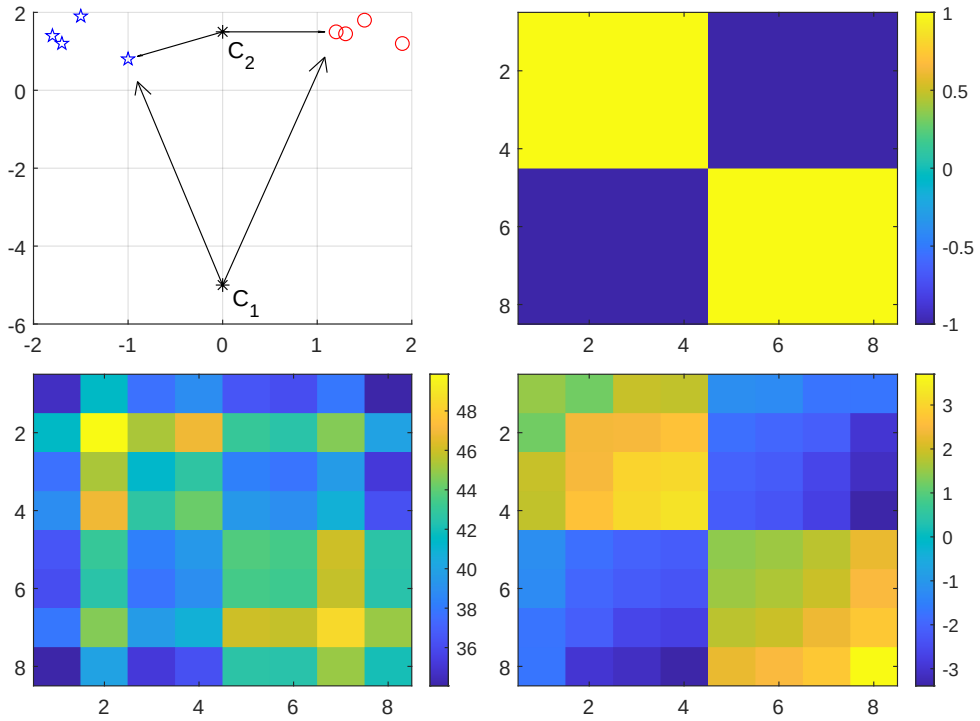


Figure A.4: Le choix de l'origine du système de coordonnées (ou centre) est important. Nous représentons les matrices noyau obtenues pour deux centres possibles et quelques données simulées (représentées en haut à gauche): la matrice des étiquettes est représentée en haut à droite et les matrices noyau observées en bas à gauche et en bas à droite pour resp. C_1 et C_2.

En pratique, nous remplaçons λ par son équivalent normalisé $\lambda_{rel} = \lambda \frac{\mu_{max}}{N_S}$. L'étude du comportement asymptotique de notre méthode montre que la solution obtenue est la moyenne des sorties des détecteurs $\frac{1}{N_S}(\sum_{i=1}^{N_S} Y_T^{\{S_i\}})$ pour $\lambda_{rel} = 0$, solution entièrement supervisée (minimisant le coût quadratique seul) et le vecteur propre associé à μ_{max} quand $\lambda_{rel} \to 1$, solution entièrement non-supervisée dans ce second cas.

A.3.2 Kernel Alignment Transfer Learning

Avant de présenter la méthode (KxATL) nous introduisons une extension simple du KTA: l'alignement noyau-croisé (KCA), qui mesure tout comme le KTA la similarité entre une matrice noyau et une matrice des étiquettes, mais adapté à une source et une cible: la matrice noyau est calculée entre les observations source et cible, la matrice des étiquettes entre les étiquettes source et cible:

$$\begin{aligned} \mathcal{A}(K_{TS},\, Y_T Y_S') &= \frac{\langle K_{TS},\, Y_T Y_S' \rangle_F}{\sqrt{\langle K_{TS},\, K_{TS} \rangle_F \langle Y_T Y_S',\, Y_T Y_S' \rangle_F}} \\ &= \frac{Y_T' K_{TS} Y_S}{||K_{TS}||_F ||Y_T|| ||Y_S||} \end{aligned} \tag{A.6}$$

où K_{TS} est la matrice noyau-croisée, Y_T et Y_S sont les vecteurs des étiquettes cible (estimés) et source (connus), respectivement.
Notre seconde méthode de transfert d'apprentissage est le KxATL [53], qui consiste simplement à maximiser l'alignement selon les étiquettes cible recherchées: KTATL dans le cas de la maximisation du KTA et KCATL dans le cas de la maximisation du KCA. Ces derniers sont maximisés lorsque les étiquettes recherchées sont colinéaires à resp. $K_{TT} Y_T^S$ et $K_{TS} Y_S$, entrainant une infinité de solutions. Afin de rendre la solution unique, nous ajoutons la contrainte suivante: $\hat{Y_T}$ doit être de norme unité; nous pourrions également imposer qu'il soit de norme égale à celle de Y_S, ou simplement centrer comme nous le faisons pour le noyau, mais quelle que soit l'approche retenue, une problématique importante est le choix du seuil de décision appliqué à la statistique en résultant. Les solutions respectives du KTATL et KCATL sont donc:

$$\hat{Y_T} = \frac{K_{TT} Y_T^S}{||K_{TT} Y_T^S||} \tag{A.7}$$

où K_{TT} est la matrice noyau cible calculée au sein des observations cible et Y_T^S est la sortie du détecteur source appliqué aux données cible, et

$$\hat{Y_T} = \frac{K_{TS} Y_S}{||K_{TS} Y_S||} \tag{A.8}$$

où K_{TS} est la matrice noyau croisée calculée entre les observations source et cible, et Y_S est le vecteur des étiquettes source, connues (vérité-terrain).

A.3.3 Centrage optimisé des matrices croisées

Le centrage des matrices noyau croisées repose sur un centre source C_S et un centre cible C_T, et se développe de manière similaire à celui des matrices noyau cibles.
Nous considérons alors l'optimisation du KCA (ou KCP) selon le choix du centre source et/ou cible [54]. Pour ce faire, nous proposons les expressions analytiques nécessaires à l'établissement d'un algorithme du gradient. L'objectif étant toujours la détermination des étiquettes cible, à chaque mise à jour des centres utilisés pour centrer K_{TS}, nous mettons également à jour le vecteur des étiquettes Y_T comme sortie du KCATL (immédiat); cela résulte in fine en un algorithme d'optimisation par directions alternées.
Les résultats observés montrent un intérêt clair de cette méthode, mais aussi que le problème résulte en des résultats souvent imprédictibles. La nature itérative de cette solution la rend également moins intéressante pour notre application. Les détails computationnels sont détaillés dans le cœur du texte (hors le présent résumé).
Nous avons considéré six variantes de la méthode, selon si on maximise le KCA ou le KCP, si on maximise selon les deux centres ou selon le centre source seul, et selon si on exprime les centres dans l'espace des observations ou dans le RKHS. La maximisation du KCP par centrage dans le RKHS n'a pas été considérée, le problème étant mal posé.

A.3.4 Sélection de source(s)

Au vu du nombre important d'enregistrements étiquetés dont nous disposons, une idée complémentaire au transfert d'apprentissage est la sélection de source: pour une cible donnée, il doit exister un ou plusieurs détecteurs sources résultant en des performances en classification optimales [55].
Nous cherchons donc un critère - prédicteur - se corrélant bien avec les performances en application d'une source sur une cible et permettant ainsi une sélection adéquate; pour établir un tel critère, nous nous sommes basés sur deux hypothèses complémentaires associées au transfert d'apprentissage:

- Si le décalage des données entre la source et la cible est faible, l'information mesurée dans la matrice noyau devrait être fortement similaire à l'information estimée et/ou fournie par la source.
- Si le décalage des données est faible, le transfert d'apprentissage ne devrait influer que marginalement sur la décision fournie par la source.

Sur base de ces hypothèses, nous avons donc établi quelques prédicteurs dont nous avons alors pu étudier les capacités de prédiction:

- Le KTA (resp. KCA) *observé*, où Y_T est la sortie du détecteur source: $Y_T = Y_T^S$.
- Le KTA (resp. KCA) *optimisé*, où Y_T est la sortie du KTATL (resp. KCATL).
- La corrélation entre la sortie du détecteur sans transfert d'apprentissage, Y_T^S, et celle du KTATL (resp. KCATL).

Les performances de la sélection de détecteur(s) source(s) (TSS) reposent in fine:

- Sur la qualité des prédicteurs (la précision avec laquelle ils arrivent à prédire les vraies performances en détection).
- Les sources considérées; il faut constituer un ensemble efficace de sources proposant des informations complémentaires et adapté à toute cible à classifier.
- La manière dont la décision est fournie: nous pouvons considérer uniquement la source pour laquelle le prédicteur est maximal, ou, par exemple, fusionner les sorties des k sources le maximisant.

A.4 Étude approfondie des données

Avant d'appliquer nos méthodes, nous avons étudié de manière extensive les données d'application, principalement au travers des matrices de Gram.

