THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique
Spécialité : Mathématiques Appliquées
Unité de recherche : Laboratoire d'Informatique de Grenoble

garanties théoriques et amélioration de la classification multi-tâches semi-supervisée en grande dimension

theoretical guarantees and improvement of large dimensional multi-task semi-supervised classification

Présentée par :

Victor LEGER

Direction de thèse :

Romain COUILLET — Directeur de thèse
PROFESSEUR DES UNIVERSITES, Université Grenoble Alpes

Rapporteurs :

Pierre BORGNAT
DIRECTEUR DE RECHERCHE, Ecole Normale Supérieure de Lyon
Guillaume GINOLHAC
PROFESSEUR DES UNIVERSITES, Polytech Annecy-Chambéry

Thèse soutenue publiquement le **14 novembre 2024**, devant le jury composé de :

Romain COUILLET, PROFESSEURE DES UNIVERSITES, Université Grenoble Alpes	Directeur de thèse
Pierre BORGNAT, DIRECTEUR DE RECHERCHE, Ecole Normale Supérieure de Lyon	Rapporteur
Guillaume GINOLHAC, PROFESSEUR DES UNIVERSITES, Polytech Annecy-Chambéry	Rapporteur
Abla KAMMOUN, SENIOR SCIENTIST, King Abdullah University of Science and Technology	Examinatrice
Paulo GONçALVES, DIRECTEUR DE RECHERCHE, Ecole Normale Supérieure de Lyon	Examinateur
Florent CHATELAIN, MAITRE DE CONFERENCES, Université Grenoble Alpes	Examinateur
Jean-François COEURJOLLY, PROFESSEUR DES UNIVERSITES, Université Grenoble Alpes	Examinateur

Theoretical Guarantees and Improvement of Large Dimensional Multi-Task Semi-Supervised Classification

Victor Léger

Abstract

In the field of machine learning, the specific subfield of deep learning has gathered particular interest in the last decade. If deep learning has allowed quick and significant progress in a wide range of fields, this progress has been made at the expense of interpretability, accessibility, robustness and flexibility, not to mention the consecutive rebound effects of data center deployment and energy consumption implied by the training of such algorithms. In this context, the present manuscript aims on the contrary to open the path to tractable and flexible tools for classification problems, buttressed on elementary machine learning notions and rather basic mathematical tools.

This thesis conducts a large dimensional study of a simple yet quite versatile classification model, encompassing at once multi-task and semi-supervised learning, and taking into account uncertain labeling. Using tools from random matrix theory, the asymptotics of some key functionals are characterized, which allows on the one hand to predict the performances of the proposed algorithm, and on the other hand to reveal some counter-intuitive guidance on how to use it efficiently. The model, powerful enough to provide good performance guarantees, is also straightforward enough to provide strong insight into its behavior. The resulting algorithm is also compared to an optimal bound derived from statistical physics, which gives a lower bound of the least achievable probability of misclassification for a given problem. This bound is computed in the extended case of uncertain labeling, and is used to evaluate the performances of the algorithm.

Résumé

Dans le domaine de l'apprentissage machine, le sous-domaine spécifique de l'apprentissage profond a fait l'objet d'un intérêt particulier au cours de la dernière décennie. Si l'apprentissage profond a permis des avancées significatives dans de nombreux domaines, ces avancées se sont faites au détriment de l'interprétabilité, de l'accessibilité, de la robustesse et de la flexibilité, sans parler des effets rebonds associés, en terme de déploiement de centre de données et de consommation énergétique, induits par l'entraînement de tels algorithmes. Dans ce contexte, l'objectif de ce manuscrit est à l'inverse d'ouvrir la voie à des outils maîtrisables et flexibles pour résoudre des problèmes de classification, qui s'appuient sur des notions élémentaires d'apprentissage statistique et des outils mathématiques assez accessibles.

Cette thèse mène une analyse statistique en grande dimension d'un modèle de classification à la fois simple et versatile, qui unifie dans un même modèle l'apprentissage multi-tâches et l'apprentissage semi-supervisé, et qui prend en compte la possibilité d'étiqueter les données de manière incertaine. En utilisant des outils issus de la théorie des matrices aléatoires, les statistiques asymptotiques de certaines fonctions clés sont caractérisées, ce qui permet d'une part de prédire les performances de l'algorithme proposé, et d'autre part de révéler certaines astuces contre-intuitives sur la manière de l'utiliser efficacement. Le modèle, suffisamment puissant pour donner de bonnes garanties de performance, est aussi suffisamment lisible pour apporter de bonnes intuitions sur son fonctionnement. L'algorithme produit est également comparé à une borne optimale issue de la physique statistique, qui donne une borne inférieure de la plus petite probabilité d'erreur atteignable pour un problème donné. Cette borne est calculée dans le cas étendu de l'étiquetage incertain, et est utilisée pour évaluer les performances de l'algorithme.

Contents

0.1 List of Acronyms

ML Machine Learning

AI Artificial Intelligence

DL Deep Learning

DNN Deep Neural Networks

RMT Random Matrix Theory

CLT Central Limit Theorem

ESD Empirical Spectral Distribution

GAN Generative Adversarial Networks

RS Replica Symmetric

SNR Signal-to-Noise Ratio

MTL Multi-Task Learning

SSL Semi-Supervised Learning

1 Introduction

1.1 The current trend of Machine Learning

Machine Learning (ML) is a field of Artificial Intelligence (AI) which consists in designing and studying statistical algorithms that learn from data how to complete the task they are assigned to. The particularity of these statistical algorithms is that even though they are coded at first by a human being, their final behavior is mostly guided by the input data they are trained on. In the field of ML, one specific subfield has gathered particular interest in the last decade, namely Deep Learning (DL), and in particular Deep Neural Networks (DNN) [1]. Thanks to their multiple layers, DNN allow a high level of expressiveness and are able to build advanced abstract representations of the data.

No one missed out on the impressive achievements of DL during the past few years. These technologies have allowed quick and significant progress in a wide range of fields, as image classification, speech recognition, translation or gene prediction. However, by taking another look at the narrative of DL, one must come to the deceiving conclusion that these results are mostly due to increases in available computer power [2]. That is, instead of gaining ground in the theoretical understanding of DL techniques, the research effort focused on building larger models, with exponentially higher numbers of parameters.

This escalation necessarily implies extensive energy and computer power consumption, not to mention the unprecedented exploitation of mineral resources and fossil fuels necessary to build the digital infrastructures they rely on [3]. These requirements are at odds with the current and forthcoming socio-environmental crisis, which imposes us to reduce our overall energy consumption. Large models also require high amounts of data, which come at a cost as well. Designing proper datasets (labeling the data, handling potential biases in the datasets, dealing with uncomplete datasets, harmonizing heterogeneous datasets...) is often a tedious task, which has to be done by human beings, with all the ethical issues that it implies. A striking example is the use of kenyans underpaid laborers by OpenAI to make sure that their chatbot model chatGPT would not formulate violent, sexist or racist sentences [4].

More fundamentally, because of their design, DNN are untractable and hard to analyze theoretically. In the litterature, the theoretical understanding of such algorithms seems to be limited to very basic results, like the universal approximation theorem [5]. This comes from their highly non-linear behavior, which is precisely the reason of their success. On the one hand, the multiple layers allow a high level of expressiveness, and therefore a really accurate output. But on the other hand, the complexity of the mechanism which gives this output is not decipherable by a human being. Thus, DNN being far from technical accessibility for their own designers, it is clear that it cannot be the case for their users. Such a tool is at the opposite of what Ivan Illich defines as a *convivial* tool [6].

A convivial tool is at once robust, reliable, increases the power of action of the user without making them dependant, is accessible to most, and does not require many resources. With these metrics in mind, it becomes obvious that not only DL, but more generally ML and even digital technologies are not convivial tools. That being said, given the context of this thesis, we will aim to open the path to convivial ML algorithms, as contradictory as this might sound. That is, designing sufficient algorithms, which only match their purpose, do not create new needs, are built on elementary ML notions and do not need additional scientific complexification. Such algorithms must be interpretable, understandable by the user and flexible enough to be adapted to the needs they answer, while still having decent performances.

As said before, DNN are difficult to analyze mathematically. But surprisingly, even classical ML algorithms suffer from severe theoretical limitations. Most of these algorithms are unable to deal properly with large dimensional data. Indeed, most of these algorithms have been designed over heuristics that only make sense in a low dimensional setting. From this point of view, the unexpected behavior of large dimensional data is often reffered to as an example of the *curse of dimensionality*. However, tools from Random Matrix Theory (RMT) have proven to be useful to understand partially the behavior of such algorithms [7]. Most importantly, RMT brings some intuition over the statistical behavior of data in the large dimensional regime, which is at the core of modern ML.

1.2 The large dimensional regime

The usual assumption made to analyze the performance of ML algorithms is that $n \to \infty$, where n denotes the number of data samples. In this limit, the algorithm is expected to have enough information to classify the data adequately. To perform

such analysis of ML algorithms, the input data is modeled as random variables. When $n \to \infty$, the behavior of these random variables becomes asymptotically deterministic, in particular through the law of large numbers and the Central Limit Theorem (CLT). If p denotes the dimension of the data, which is assumed to be small in that setting, this actually means that $\frac{n}{p} \to \infty$. For decades, asymptotic results have been obtained within this framework of low-dimensional data [8]. However, in modern datasets, p is large as well, too large to be considered negligeable in comparison with n. A more realistic assumption would be to consider that n and p have the same order of magnitude. But this assumption discards the previous analysis of algorithms. Indeed, as it has been shown decades ago, the deterministic behavior mentionned above do not hold when p and n have the same magnitude [9, 10].

To understand the counter-intuitive effects of a large dimensional setting, let us consider a mere model of binary Gaussian mixture, which is the simplest possible mixture model. In this model, the data follows the distribution $\mathbf{x} \sim \mathcal{N}(\pm\boldsymbol{\mu}, \mathbf{I}_p)$, representing two classes of data with the same isotropic covariance. For small values of p, data from this model can be visualized as two groups of datapoints, one being centered around $\boldsymbol{\mu}$ and the other one being centered around $-\boldsymbol{\mu}$. This setting allows an easy and convenient visualization, which is at the core of most ML algorithms.

Let us consider the specific case of kernel-based algorithms. To build a classifier, these algorithms compute a similarity function over all pairs of data points, and use these pairwise comparisons to extract the structural data information they need. For practical reasons, most of the generic affinity functions used in kernel-based algorithms are $f(\|\mathbf{x}_i - \mathbf{x}_j\|^2)$ or $f(\mathbf{x}_i^\intercal \mathbf{x}_j)$, where f is a nonincreasing function. Such kernel functions, which are often referred to as Mercer kernels, share some theoretically convenient properties [11]. The idea behind these choices is the following: if samples $\mathbf{x}_i$ and $\mathbf{x}_j$ are "close", they must have a high affinity, while "distant" samples must have a low affinity. One of the most popular choice of such function is $f(t) = e^{-\frac{t}{2}}$, which is referred to as *radial basis function*, or *heat kernel*. One of its biggest benefit is that the feature space of this kernel has an infinite number of dimensions, giving it a strong power of representation.

However, in high dimension, the Euclidean distance does not behave as expected. Let us consider some i.i.d. samples $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}_p, \mathbf{I}_p)$. For such samples, the squared Euclidean norm $\|\mathbf{z}_i\|^2$ is the sum of p squared standard normal samples, so it follows the χ^2 distribution, which is asymptotically equivalent to the distribution

$\|\mathbf{z}_i\|^2 \sim \mathcal{N}(p, 2p)$. Similarly, the squared distance between two distinct samples follows the distribution $\|\mathbf{z}_i - \mathbf{z}_j\|^2 \sim \mathcal{N}(2p, 8p)$. By normalizing, we obtain:

$$\frac{1}{p}\|\mathbf{z}_i - \mathbf{z}_j\|^2 \sim \mathcal{N}\left(2, \frac{8}{p}\right). \tag{1.1}$$

Therefore, in the limit $p \to \infty$, we have the following result:

$$\max_{1 \leq i \neq j \leq n} \left\{ \frac{1}{p}\|\mathbf{z}_i - \mathbf{z}_j\|^2 - 2 \right\} \to 0. \tag{1.2}$$

Going back to the binary Gaussian mixture model discribed above, the normalized distance between samples is

$$\frac{1}{p}\|\mathbf{x}_i - \mathbf{x}_j\|^2 = \begin{cases} \frac{1}{p}\|\mathbf{z}_i - \mathbf{z}_j\|^2 & \text{if } \mathcal{C}_i = \mathcal{C}_j \\ \frac{1}{p}\|\mathbf{z}_i - \mathbf{z}_j\|^2 \pm \frac{4}{p}(\mathbf{z}_i - \mathbf{z}_j)^\intercal \boldsymbol{\mu} + \frac{4}{p}\|\boldsymbol{\mu}\|^2 & \text{if } \mathcal{C}_i \neq \mathcal{C}_j \end{cases} \tag{1.3}$$

where $\mathcal{C}_i$ denotes the class the sample $\mathbf{x}_i$ belongs to. In order for the classification to be feasible and not trivial, we make the assumption, as in [12], that $\|\boldsymbol{\mu}\|^2 \sim O(1)$. Therefore, the term $\frac{4}{p}\|\boldsymbol{\mu}\|^2$ is of order $O(\frac{1}{p})$, as well as $\frac{4}{p}(\mathbf{z}_i - \mathbf{z}_j)^\intercal \boldsymbol{\mu}$. Consequently, in the limit $p \to \infty$, the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is dominated by the noise rather than the signal. To go further, because of the CLT, the noise term follows $\frac{1}{p}\|\mathbf{z}_i - \mathbf{z}_j\|^2 = 2 + O(\frac{1}{\sqrt{p}})$, while the information term is of order $O(\frac{1}{p})$. Therefore, the pairwise distances between data points do not contain any information that can be exploited statistically, and all the heuristics behind kernel-based algorithms collapse. Thus, the convenient theoretical and practical properties of Mercer kernels seem to be of no interest when it comes to high dimensional data. On the contrary, simpler kernels, such as the linear kernel, will turn out to outperform them.