A.4.1 Informations discriminantes des matrices de Gram

L'analyse se fait au travers d'une décomposition en valeurs et vecteurs propres dans le cas d'une matrice noyau cible:

$$K_{TT} = V\Sigma V' = \sum_{i=1}^{N_T} \lambda_i V_i V_i' \tag{A.9}$$

où Σ est une matrice diagonale contenant les valeurs propres et V est une matrice orthogonale (dite de passage) composée des vecteurs propres associés; les $\lambda_i$dénotent les valeurs propres de K_{TT} et les V_i sont les vecteurs propres associés ($i^{\text{ème}}$ colonne de V), formant une base orthogonale.
Dans le cas de matrices non-carrées, telles que nos matrices noyau-croisées, nous pouvons effectuer une décomposition en valeurs singulières (SVD):

$$K_{TS} = V\Sigma U' = \sum_{i=1} \lambda_i V_i U_i' \tag{A.10}$$

où Σ contient les valeurs singulières de K_{TS}, λ_i (égales aux racines carrées des valeurs propres de $K_{TS}K_{TS}'$), U est une matrice orthogonale (dite d'entrée) et V une autre matrice orthogonale (dite de sortie).

Lorsque nous regardons plus précisément les *premiers* vecteurs propres de K_{TT} (ceux associés aux plus grandes valeurs propres), ou encore les *premiers* vecteurs singuliers de sortie de K_{TS}, ils se corrèlent généralement bien avec le REM de la vérité-terrain cible, Y_T. Telle est l'information non-supervisée (au sens que les étiquettes source Y_S ne sont ici pas exploitées) dont nous tirons avantage au travers de nos méthodes.
Nous avons représenté en la Figure (A.5) le vecteur propre de K_{TT_C} associé à sa plus grande valeur propre, ainsi que le REM de la vérité-terrain. Il apparaît clairement que K_{TT_C} porte une information permettant de discriminer le REM du reste.

A.4.2 Effet du centrage moyen dans le RKHS

Nous avons observé les propriétés d'entrelacement des valeurs propres de la matrice non-centrée, avec celles de son équivalent centré sur la moyenne des observations dans le RKHS [56]:

$$\lambda_1 \geq \lambda_{C_1} \geq \lambda_2 \geq \lambda_{C_2} \geq \lambda_3 \geq [\cdots] \tag{A.11}$$

où λ_i est la $i^{\text{ème}}$ valeur propre de K_{TT} et λ_{C_i} la $i^{\text{ème}}$ de K_{TT_C}.
Cela s'observe également sur les vecteurs propres associés: le vecteur propre de K_{TT} associé à sa seconde plus grande valeur propre se corrèle très fortement au vecteur propre de K_{TT_C} associé à sa plus grande valeur, et de même pour la troisième de K_{TT} avec la seconde de K_{TT_C}, la quatrième avec la troisième, etc.
Nous avons également pu observer (sans démontrer) des effets similaires dans le cas des matrices noyau-croisées K_{TS} et K_{TS_C}: leurs valeurs singulières s'entrelacent, et leurs vecteurs singuliers associés sont fortement corrélés: $U_{(i+1)}$ avec U_{C_i} et $V_{(i+1)}$ avec V_{C_i} $\forall i$.
Nous indiquons également que les vecteurs propres d'entrée de K_{TS} sont fortement corrélés avec les vecteurs propres de K_{TT} et les vecteurs singuliers de sortie avec les vecteurs propres de K_{SS} (matrice noyau-source calculée au sein des observations source).
Nous avons représenté les vecteurs singuliers d'entrée de K_{TS} et K_{TS_C} en la Figure (A.6), faisant clairement apparaître l'effet du centrage "moyen" dans le RKHS, qui résulte en des vecteurs propres similaires, seulement décalés d'un indice.

A.4.3 Observations sur l'application

Nous faisons les observations suivantes:

- Les features utilisés pour discriminer le REM du reste sont aussi sensibles à l'éveil W. C'est cependant peu problématique, l'éveil pouvant être discriminé du REM par d'autres moyens.
- La détection de la première phase de REM peut être problématique; dans certaines situations, nous détectons à tort un premier REM qui n'existe pas, dû généralement à des transitions courtes vers l'éveil en début de nuit; dans d'autres situations, nous ratons le premier REM qui est trop court et n'influe pas assez la statistique de décision pour dépasser le seuil de décision. Un filtrage adapté de la statistique de décision pourrait aider à pallier ce problème de premier REM.
- Le décalage des probabilités a priori des étiquettes est un problème récurrent, notamment dans le cas de sujets qui ne dorment pas, ou peu.

Nous avons également pu illustrer à nouveau le problème de décalage des données mis en évidence dans les Figures (A.2 et A.3) au travers de leur matrice noyau-croisée K_{TS_C}, représentée en Figure (A.7)

A.5 Optimisation technique des méthodes

Jusqu'à présent, nous avons introduit notre application, l'ensemble de la méthodologie (état de l'art et nos contributions) et quelques observations effectuées en utilisant les matrices noyau.
Nous appliquons à présent nos méthodes sur l'exemple précédent du couple (source, cible) présentant un décalage des données. Cela sert à étudier nos méthodes sur un exemple concret de décalage. Les choix de paramètres résultant de cet exemple ont été confirmés lors de l'application sur l'ensemble de validation, ce qui sera l'objet de la section suivante.

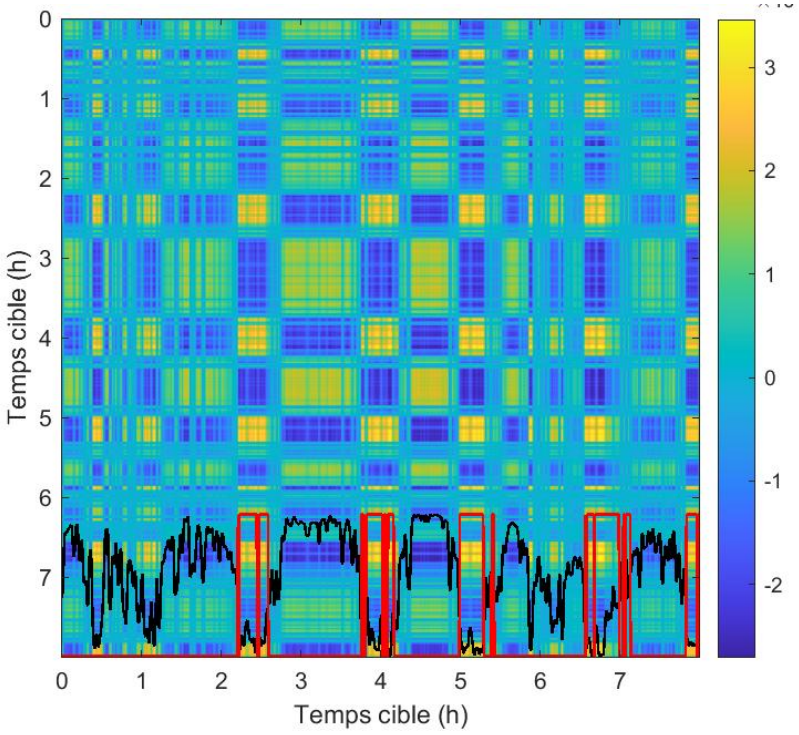


Figure A.5: Nous illustrons V_{C_1} le vecteur propre de K_{TT_C} associé à sa plus grande valeur propre (ligne noire). Il est clairement négativement corrélé au REM de la vérité-terrain (ligne rouge). La matrice en échelle de couleurs est la représentation de K_{TT_C}.

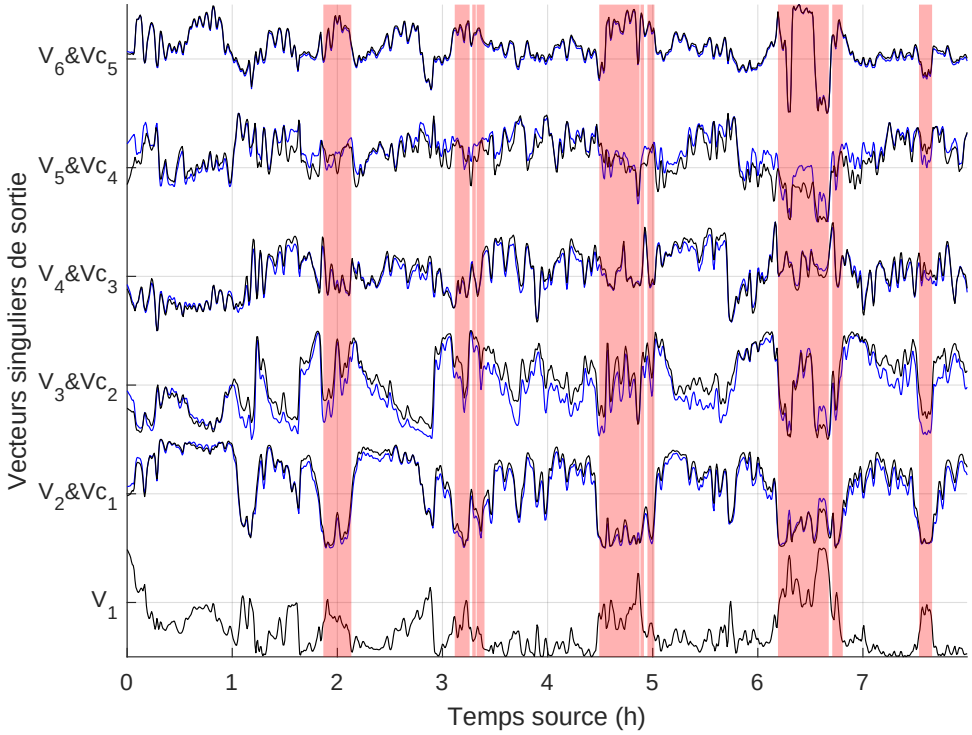


Figure A.6: Correspondance entre les $(i+1)^{èmes}$ vecteurs singuliers de K_{TS} et les $i^{èmes}$ de K_{TS_C}, son équivalent centré sur la moyenne des observations dans le RKHS. Les zones en rouge correspondent au REM de la vérité-terrain.

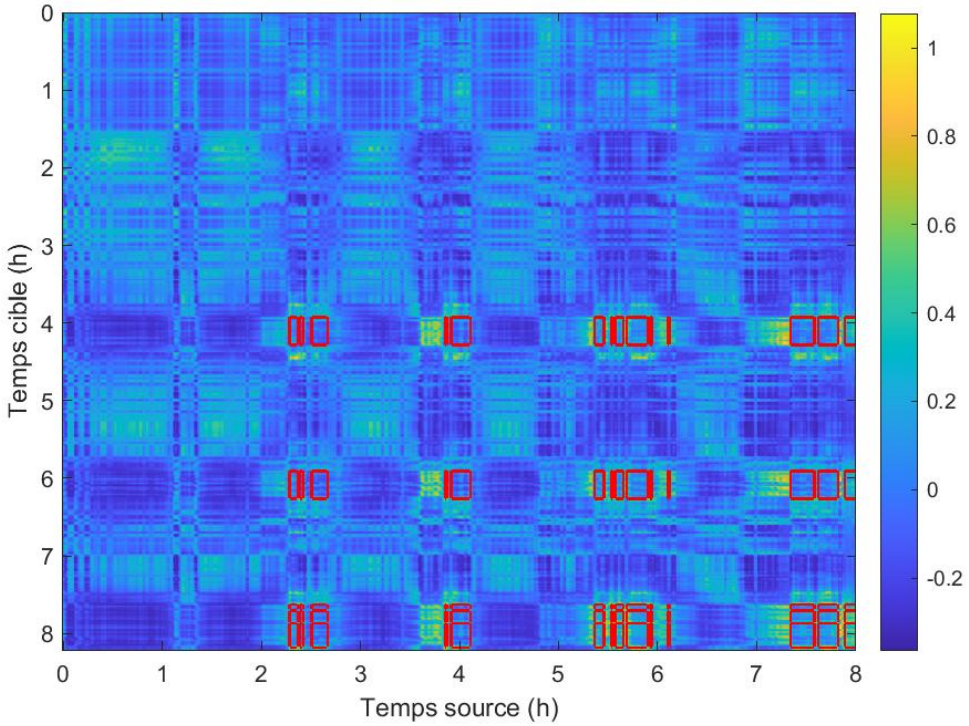


Figure A.7: Décalage des données des figures (A.2 et A.3), illustré utilisant cette fois une matrice noyau-croisée K_{TS_C} (matrice en échelle de couleurs). Les rectangles rouges correspondent aux instants de REM communs à la source et à la cible.

A.5.1 Estimation des performances

Afin d'évaluer l'évolution des performances par variation des paramètres de nos méthodes, nous considérons principalement deux métriques:

- Le point de fonctionnement pour un taux de fausse-alarme fixé à 7%. Cela permet de comparer précisément l'effet d'un choix de paramètres sur la statistique de décision. Cependant, ce n'est pas utilisable en pratique car cela nécessiterait de savoir choisir le seuil de décision adéquatement pour assurer 7% de fausse-alarme.

- La distance entre le point de fonctionnement observé et le point de fonctionnement idéal, ce dernier correspondant à une détection parfaite (taux de détection égal à 100%) sans fausse-alarme. Cette seconde métrique est plus proche de la sortie effective d'une de nos méthodes.

Nous représentons les sorties de nos méthodes de transfert en Figure (A.8) et les performances associées en Figure (A.9).

A.5.2 Paramètres de transfert d'apprentissage

Les différentes études menées ont conduit aux conclusions suivantes:

- QLTL, paramètre λ: ce paramètre joue sur le compromis entre supervision et non-supervision; une valeur fixée à 0.35 permet des performances optimales sur l'ensemble de validation, bien qu'une plus grande valeur soit désirable sur certains exemples, notamment quand l'information supervisée est de mauvaise qualité.

- Toutes méthodes, paramètre de noyau σ: joue de manière importante sur la dynamique de la matrice de Gram et donc sur l'information non-supervisée qu'elle contient; une valeur fixée à 2 donne un bon compromis entre sensibilité aux fortes variations et "lissage" de la statistique de décision.

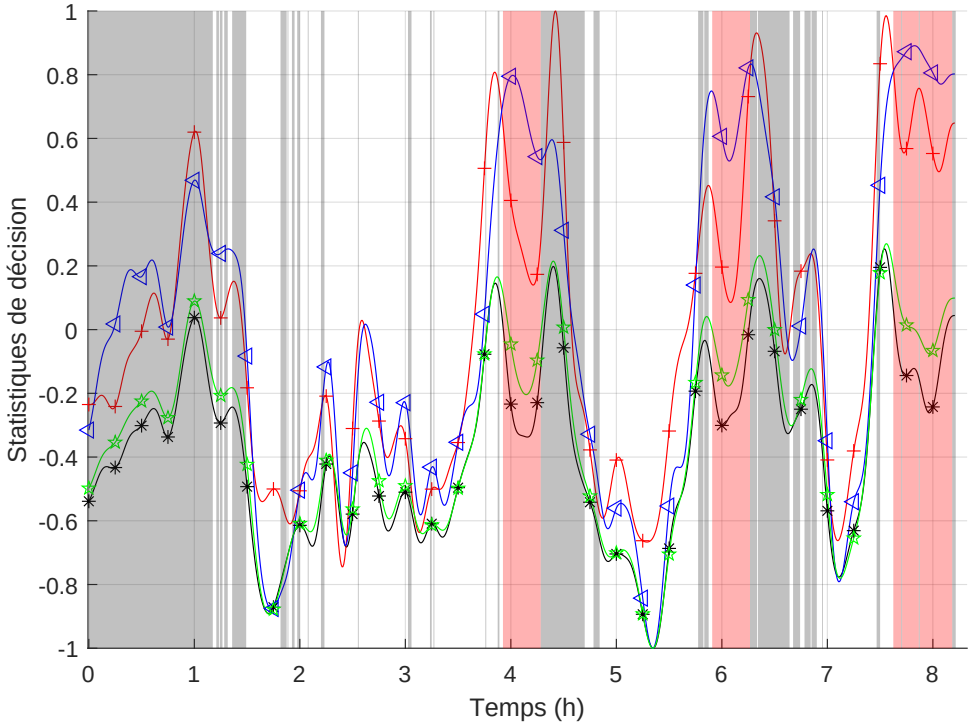


Figure A.8: Sorties du détecteur sans transfert (ligne noire avec astérisques), du QLTL (ligne verte avec étoiles), du KTATL (ligne bleue avec triangles) et du KCATL (ligne rouge avec croix). Les zones en rouge correspondent au REM de la vérité-terrain, celles en gris ne sont pas considérées pour l'estimation des performances.

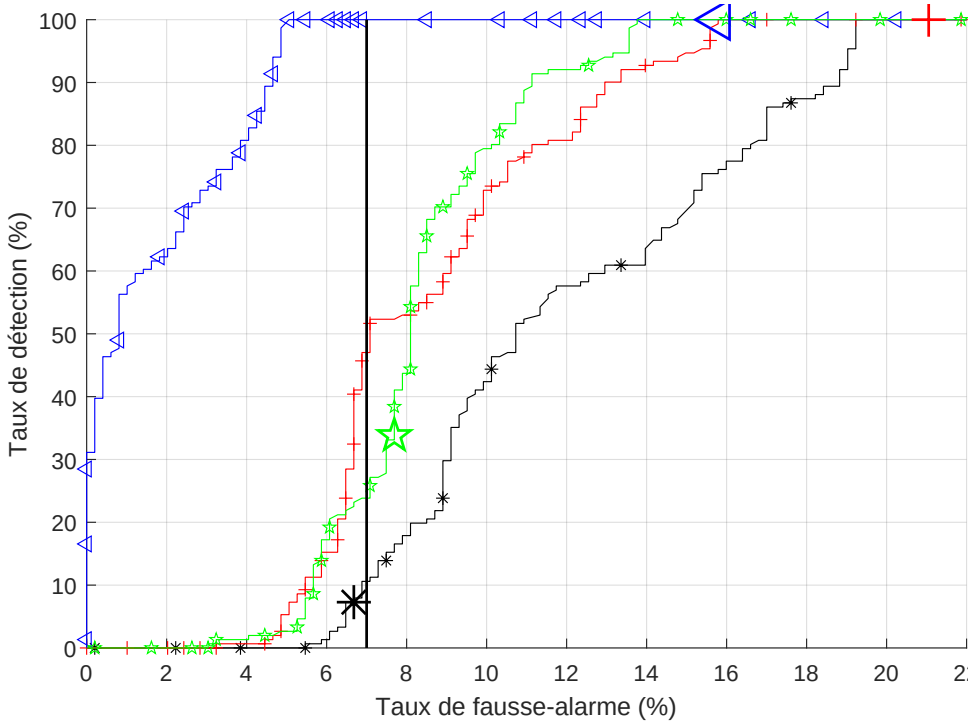


Figure A.9: Courbes des caractéristiques de fonctionnement du récepteur (ROC) correspondant aux statistiques de décision de la Figure (A.8): les grands marqueurs correspondent aux points de fonctionnement de chaque détecteur pour un seuil de décision à 0, tandis que la barre noire correspond à un taux de fausse-alarme fixé à $= 7\%$, possible uniquement en adaptant le seuil de décision adéquatement.