More precisely, *regardless of the class they belong to*, any pair of data points $\mathbf{x}_i$ and $\mathbf{x}_j$ will lay at the same distance from each other :

$$\max_{1 \leq i \neq j \leq n} \left\{ \frac{1}{p}\|\mathbf{x}_i - \mathbf{x}_j\|^2 - 2 \right\} \to 0. \tag{1.4}$$

This phenomenon is known as the *concentration of distances* [12]. To put it in a more visual way, let us consider the behavior of the Euclidean norm $\|\mathbf{z}\|$ of a sample in high dimension. As a reminder, $\frac{1}{p}\|\mathbf{z}\|^2 \sim \mathcal{N}(1, \frac{2}{p})$. Using a series expansion, we have asymptotically $\frac{1}{\sqrt{p}}\|\mathbf{z}\| \sim \mathcal{N}(1, \frac{1}{2p})$. Then, we obtain $\|\mathbf{z}\| \sim \mathcal{N}(\sqrt{p}, \frac{1}{2})$, which means that $\|\mathbf{z}\|$ has an order of magnitude $\|\mathbf{z}\| \sim O(\sqrt{p})$, while its spread is of order $\|\mathbf{z}\| - \mathbb{E}[\|\mathbf{z}\|] \sim O(1)$. Therefore, the data lies around a sphere of very large radius of order $O(\sqrt{p})$, while the centroid $\boldsymbol{\mu}$ of the distribution is of order $\|\boldsymbol{\mu}\|^2 \sim O(1)$, and

is therefore in an empty region of the distribution. It is equivalently far from any point of distribution, regardless of the class they belong to.

This means that, in a high dimensional regime, data pairs lose their individual discriminative power. However, by taking into account data as a whole, it is still possible to extract some information. What is required, though, is to understand ML algorithms through this new high dimensional paradigm. RMT precisely offers the mathematical tools to handle such a framework, and enables the statistical analysis of ML algorithms where n and p have comparable orders of magnitude.

1.3 Information-theoretic bounds of performance

The highly non-linear behavior of DNN mentionned above has another dramatic consequence: it makes the output of DNN completely unforeseeable. Even knowing whether the training process will converge or diverge is out of reach for DNN designers [13]. Thus, designing DNN boils down to tuning some hyperparameters through a trial and error approach. In such a process, it is impossible to know when to stop, because there is no way to predict when the performances will reach a limit. As an example, Figure 1.1 displays the performances over time of every classification model tested on Imagenet dataset [14]. The state-of-the-art performance is increasing over the years, but over the dozens of models produced every year, only a few ones are able to compete with previous state-of-the-art models. This quest of performance comes down to blind attempts to beat previous models. In such a paradigm, the performances will never be considered *sufficient*. In this headlong rush for performance, all the conviviality criterions mentionned earlier are let aside.

Even though most of the research in ML has been a blind attempt to constantly increase the performances without knowing how much actually remains to be improved, some tried a drastically opposite approach, and started to compute some physical limits of performances to a given learning framework. Specifically, a field of research has focused on analysing Gaussian mixture models with statistical physics [15–18]. Such analysis brings an optimal bound for a given problem, meaning that any possible algorithm could not reach better performances. This optimal bound is called the *Bayes risk*, *i.e.,* the minimal achievable probability of misclassification for a new data point. Similarly to the asymptotical statistics of algorithms computed through RMT, the Bayes risk turns out to converge to a deterministic value in the limit $n, p \to \infty$.

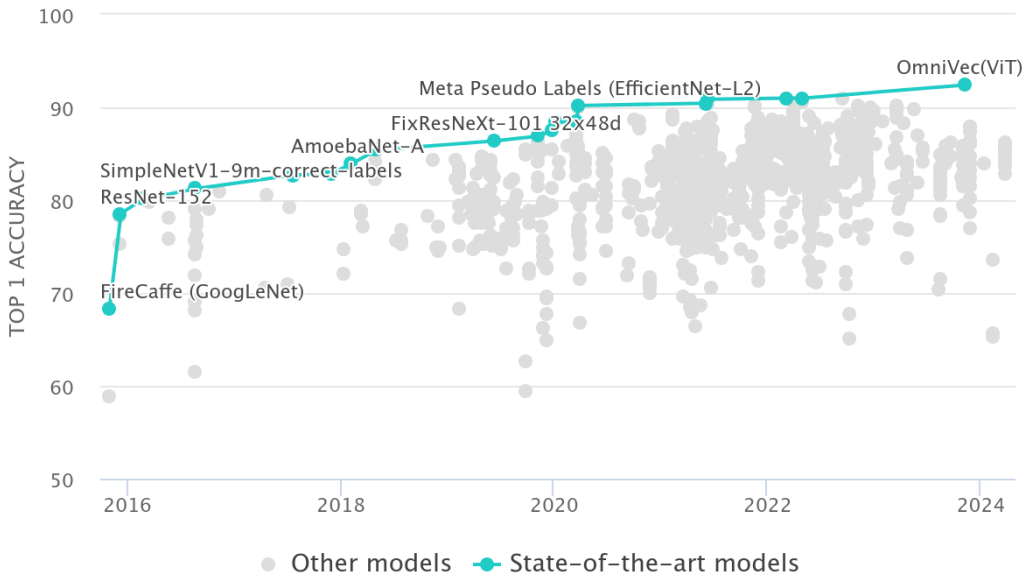

Fig. 1.1.: Accuracy over time of image classification models on ImageNet dataset

Even if performance should not be the only criterion to judge the quality of an algorithm, these optimal bounds are precious tools to understand whether an algorithm has poor performances because of its design or because of the inherent difficulty of the problem it tries to solve. By knowing in advance how far from optimal an algorithm is, one can avoid spending too much energy to improve an algorithm which is already quasi-optimal, or conversely to spot some learning settings for which the algorithm is sub-optimal. Therefore, figuring out the link between the performances of an algorithm and its optimal bound gives a precious feedback to its designer. Optimal bounds capture the very reasons for a task to be easy or hard, and can help to spot favorable learning settings.

Furthermore, the similarity of behavior between the algorithm and the Bayes risk allows to understand algorithms from an other perspective. Bayesian statistics sometimes allow some interpretation of existing algorithms [19, 20], and can even be a source of inspiration to build new ones [21, 22]. All of this contributes to build algorithms that are more interpretable and accessible. We are therefore at the opposite of the current trend of ML described before, in which constantly better performances are achieved for problems, without never questioning if the problem is relevant, well-defined, or even physically feasible. If a problem which is far from being solved can offer some good perspectives of work for a researcher, it does not mean that it is a relevant direction to be explored. This opens the path to a completely different approach of the research in ML. Instead of building the most efficient tool, through the only filter of performance, the research can work on

building a better understanding of technical tools, in order to make them more easily appropriated by a user, who is not necessarily an expert of ML.

1.4 Outline and contributions

As explained previously, this thesis aims to open the path to all-encompassing flexible tools for classification problems, buttressed on elementary ML notions and rather basic RMT results. Specifically, we propose in the course of this thesis a classification method that addresses a wide variety of real-life conditions, such as the possibility for only part of the input data to be prelabeled (thereby encompassing supervised, semi-supervised and unsupervised learning at once), the possibility for erroneous or uncertain labels (here considering biases or controversial data labeling), the possibility for strong imbalances in prelabeled classes of data (a very classical issue when some key classes are rarely observed)... As such, our base model combines what are usually considered as distinct learning frameworks (multi-task learning, semi-supervised learning, uncertain labeling) under a common umbrella. The proposed algorithm is therefore robust and versatile, as it allows to unify a wide variety of classical learning frameworks, and it can be run with low computational resources (a laptop is sufficient for all the experiments presented in this thesis). Even though the technical aspects of RMT used are not accesible for most, the key insights they bring can be easily understood and interpretable by those who are not experts of ML.

More specifically, we develop a simple and quite technically accessible method which perfoms linear classification in a multi-task and semi-supervised framework, in a basic two-class Gaussian mixture model. As mentionned earlier, we consider high dimensional data (*i.e.,* the dimension p and the number of data n are of the same order of magnitude – which is more a fact than an assumption in modern data), in contrast to classical statistics. We compare the performances of our algorithm to an optimal bound derived from statistical physics, and we compute this optimal bound in the specific case of uncertain labeling.

The next chapters are organized as follows:

- Chapter 2 introduces the main technical tools that will be required to obtain the theoretical results of this thesis. Specifically, we present some basic RMT results (in particular the Marčenko-Pastur law and the notion of deterministic equivalent), as well as some tools from statistical physics, which will be useful in Chapter 6.

- Chapter 3 presents the family of graph-based semi-supervised learning algorithms, and how they can be adapted to our multi-task framework. We state our assumptions, and explain some of the fundamental choices made thanks to previous analyses of such models. Both regularized and unregularized solutions are then computed, which will turn out to be useful in the following chapters. The proposed method presented in this chapter is based on the following articles:

  V. Léger and R. Couillet, "A large dimensional analysis of multi-task semi-supervised learning", *submitted to* IEEE transactions on signal processing, 2024.

  V. Léger, M. Tiomoko and R. Couillet, "Classification multi-tâches semi-supervisée en grande dimension", GRETSI, Nancy, 2022.

- In Chapter 4, we perform the large dimensional analysis of the optimization problem stated in Chapter 3, with the previously introduced RMT tools. Thanks to this theoretical analysis, we propose some fundamental improvements of the algorithm, and in particular the counter-intuitive choice of label values. We conclude with some experiments on our newly created algorithm. This analysis is presented in the articles:

  V. Léger and R. Couillet, "A large dimensional analysis of multi-task semi-supervised learning" *submitted to* IEEE transactions on signal processing, 2024.

  V. Léger, M. Tiomoko and R. Couillet, "Classification multi-tâches semi-supervisée en grande dimension", GRETSI, Nancy, 2022.

- In Chapter 5, we aim to extend the results of Chapter 4 to the case of erroneous or uncertain data. In particular, it implies to reconsider the previous framework of labeled and unlabeled data. We perform once again the statistical analysis of this modified framework, and we conlude with some experiments over uncertain data. This new framework and associated experiments are presented in the articles:

  V. Léger and R. Couillet, "A large dimensional analysis of multi-task semi-supervised learning", *submitted to* IEEE transactions on signal processing, 2024.

  V. Léger and R. Couillet, "Apprentissage semi-supervisé avec données partiellement étiquetées", GRETSI, Grenoble, 2023.

- Chapter 6 is focused on deriving an asymptotic optimal bound in the case of uncertain labeling. If the Bayes risk was already known for the multi-task semi-supervised model of Chapter 4, it is not the case for the model of Chapter 5. This computation is based on this article:

  V. Léger and R. Couillet, “Asymptotic Bayes risk of semi-supervised learning with uncertain labeling”, European Signal Processing Conference (EUSIPCO), Lyon, 2024.

2 Technical tools

2.1 Basics of Random Matrix Theory

2.1.1 Limiting distribution of eigenvalues

The previous chapter introduced some of the counter-intuitive phenomenons that occur in a high dimensional setting. In particular, we saw how, in the specific case of kernel-based algorithms, data pairs lose their individual discriminative power. At the opposite, RMT enables the analysis of random matrices as whole objects, and not only as collections of random variables. That is, instead of studying the individual behavior of random matrix entries, one can study a random matrix through its spectral properties. To understand this, we will consider the example of the sample covariance matrix.

Let n samples $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$ with a population covariance $\mathbf{C} \in \mathbb{R}^{p \times p}$. If one is interested in estimating the population covariance $\mathbf{C}$ from the available samples, the maximum likelihood estimator is the sample covariance matrix, defined by:

$$\hat{\mathbf{C}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\intercal = \frac{1}{n} \mathbf{X}\mathbf{X}^\intercal,$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$. Entry-wise, the sample covariance matrix is a consistent estimator of $\mathbf{C}$, in the sense that $\|\hat{\mathbf{C}} - \mathbf{C}\|_\infty \to 0$ as $n, p \to \infty$. However, when the operator norm is considered, $\hat{\mathbf{C}}$ turns out to be a very poor estimator, in the sense that $\|\hat{\mathbf{C}} - \mathbf{C}\| \not\to 0$ as $n, p \to \infty$. If matrix norms are equivalent for fixed values of n and p, this equivalence does not hold anymore for large values of n and p. This comes from the fact that the coefficients involved in the equivalence depend themselves on n or p. For instance, for symmetric matrices $\mathbf{A} \in \mathbb{R}^{p \times p}$, we have:

$$\|\mathbf{A}\|_\infty \leq \|\mathbf{A}\| \leq p\|\mathbf{A}\|_\infty$$

While the norm $\| \ \|_\infty$ mesures the point-wise convergence, the operator norm $\| \ \|$ mesures the spectral behavior of objects. And it is precisely by studying the spectral

distribution of $\hat{\mathbf{C}}$ that we understand why it poorly estimates $\mathbf{C}$ with respect to the operator norm. To that end, we define the following object.

Definition 1 (Empirical Spectral Distribution)
The Empirical Spectral Distribution (ESD) of a sample covariance matrix $\hat{\mathbf{C}} \in \mathbb{R}^{p\times p}$ is:

$$\mu_p = \frac{1}{p}\sum_{i=1}^{p} \delta_{\lambda_i(\hat{\mathbf{C}})}$$

where $\lambda_i(\hat{\mathbf{C}})$ is the i-th eigenvalue of $\hat{\mathbf{C}}$.