- Toutes méthodes, effet du codage de l'information supervisée: nous avons le choix pour le KTATL et le QLTL de binariser ou non l'estimation fournie par la source, et dans le cas du KCATL d'utiliser les labels source connues, ou de les ré-estimer. Nous conseillons de ne pas binariser, et d'utiliser la statistique de décision ré-estimée dans le cas du KCATL: cela augmente la cohérence entre matrices de Gram et information supervisée, et résulte généralement en des performances accrues en application.

Nous avons également comparé le KTATL à JDOT, une méthode de transport de densité (des observations conjointement aux étiquettes) de la littérature [32]. Les performances de classification sont similaires, cependant JDOT est bien plus lourd en temps d'exécution (méthode itérative) et cela empire pour des taux de décimation des données moins fortes (temps d'exécution non-linéaire au nombre d'observations à classer).

A.5.3 Paramètres de sélection de sources

Nous ajoutons trois autres sources au couple (source, cible) présentant un décalage des données, incluant:

- Une source adéquate pour la cible considérée et résultant en de très bonnes performances en classification.
- Une source moyennement adéquate.
- Une source inadéquate et résultant en de mauvaises performances en classification.

Nous appliquons nos prédicteurs de source sélection afin d'essayer de sélectionner la meilleure source parmi les quatre sources proposées. Cela mène aux conclusions (partielles) suivantes:

- Sur cet exemple, mis à part le KCA observé, tous nos prédicteurs sélectionnent systématiquement la bonne source.
- La plupart des valeurs de prédiction se corrèlent bien avec les vraies performances de classification.
- Généralement, les prédicteurs basés sur le KTA donnent lieu à de meilleurs résultats que ceux basés sur le KCA.
- Notre meilleur prédicteur semble être la corrélation entre résultats sans transfert d'apprentissage, et résultats du KTATL.

A.6 Résultats sur l'application

Nous considérons à présent l'application de nos méthodes sur tous les enregistrements de l'ensemble de validation.

A.6.1 Mesures de performances

Afin d'évaluer les performances en application de nos méthodes sur l'ensemble de validation, nous considérons les mesures suivantes:

- Taux de détection moyen pour un taux de fausse-alarme fixé à 7%.
- La distance entre le point de fonctionnement moyen et le point de fonctionnement "parfait" (taux de détection égal à 100% pour un taux de fausse-alarme de 0%).

A.6.2 Résultats du transfert d'apprentissage

Nous considérons chacune de nos sources pour effectuer le transfert d'apprentissage sur l'ensemble des cibles; les résultats sont représentés en Figure (A.10), où les indices des sources ont été triées par performances de classification croissances et sans transfert. Les méthodes ont été appliquées pour $\sigma = 2$, $\lambda_{rel} = 0.35$ (QLTL), les matrices noyau ont été centrées sur la moyenne dans le RKHS et les étiquettes sont estimées (ou ré-estimées dans le cas du KCATL) sans binarisation.
Nous faisons les observations suivantes:

- Le QLTL semble améliorer systématiquement les performances de classification, de manière mineure. Cela traduit l'apport bénéfique qu'apporte la prise en compte de l'information non-supervisée contenue dans la matrice noyau cible.
- Le KCATL améliore de manière importante nombre de détecteurs sources, mais présente aussi deux situations de transfert négatif (voir indices 41 et 50) et les gains observés sont assez variables.
- Le KTATL améliore fortement les performances, d'autant plus que le détecteur source (sans transfert donc) est mauvais.

Considérons à présent la meilleure source aux sens des performances de classification sans transfert (indice le plus à droite dans la représentation); les performances associées sont:

- $7.00 \pm 5.71\%$ taux de fausse-alarme (FAR) pour un taux de détection moyen (DR) égal à $81.73 \pm 15.75\%$.
- FAR égal à $7.02 \pm 5.66\%$ pour un DR égal à $82.92 \pm 15.59\%$ dans le cas du QLTL.
- FAR égal à $7.01 \pm 5.03\%$ pour un DR égal à $82.02 \pm 15.15\%$ dans le cas du KCATL.
- FAR égal à $7.00 \pm 6.13\%$ pour un DR égal à $83.26 \pm 13.07\%$ en utilisant le KTATL.

Cela signifie que même le meilleur détecteur sans transfert est amélioré, pour toutes nos méthodes de transfert d'apprentissage, la meilleure méthode étant le KTATL. Il existe plusieurs sources pour lesquelles le transfert résulte en de meilleures performances encore, le maximum atteint étant d'un FAR égal à $7.01 \pm 5.86\%$ pour un DR de $84.14 \pm 12.56\%$.
Nous rappelons à présent que l'objectif n'est pas tant d'améliorer les performances en moyenne que celles des nuits problématiques (présentant, par exemple, un décalage des données par rapport à la source considérée). Nous représentons donc les performances de la meilleure source (sans transfert) appliquée à chaque cible, et considérons l'évolution des performances par application de nos méthodes de transfert; la Figure (A.11) illustre le taux de détection lorsqu'on fixe la fausse-alarme à 7%, traduisant de la qualité de la statistique de décision, et la Figure (A.12) la distance du point de fonctionnement de chaque cible au point de fonctionnement "parfait", traduisant de la qualité de la détection obtenue. Nous faisons les constats suivants:

- Pour la plupart des cibles, le transfert n'affecte quasiment pas le point de fonctionnement à FAR égal à 7%.
- Dans certains cas, le transfert de connaissance est correctement effectué (voir par exemple indices 6, 8, 9, 11, 12 et 26 de la Figure (A.11), résultant également en une amélioration effective du point de fonctionnement associé (indices 8, 9, 12, 18 et 26 de la Figure (A.12).
- Il existe des cas de transfert négatif (indices 10 et 33) traduisant probablement de matrices noyau contenant une information non-discriminante pour ces enregistrements. Cela reste à confirmer par l'analyse des données.
- Les enregistrements cible aux indices 1 et 2 sont particuliers et correspondent à des situations où les features utilisés sont inadaptés et ne permettent pas de discriminer le REM du reste. C'est pour ces enregistrements en particulier qu'il faudrait considérer des méthodes de transfert hétérogènes. Ceux d'indice 2 et 3 sont quand même améliorés grâce au transfert.

Finalement, concernant les méthodes d'optimisation du centre pour amélioration des résultats du KCATL, ils résultent en des résultats similaires à ceux sans optimisation (voir dégradés). Les meilleurs résultats sont obtenus en maximisant le KCP selon le centre source dans l'espace des observations. Nous ne développons pas sur ces résultats, la méthode restant itérative et nécessitant d'être améliorée.

A.6.3 Résultats de la sélection de source(s)

Nous avons procédé en trois temps pour évaluer la sélection de source(s):

- Dans un premier temps nous considérons toutes les sources et cherchons à choisir, pour une cible donnée, la source maximisant le critère de sélection. Nous considérons alors comme statistique de décision celle fournie par la source sélectionnée.

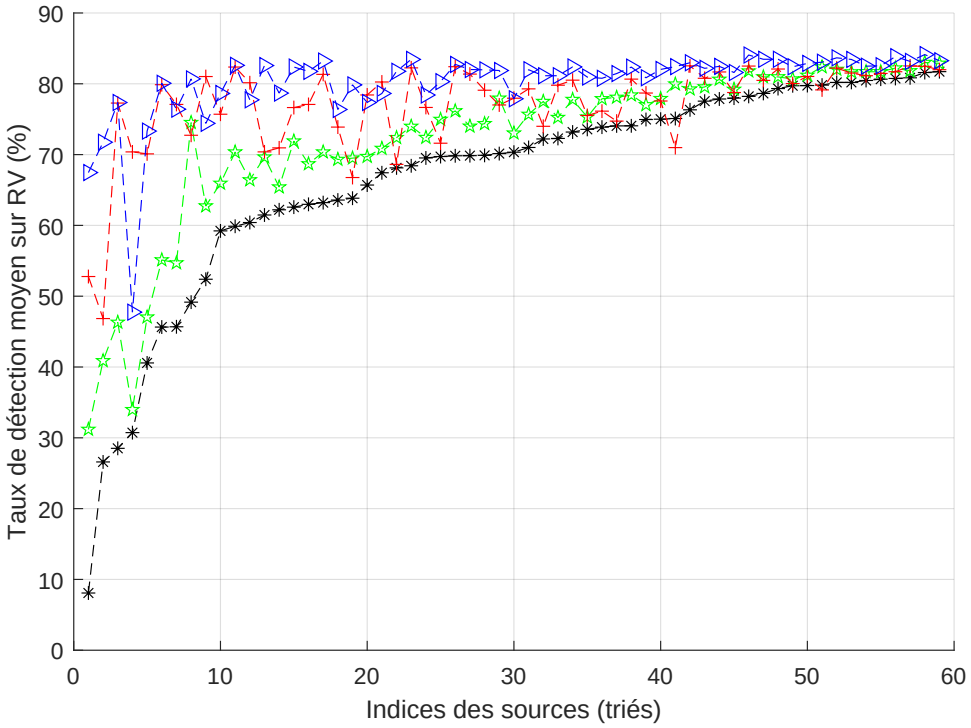


Figure A.10: Résultats des méthodes de transfert d'apprentissage sur l'ensemble de validation RV, point de fonctionnement moyen pour une fausse-alarme moyenne fixée à 7%. Les sources sont indexées par performances sans transfert croissantes; détecteurs sans transfert (astérisques noires), résultats du QLTL (étoiles vertes), du KCATL (croix rouges) et du KTATL (triangles bleus).

- Dans un deuxième temps nous considérons les $k = 5$ sources maximisant le critère et fusionnons leur statistique de décision, en pondérant les sorties par la valeur du critère et en normalisant par la somme des valeurs des critères.
- Dans un troisième temps, nous ne considérons plus toutes les sources mais un ensemble réduit de sources. Cet ensemble réduit a été constitué suivant différents principes: sources pour lesquelles la corrélation entre prédicteurs et performances (sur l'ensemble des cibles) est supérieur à un seuil; sources pour lesquelles les performances sont maximales; sources sélectionnées par une méthode séquentielle (*Sequential Floating Forward Selection* - SFFS) avec maximisation des performances de sélection de source(s).

La fusion de multiples sources a résulté en une amélioration des performances (plutôt que d'utiliser $k = 1$) et l'utilisation d'un ensemble restreint de sources parmi lesquelles sélectionner a résulté en une seconde amélioration des performances (et également en un temps d'exécution réduit, ce dernier étant proportionnel au nombre de sources considérées).
Les meilleurs résultats ont donc été obtenus pour $k = 5$, avec les sources retenues par le SFFS, et en utilisant comme prédicteur la corrélation entre sortie binarisée du détecteur de base (sans transfert) et sortie binarisée du KTATL: taux de fausse-alarme égal à $7.02 \pm 5.94\%$ pour un taux de détection de $83.43 \pm 13.77\%$.

Conclusions et perspectives

En conclusion, ce projet doctoral à résulté en deux méthodes de transfert d'apprentissage (QLTL, KxATL), un algorithme d'optimisation du centrage des matrices noyau-croisées pour maximisation du KCA (ou KCP), et d'une méthode de sélection de sources. Ces travaux ont donné lieu a deux participations à des conférences internationales, un article de journal scientifique international et au dépôt d'un brevet; cela est détaillé et résumé en Table (A.1).
Concernant l'application:

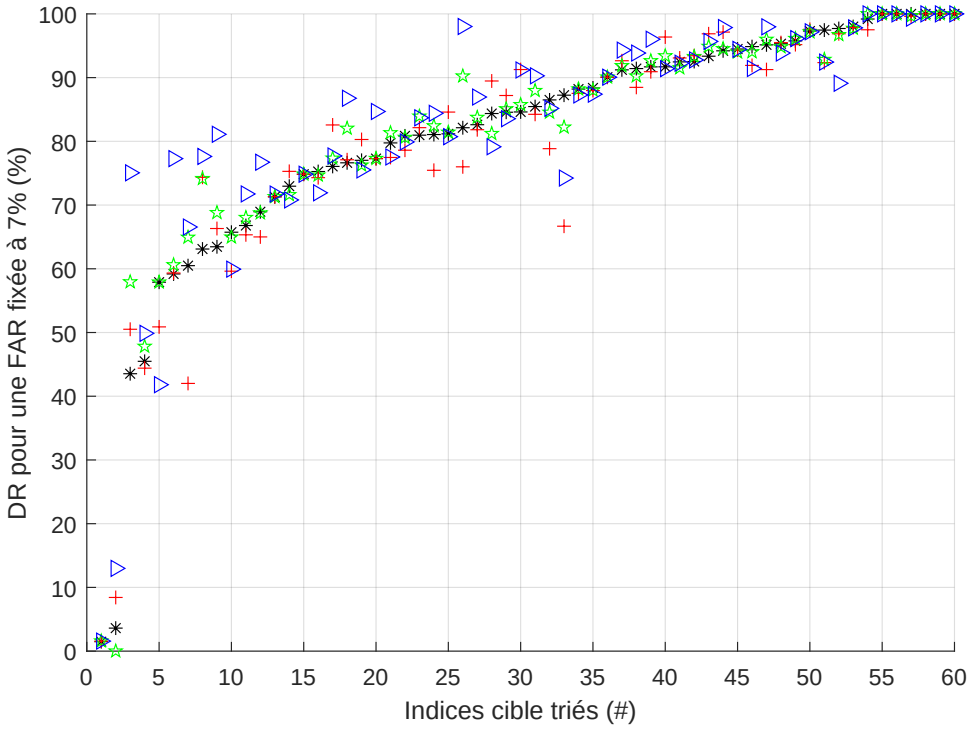


Figure A.11: Résultats pour chaque cible du meilleur détecteur source (fausse-alarme fixée à 7%). Les cibles sont indexées par performances croissantes du détecteur de référence. Détecteurs sans transfert (astérisques noires), résultats du QLTL (étoiles vertes), du KCATL (croix rouges) et du KTATL (triangles bleus).

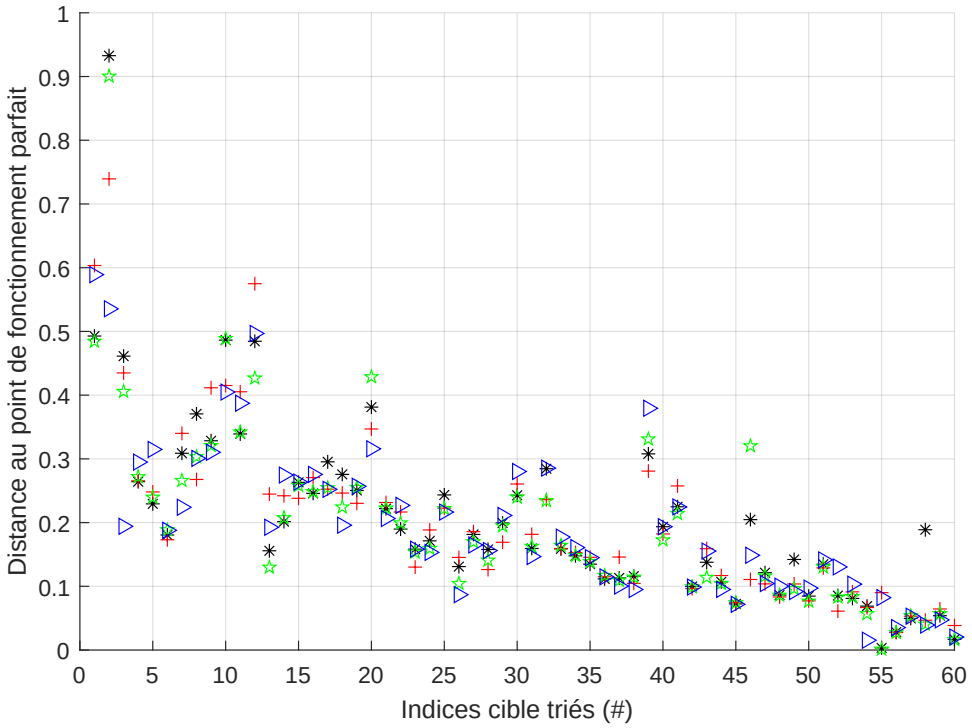


Figure A.12: Résultats pour chaque cible du meilleur détecteur source (distance au point de fonctionnement parfait). Cibles indexées dans le même ordre que la Figure (A.11). Détecteurs sans transfert (astérisques noires), résultats du QLTL (étoiles vertes), du KCATL (croix rouges) et du KTATL (triangles bleus).

- Nous considérons comme performances de référence celles du meilleur détecteur source sans transfert, pour lequel le point de fonctionnement à $7.00 \pm 5.71\%$ de taux de fausse alarme moyen pour un taux de détection moyen de $81.73 \pm 15.75\%$ sur l'ensemble de validation.
- La sélection de source dans sa meilleure configuration donne lieu à un FAR moyen égal à $7.02 \pm 5.94\%$ pour un DR moyen de $83.43 \pm 13.77\%$.
- Notre meilleure méthode de transfert, le KTATL, également dans sa meilleure configuration, résulte en un FAR moyen égal à $7.01 \pm 5.86\%$ pour un DR moyen de $84.14 \pm 12.56\%$, meilleurs résultats obtenus jusqu'à présent sur nos données.