It turns out that, in the limit $n, p \to \infty$, the ESD of $\hat{\mathbf{C}}$ has an asymptotic behavior [10, 23]. One of the most fundamental result in RMT, namely Marčenko-Pastur theorem, shows the convergence in law of the ESD of $\hat{\mathbf{C}}$, for $\mathbf{C} = \mathbf{I}_p$.

Theorem 2 (Marčenko-Pastur)
Consider the sample covariance matrix $\hat{\mathbf{C}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\intercal$, where $\mathbf{X} \in \mathbb{R}^{p\times n}$ is a random matrix of i.i.d. entries with zero mean and unit variance.
As $n, p \to \infty$ with $\frac{p}{n} \to c \in]0, +\infty[$, the ESD of $\hat{\mathbf{C}}$ converges almost surely to a deterministic measure μ, with density given by:

$$\mu(dx) = \left(1 - \frac{1}{c}\right)^+ \delta_0(x) + \frac{1}{2\pi c x}\sqrt{(x-a)^+(b-x)^+}dx,$$

where $a = (1-\sqrt{c})^2$, $b = (1+\sqrt{c})^2$ and $(x)^+ = max(x, 0)$.

The main consequence of this theorem is that the eigenvalues of $\hat{\mathbf{C}}$ do not concentrate at 1, but are instead spread in the interval $[a, b]$. To understand better the implication of this statement, let us consider a setting where $p = 200$ and $n = 5000$. First of all, even if the values of n and p are not excessively large, Figure 2.1 shows that the empirical spectral distribution of $\hat{\mathbf{C}}$ matches adequately the prediction of Theorem 2, meaning that we are already in an asymptotic regime, where the assumption $n, p \to \infty$ is reasonable. Most importantly, in such a setting, one could guess that the number of samples n is much larger than p, as $n = 25p$, and therefore that the eigenvalues of $\hat{\mathbf{C}}$ approximately concentrate at 1. However, the support of the eigenvalues of $\hat{\mathbf{C}}$ is an interval of range $(1+\sqrt{c})^2 - (1-\sqrt{c})^2 = 4\sqrt{c}$. As $\sqrt{c} = \frac{1}{5}$, the eigenvalues of $\hat{\mathbf{C}}$ span on a range of 0.8 around 1, which is far from negligeable. Therefore, even in this apparently favorable setting, the usual asymptotic will fail to give an accurate estimation of the eigenvalues of $\mathbf{C}$. This simple example shows the interest of the theoretical assumption $n, p \to \infty$ with $\frac{p}{n} \to c \in]0, +\infty[$.

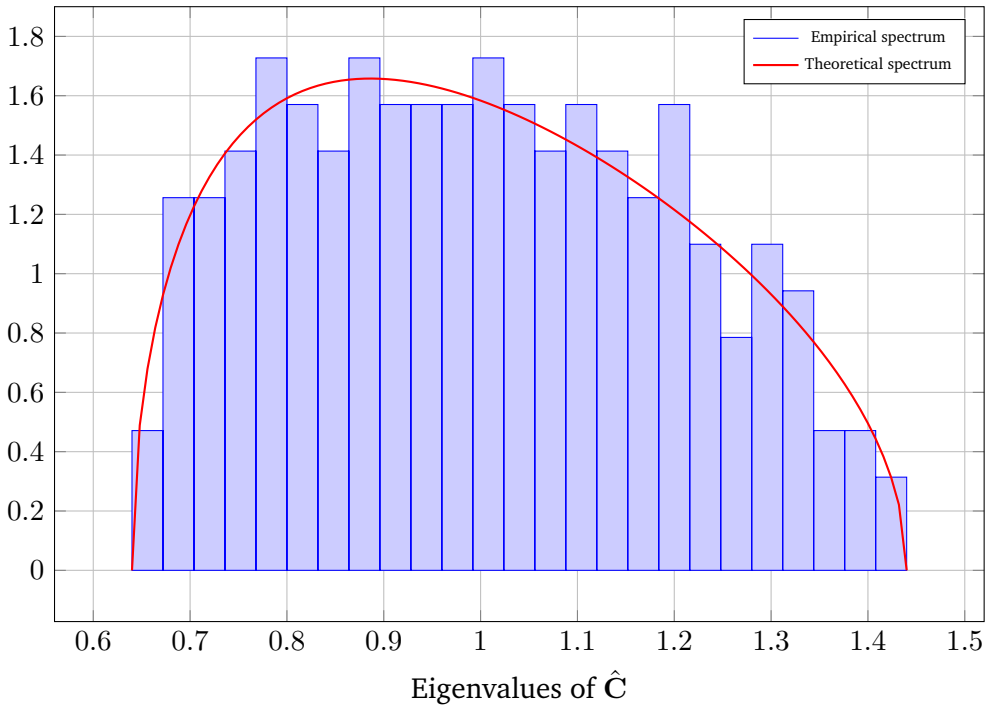

Fig. 2.1.: Histogram of the eigenvalues of $\hat{\mathbf{C}}$ compared to their theoretical distribution from Marčenko-Pastur theorem. $\mathbf{X} \in \mathbb{R}^{p \times n}$ has standard Gaussian entries, with $n = 5000$ and $p = 200$.

As the ESD is a non-continuous probability distribution, the historical proof of Theorem 2 makes use of the Stieltjes transform, which has some much more convenient properties.

Definition 3 (Stieltjes transform)
The Stieltjes transform of a probability measure μ is the function m_μ, defined for all $z \in \mathbb{C} \backslash supp(\mu)$ as:

$$m_\mu(z) = \int \frac{1}{t - z} \mu(dt),$$

where $supp(\mu)$ denotes the support of the measure μ.

One of the way to demonstrate Theorem 2 is to show the convergence of the Stieltjes transform of the ESD to a given limiting function, and then to deduce, from this limiting function, the limiting spectral distribution of the considered matrix. This is motivated by two facts:

- There exists an inverse formula to recover a probability distribution from its Stieltjes transform.
- The pointwise convergence of a sequence of Stieltjes transforms implies the convergence of their associated probability measures.

More precisely, we have the two following theorems.

Theorem 4 (Inverse Stieltjes transform)
*If a and b are continuity points of the probability measure μ (*i.e., $\mu(\{a\}) = \mu(\{b\}) = 0$*), then*

$$\mu([a,b]) = \frac{1}{\pi} \lim_{y \downarrow 0} \int_a^b \Im[m_\mu(x+iy)]dx,$$

and if μ has an isolated mass at x, then

$$\mu(\{x\}) = \frac{1}{\pi} \lim_{y \downarrow 0} -iym_\mu(x+iy).$$

The important consequence of this statement is that there is a one-to-one matching between a probability measure and its Stieltjes transform. Having access to the Stieltjes transform of a probability measure is enough to recover this probability measure.

Theorem 5
If $(\mu_n)_n$ is a sequence of probability measures with bounded support such that

$$\forall z \in \mathcal{D}, m_{\mu_n}(z) \to m_\mu(z),$$

where $\mathcal{D}$ is a subset of $\mathbb{C}^+$ that contains an accumulation point, then for any subset A of $\mathbb{R}$, we have

$$\mu_n(A) \to \mu(A).$$

As the Stieltjes transform is an analytic function, characterize it locally is sufficient to characterize it globally. Precisely, Vitali's convergence theorem [24] states that under assumptions of Theorem 5, the limit m_μ of the sequence m_{μ_n} is also an analytic function, and therefore a valid Stieltjes transform, for which we can apply the inversion formula. Thus, finding the limiting Stieltjes transform of the ESD is sufficient to retrieve the limiting spectral distribution.

The Stieltjes transform of the ESD is strongly linked with another ubiquitous object of RMT, namely the resolvent. The main interest of the resolvent, beyond the fact that it appears naturally in many fields of research, even far from RMT (for instance in the study of linear operators in Hilbert space [25] or in convex optimization [26]), is that its singular points provide some information on the spectrum of $\hat{\mathbf{C}}$.

Definition 6 (Resolvent)
The resolvent of the matrix $\hat{\mathbf{C}} \in \mathbb{R}^{p \times p}$ is defined, for $z \in \mathbb{C}$ not an eigenvalue of $\hat{\mathbf{C}}$, as:

$$\mathbf{Q}(z) = (\hat{\mathbf{C}} - z\mathbf{I}_p)^{-1}.$$

The link between the Stieltjes transform of the ESD and the resolvent lies in this simple equality:

$$m_{\mu_p}(z) = \frac{1}{p}\sum_{i=1}^{p}\int \frac{\delta_{\lambda_i(\hat{\mathbf{C}})}(t)}{t-z} = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{\lambda_i(\hat{\mathbf{C}})-z} = \frac{1}{p}\operatorname{Tr}\mathbf{Q}(z).$$

Therefore, combining this equality and the previous statements, we deduce that studying the spectral distribution of $\hat{\mathbf{C}}$ boils down to compute the limit of $\frac{1}{p}\operatorname{Tr}\mathbf{Q}(z)$.

2.1.2 Deterministic equivalents

The pioneers of RMT used the approach described until now, which consists in computing the limit of the Stieltjes transform m_{μ_p} to recover the limiting spectral measure of a matrix. However, modern RMT found other objects that allow to study more complex models of matrix. Those objects, called *deterministic equivalent*, are in particular widely used in the applications of RMT to ML.

The idea is to replace the resolvent $\mathbf{Q}$ by an object $\bar{\mathbf{Q}}$ for which scalar observations will be asymptotically identical to the scalar observations of the resolvent itself. As $\mathbf{Q}$ grows in the limit $p \to \infty$, it does not admit any kind of limit, but by knowing a deterministic equivalent, it is still possible to track the deterministic behavior of $\mathbf{Q}$. To this end, we will use the following definition of deterministic equivalents, picked out from [12].

Definition 7 (Deterministic equivalent)
We say that $\bar{\mathbf{Q}} \in \mathbb{R}^{p\times p}$ is a deterministic equivalent of the symmetric random matrix $\mathbf{Q} \in \mathbb{R}^{p\times p}$ if, for (sequences of) deterministic matrices $\mathbf{B} \in \mathbb{R}^{p\times p}$ and vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^p$ of unit norms (operator and Euclidean, respectively), we have, as $p \to \infty$:

- $\frac{1}{p}\operatorname{Tr}\mathbf{B}(\mathbf{Q}-\bar{\mathbf{Q}}) \xrightarrow{a.s.} 0$
- $\mathbf{a}^\intercal(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{b} \xrightarrow{a.s.} 0$

We will denote such equivalent matrices with the notation $\mathbf{Q} \leftrightarrow \bar{\mathbf{Q}}$.

In particular, if $\bar{\mathbf{Q}}$ is a deterministic equivalent of $\mathbf{Q}$, then $\frac{1}{p}\operatorname{Tr}\mathbf{Q}(z) - \frac{1}{p}\operatorname{Tr}\bar{\mathbf{Q}}(z) \xrightarrow{a.s.} 0$. This means that having a deterministic equivalent of the resolvent gives a natural asymptotic approximation of the limiting Stieltjes transform of the ESD. This notion of deterministic equivalent allows another formulation of the Marčenko-Pastur theorem stated above.

Theorem 8 (Marčenko-Pastur)
Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ with i.i.d. columns $\mathbf{x}_i$ such that $\mathbf{x}_i$ has independant entries with zero mean and unit variance, and denote $\mathbf{Q}(z) = (\frac{1}{n}\mathbf{X}\mathbf{X}^\intercal - z\mathbf{I}_p)^{-1}$ the resolvent of $\frac{1}{n}\mathbf{X}\mathbf{X}^\intercal$. Then, as $n, p \to \infty$ with $p/n \to c \in]0, \infty[$:

$$\mathbf{Q}(z) \leftrightarrow \bar{\mathbf{Q}}(z), \quad \bar{\mathbf{Q}}(z) = m(z)\mathbf{I}_p,$$

where $(z, m(z))$ is the unique solution in $\mathcal{Z}(\mathbb{C}\backslash[(1-\sqrt{c})^2, (1+\sqrt{c})^2])$ of

$$zcm^2(z) - (1-c-z)m(z) + 1 = 0,$$

where $\mathcal{Z}(\mathcal{A})$ denotes the set of "valid" Stieltjes transform pairs such that $z \in \mathcal{A}$.

The set $\mathcal{Z}(\mathcal{A})$, not detailed here, guarantees the unicity of the solution. Note that the function $m(z)$ is nothing more than the Stieltjes transform of the probability measure μ of Theorem 2, known as the Marčenko-Pastur distribution. This is the limiting spectral distribution of the matrix $\frac{1}{n}\mathbf{X}\mathbf{X}^\intercal$. As we are in the case $\mathbf{C} = \mathbf{I}_p$, the deterministic equivalent of $\mathbf{Q}$ is a mere identity matrix, up to a scalar multiplicative constant. In more complex covariance models, this will not be the case, as we will see for example in the case of multi-task models.

We will now give, for didactic reasons, a heuristic proof of a version of Theorem 8, in the general case $\mathbf{C} \neq \mathbf{I}_p$ and for $z = 1$. We refer the readers to [12] for a rigorous version of the proof. This heuristic demonstration is the opportunity to present the technical manipulations that will be used to prove the Theorem 21 of Chapter 4, yet in a simpler model. Following the idea of Theorem 8, the only aim here is to find a valid deterministic equivalent $\bar{\mathbf{Q}}$ of $\mathbf{Q}$ such that

$$\mathbf{Q} = \left(\mathbf{I}_p - \frac{1}{n}\mathbf{X}\mathbf{X}^\intercal\right)^{-1} = \left(\mathbf{I}_p - \frac{1}{n}\mathbf{C}^{\frac{1}{2}}\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\intercal\mathbf{C}^{\frac{1}{2}}\right)^{-1}, \tag{2.1}$$

where $\mathbf{X} = \mathbf{C}^{\frac{1}{2}}\tilde{\mathbf{X}}$ and $\tilde{\mathbf{X}} \in \mathbb{R}^{p\times n}$ with i.i.d. columns $\tilde{\mathbf{x}}_i$ such that $\tilde{\mathbf{x}}_i$ has independent entries with zero mean and unit variance. As such, $\mathbf{Q}$ has a form very similar to the resolvent matrix obtained at the end of Chapter 3.

We start by "guessing" that a deterministic equivalent must be of the form $\bar{\mathbf{Q}} = \mathbf{F}^{-1}$. Then:

$$\mathbf{Q} - \bar{\mathbf{Q}} = \mathbf{Q}\left(\mathbf{F} - \mathbf{I}_p + \frac{1}{n}\mathbf{X}\mathbf{X}^\intercal\right)\bar{\mathbf{Q}}, \tag{2.2}$$

using the fact that $\mathbf{A}^{-1} - \mathbf{B}^{-1} = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A})\mathbf{B}^{-1}$, for any invertible matrices $\mathbf{A}$ and $\mathbf{B}$. Then $\mathbf{Q} - \bar{\mathbf{Q}}$ rewrites:

$$\mathbf{Q} - \bar{\mathbf{Q}} = \mathbf{Q}\left(\mathbf{F} - \mathbf{I}_p + \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\intercal\right)\bar{\mathbf{Q}}, \tag{2.3}$$

For $\bar{\mathbf{Q}}$ to be a deterministic equivalent, we must have, for any deterministic matrix $\mathbf{B} \in \mathbb{R}^{p\times p}$, the following condition, which comes from Definition 7:

$$\begin{aligned} &\frac{1}{p}\operatorname{Tr}\mathbf{B}(\mathbf{Q} - \bar{\mathbf{Q}}) \xrightarrow{a.s.} 0 \\ \Leftrightarrow &\frac{1}{p}\operatorname{Tr}\mathbf{B}\mathbf{Q}(\mathbf{F} - \mathbf{I}_p)\bar{\mathbf{Q}} + \frac{1}{np}\sum_{i=1}^{n} \mathbf{x}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\mathbf{x}_i \xrightarrow{a.s.} 0 \end{aligned} \tag{2.4}$$

One of the difficulty that occurs when stuying quantities such as $\frac{1}{p}\mathbf{x}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\mathbf{x}_i$ is that there is a dependancy between $\mathbf{Q}$ and the data vector $\mathbf{x}_i$. In order to break this dependancy, one is interested in studying a *perturbated* version of the resolvent, for instance $\mathbf{Q}_{-i}$ defined as

$$\mathbf{Q}_{-i} = \left(\frac{1}{n}\mathbf{X}_{-i}\mathbf{X}_{-i}^\intercal - z\mathbf{I}_p\right)^{-1},$$

where the notation $\mathbf{X}_{-i}$ is standing for the matrix $\mathbf{X}$ with the i-th column removed. Such an object is a rank-one perturbation of the resolvent. To study small rank perturbations of matrices, there exists a uselful formula, which is called Woodbury matrix identity.

Lemma 9 (Woodbury)
For any invertible matrix $\mathbf{B} \in \mathbb{R}^{m\times m}$ and $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{m\times k}$ such that $\mathbf{B}+\mathbf{U}\mathbf{V}^\intercal$ is invertible, we have:

$$(\mathbf{B} + \mathbf{U}\mathbf{V}^\intercal)^{-1} = \mathbf{B}^{-1} - \mathbf{B}^{-1}\mathbf{U}\left(\mathbf{I}_k + \mathbf{V}^\intercal\mathbf{B}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^\intercal\mathbf{B}^{-1}$$

This formula is useful to deal with resolvent matrices, when k is much smaller than m, because it allows to reduce them to simpler and smaller matrices. In particular, when $k = 1$, the right term of the above formula does not involve any matrix inverse. In that case, the identity is known as the Sherman-Morrison formula.

Lemma 10 (Sherman-Morrison)
For any invertible matrix $\mathbf{B} \in \mathbb{R}^{m\times m}$ and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$, $\mathbf{B} + \mathbf{u}\mathbf{v}^\intercal$ is invertible if and only if $1 + \mathbf{v}^\intercal \mathbf{B}^{-1}\mathbf{u} \neq 0$, and we have the following formula:

$$(\mathbf{B} + \mathbf{u}\mathbf{v}^\intercal)^{-1} = \mathbf{B}^{-1} - \frac{\mathbf{B}^{-1}\mathbf{u}\mathbf{v}^\intercal\mathbf{B}^{-1}}{1 + \mathbf{v}^\intercal\mathbf{B}^{-1}\mathbf{u}}.$$

In particular,

$$(\mathbf{B} + \mathbf{u}\mathbf{v}^\intercal)^{-1}\mathbf{u} = \frac{\mathbf{B}^{-1}\mathbf{u}}{1 + \mathbf{v}^\intercal\mathbf{B}^{-1}\mathbf{u}}.$$

By applying Sherman-Morrison formula to our perturbated version of the resolvent, the quantity $\mathbf{Q}\mathbf{x}_i$ can then be decomposed as:

$$\mathbf{Q}\mathbf{x}_i = \frac{\mathbf{Q}_{-i}\mathbf{x}_i}{1 - \frac{1}{n}\mathbf{x}_i^\intercal\mathbf{Q}_{-i}\mathbf{x}_i} \tag{2.5}$$

Let us take a look at the quantity $\frac{1}{n}\mathbf{x}_i^\intercal\mathbf{Q}_{-i}\mathbf{x}_i$. It is a quantity of the form $\frac{1}{p}\mathbf{x}^\intercal\mathbf{B}\mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^p$ is a random vector with zero mean and unit variance entries. One can easily see that $\mathbb{E}\left[\frac{1}{p}\mathbf{x}^\intercal\mathbf{B}\mathbf{x}\right] = \frac{1}{p}\operatorname{Tr}\mathbf{B}$. But most importantly, it turns out that the quantity $\frac{1}{p}\mathbf{x}^\intercal\mathbf{B}\mathbf{x}$ is asymptotically deterministic, as $\frac{1}{p}\mathbf{x}^\intercal\mathbf{B}\mathbf{x} - \frac{1}{p}\operatorname{Tr}\mathbf{B}$ converges almost surely to 0 as $n \to \infty$. In order to prove that, we will make use of the following result, which comes from Markov inequality and Borel-Cantelli lemma.

Lemma 11
For any sequence X_n of random variables, if there exists $k \in \mathbb{N}^\star$ and $l > 1$ such that $\mathbb{E}|X_n|^k = O(n^{-l})$, then $X_n \xrightarrow{a.s.} 0$.

Proof: According to Markov inequality:

$$\forall t > 0, \mathbb{P}(|X_n| > t) \leq \frac{\mathbb{E}|X_n|^k}{t^k} = O(t^{-k}n^{-l})$$

Thus, $\forall t > 0$, the series $\sum_{n\geq 0}\mathbb{P}(|X_n| > t)$ converges. According to Borel-Cantelli lemma:

$$\forall t > 0, \mathbb{P}(\limsup(|X_n| > t)) = 0.$$

One of the definition of almost sure convergence of X_n is precisely :

$$\forall t > 0, \mathbb{P}(\limsup(|X_n - X| > t)) = 0,$$

where X is the limit of X_n. Therefore, X_n converges almost surely to 0.

Moreover, we have the following result from [23] (Lemma B.26).

Lemma 12

Let $\mathbf{x} \in \mathbb{R}^p$ have independent entries with zero mean, unit variance and with bounded eight-order moment, then for any matrix $\mathbf{B} \in \mathbb{R}^{p \times p}$:

$$\mathbb{E}\left[(\mathbf{x}^\intercal \mathbf{B}\mathbf{x} - \operatorname{Tr}\mathbf{B})^4\right] \leq C \operatorname{Tr}(\mathbf{B}\mathbf{B}^\intercal)^2$$

where $C > 0$ is a constant independent from $\mathbf{B}$.

A direct consequence is that for any matrix $\mathbf{B} \in \mathbb{R}^{p \times p}$ such that $\|\mathbf{B}\| \leq 1$:

$$\mathbb{E}\left[\left(\frac{1}{p}\mathbf{x}^\intercal \mathbf{B}\mathbf{x} - \frac{1}{p}\operatorname{Tr}\mathbf{B}\right)^4\right] \leq \frac{C}{p^2} \tag{2.6}$$

By applying lemma 11 with $k = 4$, $l = 2$ and $X_p = \frac{1}{p}\mathbf{x}^\intercal \mathbf{B}\mathbf{x} - \frac{1}{p}\operatorname{Tr}\mathbf{B}$, it comes immediately that, for any matrix $\mathbf{B} \in \mathbb{R}^{p \times p}$ such that $\|\mathbf{B}\| \leq O(1)$:

$$\frac{1}{p}\mathbf{x}^\intercal \mathbf{B}\mathbf{x} - \frac{1}{p}\operatorname{Tr}\mathbf{B} \xrightarrow{a.s.} 0. \tag{2.7}$$

As $\|\mathbf{Q}\|$ is precisely of order $O(1)$, we can apply this result to $\mathbf{B} = \mathbf{C}^{\frac{1}{2}}\mathbf{Q}_{-i}\mathbf{C}^{\frac{1}{2}}$ and $\mathbf{x} = \tilde{\mathbf{x}}_i$, and we have the almost sure convergence:

$$\begin{aligned} &\frac{1}{n}\tilde{\mathbf{x}}_i^\intercal \mathbf{C}^{\frac{1}{2}}\mathbf{Q}_{-i}\mathbf{C}^{\frac{1}{2}}\tilde{\mathbf{x}}_i - \frac{1}{n}\operatorname{Tr}\mathbf{C}^{\frac{1}{2}}\mathbf{Q}_{-i}\mathbf{C}^{\frac{1}{2}} \xrightarrow{a.s.} 0 \\ \Leftrightarrow &\frac{1}{n}\mathbf{x}_i^\intercal \mathbf{Q}_{-i}\mathbf{x}_i - \frac{1}{n}\operatorname{Tr}\mathbf{C}\mathbf{Q}_{-i} \xrightarrow{a.s.} 0. \end{aligned} \tag{2.8}$$

Moreover, it can be proven that $\frac{1}{n}\operatorname{Tr}\mathbf{C}\mathbf{Q}_{-i} \simeq \frac{1}{n}\operatorname{Tr}\mathbf{C}\mathbf{Q}$. Using the definition of a deterministic equivalent, we also have $\frac{1}{n}\operatorname{Tr}\mathbf{C}\mathbf{Q} - \frac{1}{n}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}} \xrightarrow{a.s.} 0$. This means that

$$\frac{1}{n}\mathbf{x}_i^\intercal \mathbf{Q}_{-i}\mathbf{x}_i - \frac{1}{n}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}} \xrightarrow{a.s.} 0. \tag{2.9}$$

Similarly, we obtain that

$$\frac{1}{p}\mathbf{x}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}_{-i}\mathbf{x}_i - \frac{1}{p}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}}\mathbf{B}\bar{\mathbf{Q}} \xrightarrow{a.s.} 0. \tag{2.10}$$

Let us define the following quantities:

$$\tilde{\delta} = \frac{1}{1-\delta} \quad \text{and} \quad \delta = \frac{1}{n}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}}. \tag{2.11}$$

Therefore, our quantity of interest rewrites as:

$$\frac{1}{p}\mathbf{x}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\mathbf{x}_i = \frac{1}{p}\frac{\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}}{1-\delta} = \frac{1}{p}\tilde{\delta}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q} \tag{2.12}$$

Going back to equation (2.4), we have the following condition that would ensure $\bar{\mathbf{Q}} \leftrightarrow \mathbf{Q}$:

$$\begin{aligned} &\frac{1}{p}\operatorname{Tr}\mathbf{B}\mathbf{Q}(\mathbf{F}-\mathbf{I}_p)\bar{\mathbf{Q}} + \frac{1}{np}\sum_{i=1}^{n}\mathbf{x}_i^\intercal\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\mathbf{x}_i \xrightarrow{a.s.} 0 \\ &\Leftrightarrow \frac{1}{p}\operatorname{Tr}\mathbf{B}\mathbf{Q}\left(\mathbf{F}-\mathbf{I}_p+\tilde{\delta}\mathbf{C}\right)\bar{\mathbf{Q}} \xrightarrow{a.s.} 0 \end{aligned} \tag{2.13}$$

Therefore, we must have:

$$\bar{\mathbf{Q}} = \mathbf{F}^{-1} = \left(\mathbf{I}_p - \tilde{\delta}\mathbf{C}\right)^{-1}, \tag{2.14}$$

where δ and $\tilde{\delta}$ are the solutions of the following system of equations :

$$\begin{cases} \delta = \frac{1}{n}\operatorname{Tr}\mathbf{C}\bar{\mathbf{Q}} = \frac{1}{n}\operatorname{Tr}\mathbf{C}\left(\mathbf{I}_p - \tilde{\delta}\mathbf{C}\right)^{-1}, \\ \tilde{\delta} = \frac{1}{1-\delta}. \end{cases}$$

It is worth to note that most of the results of this section have been obtained with light assumptions on the data distribution. Most of the time, data do not need to be gaussian, even if it is a regular assumption in RMT. In particular, the concentration result $\frac{1}{p}\mathbf{x}^\intercal\mathbf{B}\mathbf{x} - \frac{1}{p}\operatorname{Tr}\mathbf{B} \xrightarrow{a.s.} 0$ only requires a bounded eight-order moment of the entries.

In this manuscript, we will not use Gaussian data, but we will instead make use of the notion of *concentrated vectors* [27–29]. To understand this notion, we will considered once again a mere Gaussian mixture model, and see how such data distribution can be relaxed with concentrated vectors.