Concernant finalement nos méthodes:

- Le QLTL et KTATL donnent des résultats consistants améliorant généralement un détecteur source considéré (peu ou pas de cas de transfert négatif). L'information contenue dans les matrices noyau-cibles joue un rôle crucial dans le transfert de connaissances et permet l'amélioration de la décision fournie par le détecteur source par la prise en compte de l'organisation des données cibles.
- Le KTATL est plus direct que le QLTL et nécessite le choix d'un paramètre de moins (λ, paramètre de compromis entre supervision et non-supervision du QLTL).
- Le KCATL améliore généralement la sortie d'un détecteur source mais de manière bien moins consistante que le KTATL: cette méthode est bien plus sensible aux propriétés statistiques des données source, ce qui peut générer du transfert négatif dans certains cas.
- Les trois méthodes affectent la statistique de décision du détecteur de référence, ce qui pose le problème du choix du seuil de décision; cela peut résulter en une plus grande dispersion des performances de classification.
- Ce problème de choix du seuil ne concerne pas la sélection de sources, qui a également l'avantage de permettre de prédire les performances de classifications, permettant éventuellement d'anticiper les enregistrements cibles problématiques.
- Le centrage optimisé des matrices noyau-cibles n'est pas fonctionnel en l'état et est inadapté à notre application, au vu du temps d'exécution (nature itérative de la méthode). Il donne cependant lieu a de bons résultats dans certaines situations, et mérite davantage de recherches.

Les perspectives à nos travaux sont nombreuses; il est cependant primordial de poursuivre les recherches permettant d'expliquer précisément les situations de transfert négatif; cela permettrait éventuellement d'améliorer davantage nos méthodes. De manière plus forte, il est désirable d'arriver à déterminer quelles propriétés des données sont favorables, ou défavorables, au bon fonctionnement de nos méthodes: transfert efficace, sélection adéquate. Dans le cas du QLTL en particulier, il conviendra d'étudier les effets de l'assymétrie des divergences de Bregman sur les performances de la méthode.
Une autre perspective d'importance est de combiner la sélection de sources au choix d'utiliser les méthodes de transfert ou non, et éventuellement de choisir les features à utiliser pour constituer l'espace des observations $\mathcal{X}$. Cela permettrait la création d'une architecture complète permettant une adaptation totale du détecteur à un enregistrement cible donné. Cependant, les verrous méthodologiques à une telle structure sont nombreux.

Type	Titre	Conf./Journal	Statut
Conférence	QLTL: a Simple yet Efficient Algorithm for Semi-Supervised Transfer Learning	ICPRS2019(2019)	Publié
Conférence	Influence of Data Centring in Kernel-Cross Alignment: Application to Transfer Learning	ICPRS2021	Publié(2021)
Article	Cross-Gram Matrices and their use in Transfer Learning: Application to Automatic REM Detection using Heart-Rate	Computer Methods and Programs in Biomedicine (CMPB)	Publié(2021)
Brevet	A computer-implemented method of selecting a preferred training dataset from a plurality of training datasets for a target dataset	Office européen des brevets (EPO)	Déposé

Table A.1: Liste des publications et productions scientifiques achevées durant la thèse.

Appendix B

Optimised kernel centring

As briefly introduced in 3.3 and presented in [54], we consider two kernel centring strategies:

- Centring in feature space using the representer theorem; in this case the optimisation is done w.r.t. the target weights $\alpha_i, \forall i \in \{1,\ ...,\ N_T\}$ and the source weights $\beta_j, \forall j \in \{1,\ ...,\ N_S\}$
- Centring in input space, w.r.t. a couple of a target and a source element in input space $(\mathbf{x}_{CT},\ \mathbf{x}_{CS}) \in \mathcal{X}$.

In this appendix, we give the key elements to optimised kernel centring, considering that Y_T − the target labels vector − is given and fixed (resulting from a previous step of the alternate directions optimisation scheme), and that ν is a step size parameter of the method.

B.1 Using the representer theorem

$$\underset{\alpha,\ \beta}{max}\ J(K_C) \tag{B.1}$$

We have, respectively:

$$\nabla_{c_T} J(k_C) = \sum_{i=1}^{N_T}\sum_{j=1}^{N_S} \frac{\partial J}{\partial k_C(\mathbf{x}_{T_i},\ \mathbf{x}_{S_j})} \times \nabla_{c_T} k_C(\mathbf{x}_{T_i},\ \mathbf{x}_{S_j}) \tag{B.2a}$$

$$\nabla_{c_S} J(k_C) = \sum_{i=1}^{N_T}\sum_{j=1}^{N_S} \underbrace{\frac{\partial J}{\partial k_C(\mathbf{x}_{T_i},\ \mathbf{x}_{S_j})}}_{g_{T_i S_j}} \times \nabla_{c_S} k_C(\mathbf{x}_{T_i},\ \mathbf{x}_{S_j}) \tag{B.2b}$$

Right part of the expression is of components:

$$\frac{\partial k_C(\mathbf{x}_{T_i}, \mathbf{x}_{S_j})}{\partial_{\alpha_l}} = -k(\mathbf{x}_{T_l}, \mathbf{x}_{S_j}) + \sum_{k=1}^{N_S} \beta_k k(\mathbf{x}_{T_l}, \mathbf{x}_{S_j}) \tag{B.3a}$$

$$\frac{\partial k_C(\mathbf{x}_{T_i}, \mathbf{x}_{S_j})}{\partial_{\beta_k}} = -k(\mathbf{x}_{T_i}, \mathbf{x}_{S_k}) + \sum_{l=1}^{N_T} \alpha_l k(\mathbf{x}_{T_k}, \mathbf{x}_{S_l}) \tag{B.3b}$$

We can factorise:

$$\begin{aligned}\sum_{i=1}^{N_T}\sum_{j=1}^{N_S} g_{T_i S_j} k(\mathbf{x}_{T_l}, \mathbf{x}_{S_j}) &= \sum_{j=1}^{N_S} k(\mathbf{x}_{T_l}, \mathbf{x}_{S_j}) \sum_{i=1}^{N_T} g_{ij} \\ &= \sum_{j=1}^{N_S} k(\mathbf{x}_{T_l}, \mathbf{x}_{S_j}) \mathbf{g}_{S_j}^T \mathbf{1}_{(N_T,\ 1)} \\ &= \mathbf{1}_{(1,\ N_T)} \mathbf{G} \mathbf{k}_{T_l}\end{aligned} \tag{B.4a}$$

where $\mathbf{1}_{(1,\ N_T)}$ is the all-ones vector of size $1 \times N_T$ and

$$\mathbf{k}_{T_l} = \begin{pmatrix} k(\mathbf{x}_{T_l},\ \mathbf{x}_{S_1}) \\ \vdots \\ k(\mathbf{x}_{T_l},\ \mathbf{x}_{S_{N_S}}) \end{pmatrix} \quad \mathbf{g}_{S_j} = \begin{pmatrix} g_{T_1 S_j} \\ \vdots \\ g_{T_{N_T} S_j} \end{pmatrix} \quad \mathbf{G} = \begin{pmatrix} \mathbf{g}_{S_1} & \cdots & \mathbf{g}_{S_{N_S}} \end{pmatrix}$$

Equivalently, we obtain:

$$\sum_{i=1}^{N_T} \sum_{j=1}^{N_S} g_{ij} k(\mathbf{x}_{T_i}, \mathbf{x}_{S_k}) = \mathbf{k}_{S_k} \mathbf{G} \mathbf{1}_{(N_S,\ 1)} \tag{B.4b}$$

By extension, taking into consideration the dependencies on $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, it becomes apparent that:

$$\nabla_{c_T} J(k_C) = K_{TS}(\lambda \boldsymbol{\beta} - \mathbf{G}^T \mathbf{1}_{(N_T,\ 1)}) \tag{B.5a}$$

$$\nabla_{c_S} J(k_C) = K_{TS}^T(\lambda \boldsymbol{\alpha} - \mathbf{G} \mathbf{1}_{(N_S,\ 1)}) \tag{B.5b}$$

where $\lambda = \sum_{i=1}^{N_T} \sum_{j=1}^{N_S} g_{T_i S_j}$. For the gradient step size η, we update the centres through:

$$\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \eta K_{TS}(\lambda \boldsymbol{\beta} - \mathbf{G}^T \mathbf{1}_{(N_T,\ 1)}) \equiv \boldsymbol{\alpha} + \boldsymbol{\Delta\alpha} \tag{B.6a}$$

$$\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} + \eta K_{TS}^T(\lambda \boldsymbol{\alpha} - \mathbf{G} \mathbf{1}_{(N_S,\ 1)}) \equiv \boldsymbol{\beta} + \boldsymbol{\Delta\beta} \tag{B.6b}$$