Assumption 13 (Gaussian distribution)
The columns of the data matrix $\mathbf{X}$ are independent Gaussian random variables. Specifically, the data samples $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ are i.i.d. observations such that $\mathbf{x}_i \in \mathcal{C}_j \Leftrightarrow \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ where $\mathcal{C}_j$ denotes the Class j.

Assumption 14 (Concentrated vectors)
The columns of the data matrix $\mathbf{X}$ *are independent concentrated vectors. Specifically, the data samples* $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ *are i.i.d. observations such that:*
There exists $C, c > 0$ *such that for any* 1*-Lipschitz function* $\Phi : \mathbb{R}^p \to \mathbb{R}$*, we have:*

$$\forall u > 0, \quad \mathbf{x}_i \in \mathcal{C}_j \Rightarrow \mathbb{P}(|\Phi(\mathbf{x}_i) - m_j(\Phi)| \geq u) \leq Ce^{-(\frac{u}{c})^2}$$

where $\mathcal{C}_j$ *denotes the Class* j*, and* $m_j(\Phi) = \mathbb{E}\left[\Phi(\mathbf{x})|\mathbf{x} \in \mathcal{C}_j\right]$.
Under these assumtions, we can define

$$\boldsymbol{\mu}_j = \mathbb{E}[\mathbf{x}|\mathbf{x} \in \mathcal{C}_j]$$
$$\boldsymbol{\Sigma}_j = \mathrm{Cov}[\mathbf{x}|\mathbf{x} \in \mathcal{C}_j].$$

This second assumption encompasses the first one, and applies to a broader class of distributions. In particular, images produced by Generative Adversarial Networks (GAN) are concentrated vectors [30]. As GAN are known to produce very convincing fake images and text, this assumption allows to consider more realistic datasets. We see that the important features of the data distribution are the means $\boldsymbol{\mu}_j$ and the covariances $\boldsymbol{\Sigma}_j$. That is, the data distribution can be summarized only through its first-order and second-order moments.

2.2 Tools from statistical physics

In order to derive some results about the Bayes risk described in Chapter 1, we need to introduce some tools from statistical physics. The results of this section, which mainly come from [18], will be useful in Chapter 6, when they are used as a basis for Lemma 33. We give here the key ideas to understand the reasoning behind these results.

The computation of optimal bounds implies the analysis of complex statistical models, necessarily hard to decipher altogether. However, in the high-dimensional regime, such models can be decoupled in independent components that are then studied separately. To understand this phenomenon, let us consider the following binary classification model. Let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a collection of n independent data vectors of dimension p. To each vector $\mathbf{x}_i$ of $\mathbf{X}$ is attached a label $y_i \in \{-1, +1\}$ that describes to which class $\mathbf{x}_i$ belongs. Suppose we want to estimate the signal $y_i \in \mathbb{R}$ from the

data $\mathbf{X} \in \mathbb{R}^{p \times n}$, which can be split in two parts : $\mathbf{X} = (\mathbf{x}_i, \mathbf{X}_{-i})$. The first part, $\mathbf{x}_i$, is the following observation of y_i:

$$\mathbf{x}_i = y_i \boldsymbol{\mu} + z_i,$$

where $\boldsymbol{\mu} \in \mathbb{R}^p$ is unknown, $z_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p)$ and y_i, $\boldsymbol{\mu}$, z_i are independent. The second part, $\mathbf{X}_{-i}$ (*i.e.,* the matrix $\mathbf{X}$ with the i-th column removed), is independent of y_i. Before going further, we need to introduce the Replica Symmetric (RS) property.

Definition 15
The inference of $\mathbf{u} \in \mathbb{R}^p$ (or $\mathbf{u} \in \mathbb{R}^n$) from the data $\mathbf{X}$ satisfies the RS property with overlap q if, in the limit $p \to \infty$,

$$\langle \mathbf{u}, \mathbf{u}^1 \rangle, \ \langle \mathbf{u}^1, \mathbf{u}^2 \rangle, \ \langle \mathbf{u}, \hat{\mathbf{u}} \rangle, \ \langle \hat{\mathbf{u}}, \hat{\mathbf{u}} \rangle,$$

all converges to the same limit q, where $\mathbf{u}^1$ and $\mathbf{u}^2$ are independently sampled from the posterior law of $\mathbf{u}$ given $\mathbf{X}$, and $\hat{\mathbf{u}} = \mathbb{E}[\mathbf{u}|\mathbf{X}]$ is the MMSE estimator of $\mathbf{u}$ given $\mathbf{X}$.

This is a common assumption, which holds for many inference problems. We assume that this assumption holds in our case, and that $\boldsymbol{\mu}|\mathbf{X}$ and $\frac{1}{\sqrt{n}}\mathbf{y}|\mathbf{X}$ satisfies the RS property, with overlap q_u and q_v respectively, where $\mathbf{y} = (y_i)_i$ is the vector of labels. These overlaps are fundamental to compute the asymptotic Bayes risk of our model, which will turn out to be a function of q_u, as we will see later. With this assumption, we can state the following proposition.

Proposition 16
Suppose that the law $\boldsymbol{\mu}|\mathbf{X}$ satisfies the RS property with overlap q. Then, as $p \to \infty$, the posterior law of y_i given $\mathbf{X}$ is asymptotically equivalent to the law $\bar{P}$ defined as:

$$\frac{d\bar{P}(y|\mathbf{X})}{dP_{y_i}(y)} \ \alpha \ exp\left(y\langle \mathbf{x}_i, \hat{\boldsymbol{\mu}} \rangle - \frac{1}{2} q y^2\right),$$

where P_{y_i} is the prior of y_i and $\hat{\boldsymbol{\mu}} = \mathbb{E}[\boldsymbol{\mu}|\mathbf{X}]$ is the MMSE estimator of $\boldsymbol{\mu}$. As a consequence, the statistics $s_i = \langle \mathbf{x}_i, \hat{\boldsymbol{\mu}} \rangle$ is asymptotically sufficient for estimating y_i from $\mathbf{X}$.

For a proof of this proposition, we refer the interested readers to the section 5 of [18]. This proposition means that, to estimate y_i, all the relevant information from the data $\mathbf{X}$ is contained in the statistics $s_i = \langle \mathbf{x}_i, \hat{\boldsymbol{\mu}} \rangle$. Thus, in the high-dimensional regime, the inference of y_i from the data $\mathbf{X}$ is equivalent to two decoupled inferences:

- The inference of $\boldsymbol{\mu}$ from $\mathbf{X}$, through the estimator $\hat{\boldsymbol{\mu}}$.

- The inference of y_i from $\mathbf{x}_i$ and $\hat{\boldsymbol{\mu}}$.

Therefore, each one of these inference problems can be studied separately. Analyzing independently each component of the problem turns out to be much easier. Indeed, as we will see in the following proposition, the second problem is equivalent to an inference from a mere Gaussian channel, meaning that it is possible to perform some computations on them.

Proposition 17
With the same assumptions than Proposition 16, as $p \to \infty$, $\frac{s_i}{\sqrt{q}}$ converges in law to $\sqrt{q}y_i + \xi_i$, where $\xi_i \sim \mathcal{N}(0,1)$ is independent of y_i. As a result, estimating y_i from $\mathbf{X}$ is equivalent to estimating y_i from the output of a Gaussian channel with Signal-to-Noise Ratio (SNR) q.

Proof:

$$s_i = \langle \mathbf{x}_i, \hat{\boldsymbol{\mu}} \rangle = \langle y_i \boldsymbol{\mu} + z_i, \hat{\boldsymbol{\mu}} \rangle = y_i \langle \boldsymbol{\mu}, \hat{\boldsymbol{\mu}} \rangle + \langle z_i, \hat{\boldsymbol{\mu}} \rangle$$

$\langle \boldsymbol{\mu}, \hat{\boldsymbol{\mu}} \rangle$ is asymptotically equal to q, and $\langle z_i, \hat{\boldsymbol{\mu}} \rangle$ follows the distribution

$$\langle z_i, \hat{\boldsymbol{\mu}} \rangle \sim \mathcal{N}(0, \|\hat{\boldsymbol{\mu}}\|^2) = \mathcal{N}(0, q).$$

As y_i and z_i are independent, it follows that

$$\frac{s_i}{\sqrt{q}} \xrightarrow{\mathcal{L}} \sqrt{q}y_i + \xi_i,$$

where ξ_i is a standard normal random variable independent of y_i. As s_i is the sufficient statistics to estimate y_i from $\mathbf{X}$, the inference of y_i from $\mathbf{X}$ is equivalent to the inference of y_i from the Gaussian channel $u_i = \sqrt{q}y_i + \xi_i$, which concludes the proof.

We know that the statistics $s_i = \langle \mathbf{x}_i, \hat{\boldsymbol{\mu}} \rangle$ is sufficient to estimate y_i from $\mathbf{X}$. Therefore, the estimator that minimizes the Bayes risk is sign(s_i) :

- If s_i is negative, then $\mathbf{x}_i$ is closer to $-\boldsymbol{\mu}$, and $\mathbf{x}_i$ is consequently classified in the class with label -1.

- If s_i is positive, then $\mathbf{x}_i$ is closer to $\boldsymbol{\mu}$, and $\mathbf{x}_i$ is consequently classified in the class with label $+1$.

As $\frac{s_i}{\sqrt{q_u}}$ follows asymptotically the law $\mathcal{N}(\sqrt{q_u}y_i, 1)$, we deduce that the Bayes risk is $\mathcal{Q}(\sqrt{q_u})$, where $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} \, \mathrm{d}u$.

Moreover, thanks to these results, if the overlap of $\boldsymbol{\mu}|\mathbf{X}$ is known, we can express the law of $y_i|\mathbf{X}$, and therefore the overlap of $\frac{1}{\sqrt{n}}\mathbf{y}|\mathbf{X}$. With similar propositions, if the overlap of $\frac{1}{\sqrt{n}}\mathbf{y}|\mathbf{X}$ is known, one can retrieve the overlap of $\boldsymbol{\mu}|\mathbf{X}$. As a result, the overlaps of $\boldsymbol{\mu}|\mathbf{X}$ and $\frac{1}{\sqrt{n}}\mathbf{y}|\mathbf{X}$ can be expressed as functions of each other. This leads to a system of equations that gives the couple (q_u, q_v), from which we can compute the asymptotic Bayes risk.

3 Graph-based methods for multi-task semi-supervised learning

3.1 Multi-task and semi-supervised learning

3.1.1 Multi-task learning

The main difference between ML algorithms and mere optimization problems is that the purpose of a ML algorithm is to be applied on new data. Thus, ML algorithms should be able to generalize well on previously unknown data. An algorithm trained for a specific task on a narrow dataset will have a hard time to be predictive on data which differs even slightly from the training data.

A way to tackle this problem is to learn simultaneously from different similar tasks, therefore enriching the learning dataset. This approach is called Multi-Task Learning (MTL). Related tasks benefit from each other, globally enriching the data representation. For some applications, the benefits of MTL are clear, for example clinical data from different hospitals, or MRI data from different MRI scanners [31, 32]. But more broadly, the power of generalization of the algorithm is strengthen thanks to an inductive bias provided by the auxiliary tasks: the algorithm will prefer an assumption that explains the data from all tasks rather than an assumption that only explains the one task it intends to solve [33]. Even unrelated tasks can be beneficial, as long as the algorithm knows that the tasks are unrelated [34]. However, if the relation between tasks is not properly provided to the algorithm, it can lead to severe losses of performances. This effect is known as *negative transfer*, and is one of the main limitations of MTL algorithms.

Some MTL algorithms, like Least-Square Support Vector Machine have been analyzed with RMT tools, in a fully supervised setting [35, 36]. In particular, [35] introduces (for the first time, as far as we know) the counter-intuitive idea of choosing the values of labels differently from the standard ± 1. This idea, which appears to be an

easy and intuitive way to tackle the issue of negative transfer, is one of the keys to our present algorithm.

3.1.2 Semi-supervised learning

ML algorithms are often categorized as *supervised* or *unsupervised*, depending whether the input data they are fed with is *labeled* or *unlabeled*. In the supervised case, to each data point $\mathbf{x}_i$ is attached a label y_i, which is a known output value for the considered learning model. Supervised algorithms are often able to reach high accuracy, but are strongly reliant on the labeling process, which is by far the most costly process when it comes to build a database. It is indeed a tedious task, often done by a human being. At the opposite, for unsupervised learning, there are no labels attached to the data points. Thus, the goal of unsupervised algorithms is to find the underlying structure of the data, by identifying some classes of data points that share some common features.

Semi-Supervised Learning (SSL) is at the crossroads between supervised and unsupervised learning, making use of both labeled and unlabeled data. SSL can be seen as an extension of the conventional supervised learning paradigm by augmenting the (labeled) training data set with unlabeled data, which then "unsupervisably" serve to boost learning performance. It is expected to take the best of both worlds, combining the cheap cost of unsupervised learning with the high accuracy of supervised learning. In particular, SSL has long been considered to be a powerful tool to make use of large amounts of unlabeled data [37].

However, some work also point out the lack of theoretical understanding of these methods [38–40]. Indeed, many algorithms appear to be unable to learn effectively from unlabeled data, even though it is a fundamental aspect of semi-supervised learning [37]. Even by considering a mere Gaussian mixtures model (which is one of the simplest possible parametric model one could consider for a classification problem), well-known methods such as Laplacian regularization appears to be uneffective to learn from unlabeled data [41].

Fortunately, advances in Random Matrix Theory (RMT) have been exploited to design better methods, by proposing fundamental corrections of known algorithms. Specifically, [42] provides a theoretical analysis (in a single-task framework) of the graph-based semi-supervised scheme we use in this thesis [43, 44]. In particular, it reveals a fundamental flaw of graph-based algorithms, and most importantly, it proposes a simple way to tackle it, through a mere data centering. In the same

manner as the choice of label values from [35] mentionned above, this simple improvement is a strong inspiration for this thesis. Indeed, it shows the lack of theoretical understanding of ML algorithms, and reveals how a better intuition of a technical tool's behavior can lead to a major improvement of this tool.

3.1.3 Graph-based methods

Graph-based approaches, such as the Laplacian regularization method, are widely used to perform semi-supervised learning [43, 45, 46]. The idea consists in propagating the effective information of labeled data to unlabeled data, following the natural assumption that similar data points should have similar labels. This similarity between data points $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ is measured by the quantity $\omega_{ii'} = h\left(\frac{1}{p}\langle \mathbf{x}_i, \mathbf{x}_{i'}\rangle\right)$ or $\omega_{ii'} = h\left(\frac{1}{p}\|\mathbf{x}_i - \mathbf{x}_{i'}\|^2\right)$ with h an increasing function, so that similar data vectors $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ are connected with a large weight. From there, graph-based learning algorithms estimate the class of each node $\mathbf{x}_i$ through a class attachment "score" f_i, by solving an optimization problem of the form:

$$\min_{\mathbf{f}} \sum_{i,i'=1}^{n} \omega_{ii'} \left(d_i^{\gamma} f_i - d_{i'}^{\gamma} f_{i'}\right)^2 + \alpha \sum_{i=1}^{n} d_i^{2\gamma-1}(f_i - y_i), \tag{3.1}$$

where $d_i = \sum_{i'=1}^{n} \omega_{ii'}$ is the degree of the node associated to $\mathbf{x}_i$, *i.e.,* the sum of the edges $\omega_{ii'}$ connected to this node. When $\gamma = 0$, this formulation is called Standard Laplacian, and when $\gamma = \frac{1}{2}$, it is called Normalized Laplacian. The second term of (3.1) is a regularization term, and the strength of this regularization is controlled by the parameter $\alpha > 0$. This term enforces smoothly that a label y_i and its associated score f_i should be close. It is sometimes replaced by a strict fitting constraint $f_i = y_i$. For now, we let aside this regularization term and we use the strict constraint. In the Standard Laplacian-based approach, this gives the following optimization problem:

$$\min_{\mathbf{f}} \sum_{i,i'=1}^{n} \omega_{ii'} \left(f_i - f_{i'}\right)^2$$
$$\textit{such that } f_i = y_i \;\forall\; 1 \leq i \leq n_\ell.$$

We will consider both regularized and unregularized cases later, but let us first introduce our multi-task model.

3.2 Model and assumptions

Let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a collection of n independent data vectors of dimension p. The data are divided into T subsets, each one attached to an individual "task". Specifically, letting $\mathbf{X} = [\mathbf{X}^1, \dots, \mathbf{X}^T] \in \mathbb{R}^{p \times n}$, Task t is a semi-supervised binary classification task with training samples $\mathbf{X}^t = [\mathbf{X}_\ell^t, \mathbf{X}_u^t] \in \mathbb{R}^{p \times n^t}$ which consists of a set of n_ℓ^t labeled data samples $\mathbf{X}_\ell^t = \{\mathbf{x}_i^t\}_{i=1}^{n_\ell^t}$ and a set of n_u^t unlabeled data points $\mathbf{X}_u^t = \{\mathbf{x}_i^t\}_{i=n_\ell^t+1}^{n^t}$. To each labeled data point $\mathbf{x}_i^t$ from Task t is attached a label y_i^t and the goal is to predict the labels of unlabeled data $\mathbf{X}_u^t$. We focus on a two-class setting (the multi-class setting is briefly discussed in Section 4.3.1), meaning that the labels are scalar values. In this thesis, we will divide the T tasks in:

- One "target task" which is the one task we actually want to perform.
- $T-1$ "source tasks" which will help us to perform the "target task".

In a sense, our setting is close to transfer-learning because we aim to perform a single classification task. However, as the calculus done in the following is true for any choice of target task, we can easily perform sequentially each one of the T tasks with the help of the other, and therefore perform multi-task learning.

Assumption 18 (On the data distribution)
The columns of the data matrix $\mathbf{X}$ are independent Gaussian random variables. Specifically, the data samples $(\mathbf{x}_1^t, \dots, \mathbf{x}_{n^t}^t)$ from Task t are i.i.d. observations such that $\mathbf{x}_i^t \in \mathcal{C}_j^t \Leftrightarrow \mathbf{x}_i^t \sim \mathcal{N}(\boldsymbol{\mu}_j^t, \mathbf{I}_p)$ where $\mathcal{C}_j^t$ denotes the Class j of Task t.

Even though this assumption is intuitive and straightforward, one could argue that it is unrealistic. Indeed, natural data is not Gaussian, and most importantly, requiring independent entries for each observation is very constraining. Concentrated vectors, described for the first time in [27] and integrated in RMT by [28, 29], allow us to relax the previous assumption, while preserving its core idea.

Assumption 19 (On the data distribution)
The columns of the data matrix $\mathbf{X}$ are independent concentrated vectors with isotropic covariance. Specifically, the data samples $(\mathbf{x}_1^t, \dots, \mathbf{x}_{n^t}^t)$ from Task t are i.i.d. observations such that:
There exists $C, c > 0$ such that for any 1-Lipschitz function $\Phi : \mathbb{R}^p \to \mathbb{R}$, we have:

$$\forall u > 0, \quad \mathbf{x}_i^t \in \mathcal{C}_j^t \Rightarrow \mathbb{P}(|\Phi(\mathbf{x}_i^t) - m_j^t(\Phi)| \geq u) \leq Ce^{-(\frac{u}{c})^2},$$

where $\mathcal{C}_j^t$ denotes the Class j of Task t, and $m_j^t(\Phi) = \mathbb{E}\left[\Phi(\mathbf{x})|\mathbf{x} \in \mathcal{C}_j^t\right]$.
Under these assumptions, we can define

$$\boldsymbol{\mu}_j^t = \mathbb{E}[\mathbf{x}|\mathbf{x} \in \mathcal{C}_j^t],$$
$$\boldsymbol{\Sigma}_j^t = \mathrm{Cov}[\mathbf{x}|\mathbf{x} \in \mathcal{C}_j^t],$$

and we further impose that $\forall j, t, \boldsymbol{\Sigma}_j^t = \mathbf{I}_p$.

As explained in Chapter 2, this second assumption encompasses the previous one, and allows to consider more realistic datasets. All the theoretical results discussed in the remainder of the manuscript are obtained using Assumption 19. The assumption $\boldsymbol{\Sigma}_j^t = \mathbf{I}_p$ is still restrictive, but is convenient to perform our statistical analysis, which is a key to understand the fondamental behavior of our algorithm and propose crucial adjustments to improve it. More importantly, experiments of Section 4.4.3 show that the algorithm performs well on concentrated vectors for which the covariance is not necessarily isotropic. This suggests that the results obtained here could generalize to a larger class of data distribution.

As discussed earlier, we consider a large dimensional setting, where the dimension of the data and the number of data have the same order of magnitude. Moreover, to take into account the potential consequences of class unbalances, our statistical analysis will make use of the proportion of data in each class. To this end, we make the following assumption.

Assumption 20 (Growth Rate)
As $n \to \infty$,

- $p/n \to c > 0$.
- $n_j^t/n \to \rho_j^t > 0$ *(n_j^t the number of data in $\mathcal{C}_j^t$).*
- $n^t/n \to \rho^t > 0$.
- $n_{\ell j}^t/n_j^t \to \eta_j^t > 0$ *($n_{\ell j}^t$ the number of labeled data in $\mathcal{C}_j^t$).*
- $n_\ell^t/n^t \to \eta^t > 0$.

Furthermore, we also consider the $2T$-dimensional vector $\boldsymbol{\rho} = \left(\rho_1^1, \rho_2^1, \rho_1^2, \ldots, \rho_2^T\right)^\intercal$, the T-dimensional vector $\bar{\boldsymbol{\rho}} = \left(\rho^1, \ldots, \rho^T\right)^\intercal$, and similarly the vectors $\boldsymbol{\eta}$ and $\bar{\boldsymbol{\eta}}$.

3.3 Problem formulation

3.3.1 Optimization framework

We recall the previously mentionned optimization problem in the single-task setting, from [42]:

$$\min_{\mathbf{f}} \sum_{i,i'=1}^{n} \omega_{ii'} \left(f_i - f_{i'}\right)^2 \tag{3.2}$$
$$\textit{such that } f_i = y_i \; \forall \; 1 \leq i \leq n_\ell.$$

Our multi-task semi-supervised algorithm follows the same idea, yet with the addition of a hyperparameter matrix $\mathbf{\Lambda} = \{\Lambda^{tt'}\}_{t,t'=1}^{T}$ which filters how much each task should be related to each other. The optimization becomes

$$\min_{\mathbf{f}^1,\ldots,\mathbf{f}^T} \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} \omega_{ii'}^{tt'} \left(f_i^t - f_{i'}^{t'}\right)^2 \tag{3.3}$$
$$\textit{such that } f_i^t = y_i^t \; \forall \; 1 \leq i \leq n_\ell^t \textit{ and } 1 \leq t \leq T,$$

and the weights $\omega_{ii'}^{tt'}$ are now

$$\omega_{ii'}^{tt'} = \frac{1}{Tp} \langle \mathbf{x}_i^t, \mathbf{x}_{i'}^{t'} \rangle.$$

The term $f_i^t = y_i^t$ is the fitting constraint, which imposes that the score of labeled data should match the initial label assignment. This constraint is sometimes relaxed by adding a term $2\alpha_\ell \|\mathbf{f}_\ell - \mathbf{y}_\ell\|^2$ to the minimization problem, with $\alpha_\ell > 0$, where $\mathbf{f}_\ell$ and $\mathbf{y}_\ell$ respectively denotes the vectors $\mathbf{f}$ and $\mathbf{y}$ restricted to labeled data.

The classical Laplacian regularization algorithm associated to (3.2) has been studied in depth in [41] in the single-task setting. There, the authors showed the fundamental importance to "center" the weight matrix $\mathbf{W} = \{\omega_{ii'}\}_{i,i'=1}^{n}$. This centering approach corrects an important bias in the regularized Laplacian which completely annihilates the use of unlabeled data in a large dimensional setting. A significant performance increase was reported, both in theory and in practice in [42] when this basic, yet counter-intuitive, correction is accounted for. The same phenomenon evidently arises in the multi-task extension and we propose consequently the same

task-wise pre-centering in the context of multi-task learning, that is we update the weight matrices $(\mathbf{W}^{tt'})_{t,t'}$ as:

$$\hat{\mathbf{W}}^{tt'} = \mathbf{P}^t \mathbf{W}^{tt'} \mathbf{P}^{t'}, \tag{3.4}$$

with $\mathbf{P}^t = \left(\mathbf{I}_{n^t} - \frac{1}{n^t}\mathbb{1}_{n^t}\mathbb{1}_{n^t}^\intercal\right)$ the centering projector. This is equivalent to replace the data matrix $\mathbf{X}^t$ by its centered version $\mathring{\mathbf{X}}^t = \mathbf{X}^t\mathbf{P}^t$. It is worth to note that the centering is performed undifferently on labeled and unlabeled data, as the whole task is concerned and both classes are centered at once. For a matter of readability, $\mathbf{X}^t$ will denote the centered matrix in the remainder of the manuscript. Similarly, $\boldsymbol{\mu}_j^t$ will denote the centered mean of $\mathbf{x} \in \mathcal{C}_j^t$.

However, the optimization problem described in (3.3) is non convex since the entries of the weight matrix $\hat{\mathbf{W}}$ may take negative values (this must actually be the case as the mean value of the entries of $\hat{\mathbf{W}}$ is zero). To deal with this problem, we further propose (as in [42]) to constrain the norm of the unlabeled data score vector $\mathbf{f}_u$ (that is, the score vector $\mathbf{f}$ restricted to unlabeled data) by appending a regularization term $2\alpha_u\|\mathbf{f}_u\|^2$ to the previous minimization problem, with $\alpha_u > 0$. The equation (3.3) becomes:

$$\begin{aligned} &\min_{\mathbf{f}^1,\dots,\mathbf{f}^T} \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t}\sum_{i'=1}^{n^{t'}} \hat{\omega}_{ii'}^{tt'} \left(f_i^t - f_{i'}^{t'}\right)^2 + 2\alpha_u \sum_{t=1}^{T}\sum_{i=n_\ell^t+1}^{n^t} (f_i^t)^2 \\ &\textit{such that } f_i^t = y_i^t \;\forall\; 1 \le i \le n_\ell^t \textit{ and } 1 \le t \le T. \end{aligned} \tag{3.5}$$

This leads, under a more convenient matrix formulation (see Section A.1 of the appendix for the details), to

$$\begin{aligned} &\min_{\mathbf{f}\in\mathbb{R}^n} -\mathbf{f}^\intercal \tilde{\mathbf{W}} \mathbf{f} + \alpha_u \|\mathbf{f}_u\|^2 \\ &\textit{such that } \mathbf{f}_\ell = \mathbf{y}_\ell, \end{aligned} \tag{3.6}$$

with $\tilde{\mathbf{W}}$ the block matrix for which each block is $\tilde{\mathbf{W}}^{tt'} = \Lambda^{tt'}\hat{\mathbf{W}}^{tt'}, \forall (t,t') \in \{1,\dots,T\}^2$.