B.2 Using elements of input space

We remind the expression of an element of the centred cross-Gram matrix:

$$K_{C_{ij}} = \langle \phi(\mathbf{x}_{T_i}) - \phi(\mathbf{x}_{C_T}),\ \phi(\mathbf{x}_{S_j}) - \phi(\mathbf{x}_{C_S}) \rangle_F \tag{B.7}$$

This lead to the following expression for the whole matrix:

$$\begin{aligned} K_C = K_{TS} - K_{TC_S} \mathbf{1}_{(1,N_S)} - \mathbf{1}_{(N_T,1)} K_{C_T S} \\ + K_{C_T C_S} \mathbf{1}_{(N_T,N_S)} \end{aligned} \tag{B.8}$$

where K_{TC_S}, $K_{C_T S}$, $K_{C_T C_S}$ are the Gram elements computed between target data and source centre (column vector), target centre and source data (row vector), and both centres (scalar), respectively. **Optimisation w.r.t. target and source centres in input space depends on the kernel function**. We here propose the resolution of this optimisation problem for a Gaussian kernel of parameter σ, that is:

$$k(\mathbf{x}_1,\ \mathbf{x}_2) = \exp\big(-\frac{||\mathbf{x}_1 - \mathbf{x}_2||^2}{2\sigma^2}\big) \tag{B.9}$$

We have:

$$\nabla_{\mathbf{x}_{C_S}} k_C(\mathbf{x}_{T_i}, \mathbf{x}_{S_j}) = \frac{K_{T_i C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{T_i}) - K_{C_T C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{C_T})}{\sigma^2} \tag{B.10a}$$

$$\nabla_{\mathbf{x}_{C_T}} k_C(\mathbf{x}_{T_i}, \mathbf{x}_{S_j}) = \frac{K_{C_T S_j}(\mathbf{x}_{C_T} - \mathbf{x}_{S_j}) - K_{C_T C_S}(\mathbf{x}_{C_T} - \mathbf{x}_{C_S})}{\sigma^2} \tag{B.10b}$$

which, once put in matrix form, and using step size parameter η, yields the source centre update:

$$\begin{aligned} \mathbf{x}_{C_S} \leftarrow \mathbf{x}_{C_S} + \frac{\eta}{\sigma^2} \Bigg(& - \mathbf{1}_{(1,N_T)} \mathbf{G} \mathbf{1}_{(N_S,1)} K_{C_T C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{C_T}) \\ & + \sum_{i=1}^{N_T} \mathbf{G}_{i*} \mathbf{1}_{(N_S,1)} K_{T_i C_S}(\mathbf{x}_{C_S} - \mathbf{x}_{T_i}) \Bigg) \end{aligned} \tag{B.11a}$$

and equivalently for the target centre:

$$\mathbf{x}_{C_T} \leftarrow \mathbf{x}_{C_T} + \frac{\eta}{\sigma^2}\Bigg(-\mathbf{1}_{(1,N_T)}\mathbf{G}\mathbf{1}_{(N_S,1)}K_{C_TC_S}(\mathbf{x}_{C_T} - \mathbf{x}_{C_S}) + \sum_{j=1}^{N_S}\mathbf{1}_{(1,N_T)}\mathbf{G}_{*j}K_{C_TS_j}(\mathbf{x}_{C_T} - \mathbf{x}_{S_j})\Bigg) \tag{B.11b}$$

where $\mathbf{G}$ is the gradient matrix.

B.3 Elements of the gradient matrix

We remind there that the gradient matrix depends on the choice of the function to optimise. Our objective throughout our research is to optimise kernel-cross alignment (KCA):

$$\begin{aligned}\mathbf{G} &= \nabla_{K_C}\frac{Y_T^T K_C Y_S}{||K_C||_F||Y_S||\,||Y_T||} \\ &= \frac{Y_T Y_S^T}{||K_C||_F||Y_T||\,||Y_S||} - \frac{Y_T^T K_C Y_S||Y_T||\,||Y_S||K_C}{(||K_C||_F||Y_T||\,||Y_S||)^2||K_C||_F} \\ &= \frac{Y_T Y_S^T}{||K_C||_F||Y_T||\,||Y_S||} - \mathbf{KCA}(T,\,S)\frac{K_C}{||K_C||_F^2}\end{aligned} \tag{B.12}$$

In the case of the non-normalised equivalent of KCA, namely the kernel-cross polarisation (KCP), the gradient matrix expresses as:

$$\mathbf{G} = Y_T Y_S^T \tag{B.13}$$

Bibliography

[1] M. Deak and L. J. Epstein, "The history of polysomnography," *Sleep Medicine Clinics*, vol. 4, no. 3, pp. 313–321, 2009.

[2] R. B. Berry, R. Brooks, C. E. Gamaldo, S. M. Harding, C. Marcus, and B. V. Vaughn, "The AASM manual for the scoring of sleep and associated events," *Rules, Terminology and Technical Specifications, Darien, Illinois, American Academy of Sleep Medicine*, vol. 176, 2012.

[3] O. B. Le and S. Arpi, "Effect of the first sleep night in polysomnography: classification by variable sensitivity and factorial analysis of differences between nights," *Revue neurologique*, vol. 159, no. 11 Suppl, pp. 6S42–7, 2003.

[4] P. Anderer, G. Gruber, S. Parapatics, M. Woertz, T. Miazhynskaia, G. Klösch, B. Saletu, J. Zeitlhofer, M. J. Barbanoj, and H. Danker-Hopfe, "An E-health solution for automatic sleep classification according to Rechtschaffen and Kales: validation study of the Somnolyzer 24x7 utilizing the Siesta database," *Neuropsychobiology*, vol. 51, no. 3, pp. 115–133, 2005.

[5] P. J. Arnal, V. Thorey, M. E. Ballard, A. B. Hernandez, A. Guillot, H. Jourde, M. Harris, M. Guillard, P. Van Beers, M. Chennaoui, and F. Sauvet, "The Dreem headband as an alternative to polysomnography for EEG signal acquisition and sleep staging," *bioRxiv*, p. 662734, 2019.

[6] J. Trinder, J. Kleiman, M. Carrington, S. Smith, S. Breen, N. Tan, and Y. Kim, "Autonomic activity during human sleep as a function of time and sleep stage," *Journal of Sleep Research*, vol. 10, no. 4, pp. 253–264, 2001.

[7] M. de Zambotti, A. Goldstone, S. Claudatos, I. M. Colrain, and F. C. Baker, "A validation study of Fitbit charge 2™ compared with polysomnography in adults," *Chronobiology international*, vol. 35, no. 4, pp. 465–476, 2018.

[8] H. O. Kinnunen and H. Koskimäki, "The HRV of the ring - comparison of nocturnal HR and HRV between a commercially available wearable ring and ECG.," *Sleep*, vol. 41, no. 1, pp. A120–A120, 2018.

[9] M. Migliorini, A. M. Bianchi, D. Nisticò, J. Kortelainen, E. Arce-Santana, S. Cerutti, and M. O. Mendez, "Automatic sleep staging based on ballistocardiographic signals recorded through bed sensors," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 3273–3276, IEEE, 2010.

[10] M. Quante, E. R. Kaplan, M. Cailler, M. Rueschman, R. Wang, J. Weng, E. M. Taveras, and S. Redline, "Actigraphy-based sleep estimation in adolescents and adults: a comparison with polysomnography using two scoring algorithms," *Nature and science of sleep*, vol. 10, p. 13, 2018.

[11] J. Hedner, D. P. White, A. Malhotra, S. Herscovici, S. D. Pittman, D. Zou, L. Grote, and G. Pillar, "Sleep staging based on autonomic signals: a multi-center validation study," *Journal of clinical sleep medicine*, 2011.

[12] A. Muzet, S. Werner, G. Fuchs, T. Roth, J. B. Saoud, A. U. Viola, J. Y. Schaffhauser, and R. Luthringer, "Assessing sleep architecture and continuity measures through the analysis of heart rate and wrist movement recordings in healthy subjects: comparison with results based on polysomnography," *Sleep Med*, vol. 21, pp. 47–56, 2016.

[13] F. Ebrahimi, S. K. Setarehdan, J. Ayala-Moyeda, and H. Nazeran, "Automatic sleep staging using empirical mode decomposition, discrete wavelet transform, time-domain, and nonlinear dynamics features of heart rate variability signals," *Computer Methods and Programs in Biomedicine*, vol. 112, no. 1, pp. 47–57, 2013.

[14] M. O. Mendez, M. Matteucci, V. Castronovo, L. F. Strambi, S. Cerutti, and A. M. Bianchi, "Sleep staging from heart rate variability: time-varying spectral features and hidden markov models," *International Journal of Biomedical Engineering and Technology*, vol. 3, no. 3/4, pp. 246–263, 2010.

[15] T. Willemen, D. Van Deun, V. Verhaert, M. Vandekerckhove, V. Exadaktylos, J. Verbraecken, S. Van Huffel, B. Haex, and J. V. Sloten, "An evaluation of cardiorespiratory and movement features with respect to sleep-stage classification," *IEEE journal of biomedical and health informatics*, vol. 18, no. 2, pp. 661–669, 2014.