With the relaxed fitting constraint, equation (3.5) becomes:

$$\min_{\mathbf{f}^1,\dots,\mathbf{f}^T} \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} \hat{\omega}_{ii'}^{tt'} \left(f_i^t - f_{i'}^{t'}\right)^2 \tag{3.7}$$
$$+ 2\alpha_\ell \sum_{t=1}^{T} \sum_{i=1}^{n_\ell^t} (f_i^t - y_i^t)^2 + 2\alpha_u \sum_{t=1}^{T} \sum_{i=n_\ell^t+1}^{n^t} (f_i^t)^2.$$

In practice, as it is explained in Section A.1.2 of the appendix, α_ℓ and α_u should not take different values, and we can choose the same value $\alpha_\ell = \alpha_u = \alpha > 0$ for both. From now on, we will use undifferently α for both α_ℓ and α_u. Thus, it is possible to express it in a more natural way, similar to Laplacian-based formulations as (3.1), by putting the regularization term and the relaxed constraint over the labels altogether in the same term:

$$\min_{\mathbf{f}^1,\dots,\mathbf{f}^T} \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} \hat{\omega}_{ii'}^{tt'} \left(f_i^t - f_{i'}^{t'}\right)^2 + 2\alpha \sum_{t=1}^{T} \sum_{i=1}^{n^t} (f_i^t - y_i^t)^2, \tag{3.8}$$

where the convention $y_i^t = 0$ has been used for unlabeled data.

Now let us consider separately the problem with the strict fitting constraint $\mathbf{f}_\ell = \mathbf{y}_\ell$ and the problem with the relaxed fitting constraint.

3.3.2 Strict fitting constraint

With the strict constraint $\mathbf{f}_\ell = \mathbf{y}_\ell$, the problem is convex for all $\alpha > \|\tilde{\mathbf{W}}_{uu}\|$. In addition, $\mathbf{f}_u$ is the solution to a quadratic optimization problem with linear equality constraints, and can be obtained explicitly (see detail in Section A.1.1 of the appendix):

$$\mathbf{f}_u = \left(\mathbf{I}_{n_u} - \frac{\mathbf{Z}_u^\intercal \mathbf{A} \mathbf{Z}_u}{Tp}\right)^{-1} \frac{\mathbf{Z}_u^\intercal \mathbf{A} \mathbf{Z}_\ell}{Tp} \mathbf{y}_\ell, \tag{3.9}$$

where

$$\mathbf{A} = \tilde{\mathbf{\Lambda}} \otimes \mathbf{I}_p, \text{ with } \tilde{\mathbf{\Lambda}} = \frac{\mathbf{\Lambda}}{\alpha} \text{ the hyperparameter matrix},$$
$$\mathbf{Z}_\ell = \sum_{t=1}^{T} \mathbf{E}_{tt}^{[T]} \otimes \mathbf{X}_\ell^t \text{ and } \mathbf{Z}_u = \sum_{t=1}^{T} \mathbf{E}_{tt}^{[T]} \otimes \mathbf{X}_u^t \text{ the data matrices}.$$

If we further use Woodbury matrix identity, we have

$$\mathbf{f}_u = \frac{1}{Tp}\mathbf{Z}_u^\intercal \mathbf{A}^{\frac{1}{2}} \underbrace{\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}}_{=\mathbf{Q}_u} \mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell \mathbf{y}_\ell. \tag{3.10}$$

To understand the role of α, let us consider the two extremal values of the interval α can belong to, that is $]\|\tilde{\mathbf{W}}_{uu}\|, +\infty[$.

- If $\alpha \to +\infty$, $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp} \to \mathbf{O}_{Tp}$, meaning that $\mathbf{Q}_u \to \mathbf{I}_{Tp}$ and the scores become approximately $\mathbf{f}_u \simeq \frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_\ell}{Tp}\mathbf{y}_\ell$. Individually, the score of an unlabeled sample is

  $$f_i \simeq \frac{\mathbf{x}_i^\intercal\mathbf{A}\mathbf{Z}_\ell}{Tp}\mathbf{y}_\ell.$$

  For each unlabeled sample $\mathbf{x}_i$, its score only depends on itself and labeled data, so we are in a purely supervised setting.

- If $\alpha \to \|\tilde{\mathbf{W}}_{uu}\|$, then $\|\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\| = \|\frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_u}{Tp}\| \to 1$, because $\frac{\tilde{\mathbf{W}}_{uu}}{\alpha} = \frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_u}{Tp}$. Let us say that the largest eigenvalue of $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}$ is $1-\varepsilon$, with $\varepsilon \to 0$, and its associated eigenvector is $\mathbf{u}$. Then,

  $$\mathbf{Q}_u \simeq \frac{1}{\varepsilon}\mathbf{u}\mathbf{u}^\intercal.$$

  Indeed, if we consider the diagonalisation in an orthonormal basis of the symmetric matrix $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp} = \mathbf{U}\mathbf{D}\mathbf{U}^\intercal \in \mathbb{R}^{Tp\times Tp}$, we have:

  $$\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1} = (\mathbf{U}\mathbf{U}^\intercal - \mathbf{U}\mathbf{D}\mathbf{U}^\intercal)^{-1}$$
  $$= \mathbf{U}\left(\mathbf{I}_{Tp} - \mathbf{D}\right)^{-1}\mathbf{U}^\intercal = \mathbf{U}\begin{pmatrix} \frac{1}{\varepsilon} & & & \\ & \frac{1}{\lambda_2} & & \\ & & \ddots & \\ & & & \frac{1}{\lambda_{n_u}} \end{pmatrix}\mathbf{U}^\intercal,$$

  where $\lambda_i > \varepsilon, \forall i \geq 2$. In the limit $\varepsilon \to 0$, the first eigenvector is emphasized infinitely more than any other, resulting in the previous approximation. The consequence on the solution (3.10) is that

  $$\mathbf{f}_u \simeq \frac{1}{Tp\varepsilon}\underbrace{\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{u}}_{=\mathbf{v}_u^\intercal}\underbrace{\mathbf{u}^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell}_{=\mathbf{v}_\ell}\mathbf{y}_\ell.$$

Everything happens as if all the labeled data were projected along a single axis, which is the first eigenvector of $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}$. Once this projection is done, the expression of the solution is similar to the previous one, $\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u$ being replaced by $\mathbf{v}_u$ and $\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell$ being replaced by $\mathbf{v}_\ell$. It can be easily related to the well-known unsupervised classification called Principal Component Analysis (PCA), for which the first component is also the first eigenvector of the covariance matrix [47–49].

We understand that α controls the balance between supervised and unsupervised learning. The stronger the regularization is, the closer we are to a supervised algorithm. Using the previous remarks, equation (3.9) can be decomposed in two terms, associated respectively to the supervised and the unsupervised component of the final score:

$$\mathbf{f}_u = \underbrace{\left(\mathbf{I}_{n_u} - \frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_u}{Tp}\right)^{-1}}_{\text{unsupervised}} \underbrace{\frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_\ell}{Tp}\mathbf{y}_\ell}_{\text{supervised}}.$$

The right term is interpretable as a projection of labels of labeled data over unlabeled data. The left term is interpretable as a filter of the right term over the principal components of $\frac{\mathbf{Z}_u^\intercal\mathbf{A}\mathbf{Z}_u}{Tp}$, *i.e.,* the directions which are the most relevant to discriminate unlabeled data.

3.3.3 Relaxed fitting constraint

With the regularization term $2\alpha\|\mathbf{f}_\ell - \mathbf{y}_\ell\|^2$ mentionned above, the optimization problem becomes

$$\min_{\mathbf{f}\in\mathbb{R}^n} -\mathbf{f}^\intercal\tilde{\mathbf{W}}\mathbf{f} + \alpha\|\mathbf{f}_\ell - \mathbf{y}_\ell\|^2 + \alpha\|\mathbf{f}_u\|^2. \tag{3.11}$$

This problem is also convex for all $\alpha > \|\tilde{\mathbf{W}}\|$, and $\mathbf{f}$ can also be obtained explicitly (see detail in Section A.1.2 of the appendix):

$$\mathbf{f} = \left(\mathbf{I}_n - \frac{\mathbf{Z}^\intercal\mathbf{A}\mathbf{Z}}{Tp}\right)^{-1}\mathbf{y}, \tag{3.12}$$

where

$$\mathbf{A} = \tilde{\mathbf{\Lambda}} \otimes \mathbf{I}_p, \textit{ with } \tilde{\mathbf{\Lambda}} = \frac{\mathbf{\Lambda}}{\alpha} \textit{ the hyperparameter matrix},$$
$$\mathbf{Z} = \sum_{t=1}^{T} \mathbf{E}_{tt}^{[T]} \otimes \mathbf{X}^t \textit{ the data matrix.}$$

In this formula, the output vector $\mathbf{f}$ is not limited to unlabeled data, but also gives an output score to labeled data. Similarly, as paradoxal as it might seem, the vector $\mathbf{y}$ contains labels for unlabeled data, with the convention $\mathbf{y}_u = 0$.

If we further use Woodbury matrix identity, we have

$$\mathbf{f} = \mathbf{y} + \frac{1}{Tp}\mathbf{Z}^\intercal \mathbf{A}^{\frac{1}{2}} \underbrace{\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}}_{=\mathbf{Q}} \mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{y}. \tag{3.13}$$

As we are only interested in the scores of unlabeled data, and using the fact that $\mathbf{y}_u = 0$, we can simplify this formula:

$$\mathbf{f}_u = \frac{1}{Tp}\mathbf{Z}_u^\intercal \mathbf{A}^{\frac{1}{2}} \underbrace{\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}}_{=\mathbf{Q}} \mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell\mathbf{y}_\ell. \tag{3.14}$$

Let us analyze once again the behavior of this formula for extremal values of α, namely $+\infty$ and $\|\tilde{\mathbf{W}}\|$.

- If $\alpha \to +\infty$, $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp} \to \mathbf{O}_{Tp}$, meaning that $\mathbf{Q} \to \mathbf{I}_{Tp}$, and the score of an unlabeled sample is
$$f_i \simeq \frac{\mathbf{x}_i^\intercal\mathbf{A}\mathbf{Z}_\ell}{Tp}\mathbf{y}_\ell.$$

- If $\alpha \to \|\tilde{\mathbf{W}}\|$, the largest eigenvalue of $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}$ becomes close to 1. If this eigenvalue is $1-\varepsilon$, with $\varepsilon \to 0$, and its associated eigenvector is $\mathbf{u}$. Then,
$$\mathbf{Q} \simeq \frac{1}{\varepsilon}\mathbf{u}\mathbf{u}^\intercal,$$
and the consequence on the solution (3.12) is that
$$\mathbf{f}_u \simeq \frac{1}{Tp\varepsilon}\underbrace{\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{u}}_{=\mathbf{v}_u^\intercal}\underbrace{\mathbf{u}^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell}_{=\mathbf{v}_\ell}\mathbf{y}_\ell.$$

As in the previous case, everything happens as if all the data was projected along a single axis, which is the first eigenvector of $\frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_u\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}$. It can be once again related to the PCA.

3.3.4 Final outcome

The only difference between equation (3.10) and (3.14) is the resolvant matrix, which is $\mathbf{Q}_u$ in the first case, and $\mathbf{Q}$ in the second case, meaning that in this last case, the "unsupervised part" of the solution makes use of all the data, and not only unlabeled data as in the first case. In the fully-supervised setting achieved when $\alpha \to +\infty$, both equations give the same solution, which is consistent with the previous remark.

With this single remark, it is hard to decide which one of the solution is the best suited to our problem. However, the problem with the relaxed fitting constraint also gave us the equation (3.13), which provides a different interpretation of the solution. Indeed, the first term can be interpreted as an *a priori* on the scores, corrected by the data through the second term. As for unlabeled data, there is no *a priori*, so the label value is 0, but this framework allows to consider "weakly-labeled" data, for which the *a priori* is not as strong as in the case of labeled data. This will be of particular interest in Chapter 5, where we consider uncertain labeling. The solution with the strict fitting constraint does not allow the same level of flexibility, so we will keep the solution with the relaxed fitting constraint. To conclude,

$$\mathbf{f}_u = \frac{1}{Tp}\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}\underbrace{\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}}_{=\mathbf{Q}}\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell\mathbf{y}_\ell, \tag{3.15}$$

with $\mathbf{Q} = \left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}$.

Thanks to this study, we have a solution to our optimization problem and an output score $\mathbf{f}_u$ for unlabeled data. Moreover, this solution has an explicit formulation. This vector sums up all the information we have to perform the classification. This score vector depends on several quantities:

- the label vector $\mathbf{y}_\ell$,
- the hyperparameter matrix $\mathbf{\Lambda}$,
- the regularization hyperparameter α.

So far, the effect of these quantities on the output score is unknown. To understand their role, and to choose them adequately, we need to perform a statistical analysis of the asymptotic behavior of the score vector $\mathbf{f}_u$.

4 Large dimensional analysis and improvement of multi-task semi-supervised learning

Thanks to the study of Chapter 3, we have an explicit formulation of the score vector $\mathbf{f}_u$ of unlabeled data. This vector sums up all the information we have to perform the classification. In a classical machine learning setting, the score f of a sample $\mathbf{x}$ belonging to $\mathcal{C}_1^t$ (resp. $\mathcal{C}_2^t$) is expected to be close to -1 (resp. $+1$). Therefore, the classification rule is:

$$\begin{aligned} f < \zeta^t &\implies \mathbf{x} \to \mathcal{C}_1^t, \\ f \geq \zeta^t &\implies \mathbf{x} \to \mathcal{C}_2^t, \end{aligned} \tag{4.1}$$

with $\zeta^t = 0$, where $\mathbf{x} \to \mathcal{C}$ means that $\mathbf{x}$ is classified in $\mathcal{C}$. However, as we will show later, choosing $\zeta^t = 0$ is not guaranteed to be optimal, even though it seems rather intuitive. Similarly, the arbitrary choice $y = \pm 1$ is not necessarily the best. To visualize that, one is interested in understanding the asymptotic statistical behavior of the score function f.