[16] A. Smola, "Maximum mean discrepancy," in *13th International Conference on Neural Information Processing (ICONIP 2006)*, 2006.

[17] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[18] R. S. Rosenberg and S. Van Hout, "The american academy of sleep medicine inter-scorer reliability program: sleep stage scoring," *J Clin Sleep Med*, vol. 9, no. 1, pp. 81–7, 2013.

[19] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[20] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, 2016.

[21] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *arXiv preprint arXiv:1911.02685*, 2019.

[22] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[23] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR computational mathematics and mathematical physics*, vol. 7, no. 3, pp. 200–217, 1967.

[24] H. Daumé III, "Frustratingly easy domain adaptation," *arXiv preprint arXiv:0907.1815*, 2009.

[25] T. Tommasi and B. Caputo, "The more you know, the less you learn: from knowledge transfer to one-shot learning of object categories," in *BMVC*, 2009.

[26] F. Zhuang, P. Luo, H. Xiong, Q. He, Y. Xiong, and Z. Shi, "Exploiting associations between word clusters and document classes for cross-domain text categorization," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 4, no. 1, pp. 100–114, 2011.

[27] F. Zhuang, P. Luo, C. Du, Q. He, Z. Shi, and H. Xiong, "Triplex transfer learning: Exploiting both shared and distinct concepts for text classification," *IEEE transactions on cybernetics*, vol. 44, no. 7, pp. 1191–1203, 2013.

[28] X. Chen and R. Lengellé, "Domain Adaptation Transfer Learning by SVM Subject to a Maximum-Mean-Discrepancy-like Constraint," in *6th International Conference on Pattern Recognition Applications and Methods*, (Porto, Portugal), pp. 89–95, SCITEPRESS - Science and Technology Publications, Feb. 2017.

[29] X. Chen and R. Lengellé, "Domain Adaptation Transfer Learning by Kernel Representation Adaptation," in *Lecture Notes In Computer Science* (D. M. M., di Baja G., and F. A., eds.), vol. 10857, pp. 45–61, June 2018.

[30] W. M. Kouw and M. Loog, "A review of domain adaptation without target labels," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 3, 2021.

[31] I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani, "A survey on domain adaptation theory," *arXiv preprint arXiv:2004.11829*, 2020.

[32] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," *arXiv preprint arXiv:1705.08848; proceedings of NIPS*, 2017.

[33] S. Herath, M. Harandi, and F. Porikli, "Learning an invariant hilbert space for domain adaptation," pp. 3845–3854, 2017.

[34] J. Liang, R. He, Z. Sun, and T. Tan, "Distant supervised centroid shift: A simple and efficient approach to visual domain adaptation," pp. 2975–2984, 2019.

[35] S. Eetemadi, W. Lewis, K. Toutanova, and H. Radha, "Survey of data-selection methods in statistical machine translation," *Machine Translation*, vol. 29, no. 3-4, pp. 189–223, 2015.

[36] C. Passini, M. Luiza, K. Estébanez, G. Figueredo, F. Ebecken, and F. Nelson, "A strategy for training set selection in text classification problems," *International Journal of Advanced Computer Science & Applications*, vol. 4, no. 6, 2013.

[37] I. Kuzborskij, F. Orabona, and B. Caputo, "Transfer learning through greedy subset selection," in *International Conference on Image Analysis and Processing*, pp. 3–14, Springer, 2015.

[38] S. Saitoh, "Theory of reproducing kernels and its applications," *Longman Scientific & Technical*, 1988.

[39] R. Courant and D. Hilbert, "Methods of mathematical physics," tech. rep., 1966.

[40] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.

[41] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[42] V. Vapnik, "Principles of risk minimization for learning theory," in *Advances in neural information processing systems*, pp. 831–838, 1992.

[43] G. S. Kimeldorf and G. Wahba, "A correspondence between bayesian estimation on stochastic processes and smoothing by splines," *The Annals of Mathematical Statistics*, vol. 41, no. 2, pp. 495–502, 1970.

[44] G. Kimeldorf and G. Wahba, "Some results on tchebycheffian spline functions," *Journal of mathematical analysis and applications*, vol. 33, no. 1, pp. 82–95, 1971.

[45] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *International conference on computational learning theory*, pp. 416–426, Springer, 2001.

[46] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. S. Kandola, "On kernel-target alignment," in *Advances in neural information processing systems*, pp. 367–373, 2002.

[47] T. Wang, D. Zhao, and S. Tian, "An overview of kernel alignment and its applications," *Artificial Intelligence Review*, vol. 43, no. 2, pp. 179–192, 2015.

[48] I. Redko and Y. Bennani, "Kernel alignment for unsupervised transfer learning," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 525–530, IEEE, 2016.

[49] J.-B. Pothin and C. Richard, "Optimizing kernel alignment by data translation in feature space," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 3345–3348, IEEE, 2008.

[50] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *Journal of Machine Learning Research*, vol. 13, no. Mar, pp. 795–828, 2012.

[51] M. Meila, "Data centering in feature space.," in *AISTATS*, Citeseer, 2003.

[52] B. Muller and R. Lengelle, "QLTL: a simple yet efficient algorithm for semi-supervised transfer learning," in *10th International Conference on Pattern Recognition Systems (ICPRS2019)*, pp. 30–35, 2019.

[53] B. Muller and R. Lengellé, "Cross-Gram matrices and their use in transfer learning: Application to automatic REM detection using heart rate," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106280, 2021.

[54] B. Muller and R. Lengellé, "Influence of data centring in kernel-cross alignment: Application to transfer learning," in *11th International Conference on Pattern Recognition Systems (ICPRS 2021)*, vol. 2021, pp. 194–199, IET, 2021.

[55] B. Muller and R. Lengelle, "A computer-implemented method of selecting a preferred training dataset from a plurality of training datasets for a target dataset," in *Filed at the European Patent Office*, 2021.

[56] P. Honeine, "An eigenanalysis of data centering in machine learning," *arXiv preprint arXiv:1407.2904*, 2014.

[57] Q. Zhou, W. Chen, S. Song, J. Gardner, K. Weinberger, and Y. Chen, "A reduction of the elastic net to support vector machines with an application to GPU computing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.

[58] M. Jaggi, "An equivalence between the lasso and support vector machines," *Regularization, optimization, kernels, and support vector machines*, pp. 1–26, 2013.

[59] Q. Li, N. Lin, *et al.*, "The bayesian elastic net," *Bayesian analysis*, vol. 5, no. 1, pp. 151–170, 2010.

Bruno MULLER

Doctorat : Optimisation et Sureté des Systèmes

Année 2021

Transfert d'apprentissage par alignement de noyau : classification automatique des stades du sommeil

L'objectif de cette thèse est l'amélioration d'un système de classification automatique des stades de sommeil par la prise en compte des variabilités inter-et-intra individuelles, ces dernières ayant un effet détrimentaire sur la classification. Nous nous intéressons en particulier à la détection des épisodes de sommeil paradoxal au cours de la nuit. Nos recherches se focalisent sur le transfert d'apprentissage et la sélection de détecteurs adaptés, permettant l'individualisation de l'analyse par l'exploitation des propriétés des données observées. Nous avons focalisé nos travaux sur l'application des méthodes d'alignement de noyau, dans un premier temps au travers de l'utilisation du *kernel-target alignment* étudié ici de manière duale, c'est-à-dire à noyau fixé et optimisé par rapport aux étiquettes recherchées des données de tests. Dans un second temps, nous avons introduit le *kernel-cross alignment*, permettant d'exploiter plus fortement l'information contenue dans les données d'apprentissage. Les idées développées dans le cadre de ces travaux ont été étendues à la sélection automatique d'un ensemble d'apprentissage adapté à un ensemble de test donné. Les contributions de ces travaux sont à la fois méthodologiques et algorithmiques, à portée générale, mais également centrées sur l'application.

Mots clés : apprentissage automatique – méthodes à noyau – sommeil, stades, classification – sommeil paradoxal.

Transfer Learning through Kernel Alignment: Application to Adversary Data Shifts in Automatic Sleep Staging

This doctoral project aims at improving an automatic sleep staging system by taking into account inter-and-intra-individual variabilities, the latter having adversary effects on the classification. We focus on the detection of Rapid-Eye Movement periods during sleep. The core of our research is transfer learning and the selection of suitable detector(s) among a set, allowing the individualisation of the analysis by the exploitation of the observed data properties. We focus on the application of kernel alignment methods, firstly through the use of kernel-target alignment, studied here in a dual way, i.e. the kernel is fixed and the criterion is optimised with respect to the sought target labels. In a second step, we introduced kernel-cross alignment, allowing to take more efficiently advantage of the information contained in the training data. The ideas developed in the framework of this work have been extended to automatically selecting one or more efficient training sets for a given test set. The contributions of this work are both methodological and algorithmic, general in scope, but also focused on the application.

Keywords: machine learning – Kernel functions – sleep, stages – rapid eye movement sleep.

Thèse réalisée en partenariat entre :

PPRS
THE STRATEGIC PARTNERING ORGANIZATION

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

Ecole Doctorale "Sciences pour l'Ingénieur"