4.1 Statistical analysis

Before stating the theorem giving these asymptotics, one must introduce two fundamental small-dimensional quantities that will be of particular interest:

- The data matrix $\boldsymbol{\mathcal{M}} = \mathrm{M}^\intercal \mathrm{M}$, with $\mathrm{M} = \left[\boldsymbol{\mu}_1^1, \boldsymbol{\mu}_2^1, \boldsymbol{\mu}_1^2, \ldots, \boldsymbol{\mu}_2^T\right]$. It is worth to note that even though M is not accessible in practice, the data matrix $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{2T \times 2T}$ can be consistently estimated (see Section A.4.1 of the appendix).

- The parameter matrix $\boldsymbol{\mathcal{A}}$, which is the solution of the following system of equations:

$$\begin{cases} \forall t,\ \delta^t = \frac{1}{T} \boldsymbol{\mathcal{A}}_{tt}, \\ \boldsymbol{\mathcal{A}} = \tilde{\boldsymbol{\Lambda}} + \tilde{\boldsymbol{\Lambda}} \left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}^{-1} - \tilde{\boldsymbol{\Lambda}} \right)^{-1} \tilde{\boldsymbol{\Lambda}}. \end{cases} \tag{4.2}$$

with $\tilde{\delta}_j^t = \frac{\rho_j^t}{Tc(1-\delta^t)}$ and $\bar{\delta}^t = \tilde{\delta}_1^t + \tilde{\delta}_2^t$.
The equations leading to $\boldsymbol{\mathcal{A}}$ translates how the parameters and hyperparameters of each task interact with each other in the optimization problem.

These two quantities are mixed up in the following matrix:

$$\boldsymbol{\Theta}_0 = (\boldsymbol{\mathcal{A}} \otimes \mathbb{1}_2\mathbb{1}_2^\intercal) \odot \boldsymbol{\mathcal{M}}.$$

As long as uncertain labeling is not taken into account, it is natural to consider that labels are equal for datapoints in the same class, *i.e.,* $\forall\ \mathbf{x}_i^t \in \mathcal{C}_j^t, y_i^t = \tilde{y}_j^t \in \mathbb{R}$. Therefore we define $\tilde{\mathbf{y}} = [\tilde{y}_1^1, \tilde{y}_2^1, \tilde{y}_1^2, \ldots, \tilde{y}_2^T] \in \mathbb{R}^{2T}$, which sums up the label values. Precisely, we have $\mathbf{y}_\ell = \mathbf{D}\tilde{\mathbf{y}}$, with

$$\mathbf{D} = \sum_{t=1}^{T} \mathbf{E}_{tt}^{[T]} \otimes \begin{pmatrix} \mathbb{1}_{n_{\ell 1}^t} & \mathbb{0}_{n_{\ell 1}^t} \\ \mathbb{0}_{n_{\ell 2}^t} & \mathbb{1}_{n_{\ell 2}^t} \end{pmatrix}.$$

In this specific case, the statistics of the score function are given by the following theorem:

Theorem 21
Under Assumptions 19 and 20, for any unlabeled sample $\mathbf{x} \in \mathcal{C}_j^t$, and f being its associated score,

$$f \to \mathcal{N}\left(m_j^t, {\sigma^t}^2\right)$$

with $(1-\delta^t)m_j^t = {\mathbf{a}_j^t}^\intercal \tilde{\mathbf{y}}$ and $(1-\delta^t)\sigma^t = \sqrt{\tilde{\mathbf{y}}^\intercal \mathbf{B}^t \tilde{\mathbf{y}}}$,

$$\begin{aligned}
\mathbf{a}_j^t &= \left(\mathbf{e}_{t,j}^{[2T]\intercal}\left(\boldsymbol{\Theta} - \frac{Tc\delta^t}{\rho^t}\boldsymbol{\Gamma}\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\right)^\intercal, \\
\mathbf{B}^t &= \mathcal{D}_{\boldsymbol{\eta}}\left[2\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\left(\boldsymbol{\Theta} - \frac{Tc\delta^t}{\rho^t}\boldsymbol{\Gamma}\right) - \boldsymbol{\Gamma}^t\right]\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}} \\
&\quad + \mathcal{D}_{\boldsymbol{\eta}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\left(\boldsymbol{\Theta}\mathcal{D}_{\mathbf{r}^t}\boldsymbol{\Theta} + \bar{\boldsymbol{\Omega}}^t - \left(\frac{Tc\delta^t}{\rho^t}\right)^2\boldsymbol{\Gamma}^t\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}} \\
&\quad + \mathbf{T}^t \odot \mathcal{D}_{\boldsymbol{\rho}\odot\boldsymbol{\eta}}\mathcal{D}^{-1}_{(\bar{\boldsymbol{\rho}}\odot\bar{\boldsymbol{\eta}})\otimes\mathbb{1}_2},
\end{aligned}$$

and where

- $\boldsymbol{\Theta} = \left(\mathbf{I}_{2T} - \boldsymbol{\Theta}_0\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\right)^{-1}\boldsymbol{\Theta}_0 \in \mathbb{R}^{T\times T}$,
- $\boldsymbol{\Gamma} = \mathbf{I}_T \otimes \mathbb{1}_2\mathbb{1}_2^\intercal \in \mathbb{R}^{T\times T}$ *and* $\boldsymbol{\Gamma}^t = \mathbf{E}_{tt} \otimes \mathbb{1}_2\mathbb{1}_2^\intercal \in \mathbb{R}^{T\times T}$,
- $\bar{\boldsymbol{\Omega}}^t = \left(\mathbf{I}_{2T} - \boldsymbol{\Theta}_0\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\right)^{-1}\bar{\boldsymbol{\Omega}}_0^t\left(\mathbf{I}_{2T} - \mathcal{D}_{\tilde{\boldsymbol{\delta}}}\boldsymbol{\Theta}_0\right)^{-1} \in \mathbb{R}^{2T\times 2T}$,

- $\bar{\mathbf{\Omega}}_0^t = \left[\left(\boldsymbol{\mathcal{A}}\mathcal{D}_{\mathbf{e}_t^{[T]}+\bar{\mathbf{r}}^t}\boldsymbol{\mathcal{A}}\right) \otimes \mathbb{1}_2\mathbb{1}_2^\intercal\right] \odot \boldsymbol{\mathcal{M}} \in \mathbb{R}^{2T\times 2T}$,
- $\mathbf{T}^t = \mathcal{D}_{\bar{\mathbf{r}}^t \odot \bar{\boldsymbol{\eta}}} \otimes \mathbb{1}_2\mathbb{1}_2^\intercal \in \mathbb{R}^{2T\times 2T}$,
- $\mathbf{r}^t = \boldsymbol{\rho} \odot (\mathbf{S}_{t.} \otimes \mathbb{1}_2) \in \mathbb{R}^{2T}$ *and* $\bar{\mathbf{r}}^t = \bar{\boldsymbol{\rho}} \odot \mathbf{S}_{t.} \in \mathbb{R}^T$,
- $\mathbf{S} = \bar{\mathbf{S}}\left(\mathbf{I}_T - \mathcal{D}_{\bar{\boldsymbol{\rho}}}\bar{\mathbf{S}}\right)^{-1} \in \mathbb{R}^{T\times T}$ *with* $\bar{\mathbf{S}}_{tt'} = \frac{\boldsymbol{\mathcal{A}}^2_{tt'}}{T^2 c(1-\delta^{t'})^2}$.

A proof of this theorem is displayed in Section A.5 of the appendix. Beyond the cumbersome formulas, one can notice that the vector $\mathbf{a}_j^t$ and the matrix $\mathbf{B}^t$ are small dimensional deterministic quantities that only depend on estimates, known parameters and hyperparameters. Interestingly, even without analysing how $\mathbf{a}_j^t$ and $\mathbf{B}^t$ relate to the parameters of the problem, one can take a look at the implication of the Theorem 21 on our classification problem. The key information here is that the decision score f is asymptotically Gaussian with known parameters. These parameters are mere functionals of the label vector $\tilde{\mathbf{y}}$. Therefore, a good choice of $\tilde{\mathbf{y}}$ as well as ζ^t could lead to a lower classification error. First of all, we need to precise which quantity we want to minimize.

Definition 22
For a given target task t, the classification error of any unlabeled sample $\mathbf{x} \in \mathcal{C}_1^t$ is

$$\varepsilon_1^t = \mathbb{P}(\mathbf{x} \to \mathcal{C}_2^t | \mathbf{x} \in \mathcal{C}_1^t).$$

Similarly, the classification error of any unlabeled sample $\mathbf{x} \in \mathcal{C}_2^t$ is

$$\varepsilon_2^t = \mathbb{P}(\mathbf{x} \to \mathcal{C}_1^t | \mathbf{x} \in \mathcal{C}_2^t).$$

Remark 23
According to Theorem 21, and using the classification rule (4.1)*, the classification errors of Definition 22 rewrite as:*

$$\varepsilon_1^t = \mathcal{Q}\left(\frac{\zeta^t - m_1^t}{\sigma^t}\right) \quad \textit{and} \quad \varepsilon_2^t = \mathcal{Q}\left(\frac{m_2^t - \zeta^t}{\sigma^t}\right),$$

with $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{u^2}{2}}\,\mathrm{d}u$. Thus, $\zeta^t \mapsto \varepsilon_1^t$ is a decreasing function, while $\zeta^t \mapsto \varepsilon_2^t$ is an increasing function.

With this remark, we understand that the choice of the threshold ζ^t is a trade-off between the minimization of ε_1^t and ε_2^t, as displayed in Figure 4.1. $\zeta^t = m_1^t$ and $\zeta^t = m_2^t$ are extreme choices, where we have respectively $\varepsilon_1^t = \frac{1}{2}$ and $\varepsilon_2^t = \frac{1}{2}$. If $\zeta^t \notin [m_1^t, m_2^t]$, we have either $\varepsilon_1^t > \frac{1}{2}$ or $\varepsilon_2^t > \frac{1}{2}$, which is not a desirable situation.

So we will assume starting from now that $\zeta^t \in [m_1^t, m_2^t]$. Interestingly, for any such choice of ζ^t, the optimal label vector is the same.

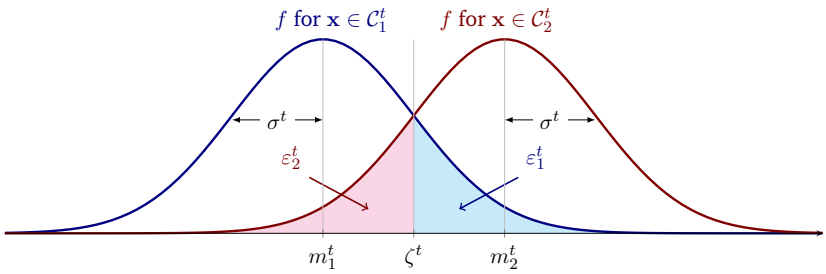

Fig. 4.1.: Asymptotic probability distribution of the score function f for samples of both classes. The classification errors expressed in Remark 23 can be interpreted as the area delimited by the density curve of f and the threshold ζ^t.

Proposition 24

For a given target task t and a given threshold $\zeta^t \in [m_1^t, m_2^t]$, there exists a unique (up to a positive multiplicative constant) score vector $\tilde{\mathbf{y}}^\star$ minimizing ε_1^t and ε_2^t, given as:

$$\tilde{\mathbf{y}}^\star = (\mathbf{B}^t)^{-1}(\mathbf{a}_2^t - \mathbf{a}_1^t). \tag{4.3}$$

This property provides the possibility, depending on the use of the algorithm, to choose the threshold accordingly. Indeed, the choice of ζ^t can be made in several different ways, allowing the user to tackle a wide variety of classification problems. For example :

- In a medical context, one could be interested to minimize the number of false positive ε_1^t while ensuring that the number of false negative ε_2^t is lower than a given threshold p (for example $p = 1\%$). Then the optimal ζ^t is given by:

$$\zeta^t = m_2^t - \sigma^t \mathcal{Q}^{-1}(p).$$

- One could also be interested in minimizing the overall classification error of the dataset $\frac{n_{u1}^t}{n_u^t}\varepsilon_1^t + \frac{n_{u2}^t}{n_u^t}\varepsilon_2^t$ (and therefore prioritize the classification of the most numerous class). Then the optimal ζ^t is given by:

$$\zeta^t = \frac{m_1^t + m_2^t}{2} + \frac{{\sigma^t}^2}{m_2^t - m_1^t} \log\left(\frac{n_{u1}^t}{n_{u2}^t}\right).$$

- If one wants to minimize undifferently ε_1^t and ε_2^t (so that $\varepsilon_1^t = \varepsilon_2^t$), then the optimal ζ^t is straightforwardly given by:

$$\zeta^t = \frac{m_1^t + m_2^t}{2}.$$

  This situation is a specific case of the previous one with $n_{u1}^t = n_{u2}^t$. It can also be interpreted as minimizing the overall classification error without any *a priori* on the genuine class of unlabeled samples. We will focus on this last simple example to provide a lower bound of the classification error.

Proposition 25
For a given target task t, the minimal value of the classification error $\varepsilon^t = \frac{\varepsilon_1^t + \varepsilon_2^t}{2}$ is achieved with the optimal label $\tilde{\mathbf{y}}^\star$, and is asymptotically given by:

$$\varepsilon_\star^t = \mathcal{Q}\left(\frac{1}{2}\sqrt{(\mathbf{a}_2^t - \mathbf{a}_1^t)^\intercal (\mathbf{B}^t)^{-1}(\mathbf{a}_2^t - \mathbf{a}_1^t)}\right). \tag{4.4}$$

Propositions 24 and 25 are obtained from classical convex optimization tools, and their proofs are provided respectively in the Sections A.2 and A.3 of the appendix.

4.2 Hyperparameter optimization

There are two hyperparameters to consider in our optimization problem, which are α and $\mathbf{\Lambda}$. They always appear jointly in the solution, through the matrix $\tilde{\mathbf{\Lambda}} = \frac{\mathbf{\Lambda}}{\alpha}$. It means that they can be optimized separately, and that the optimization of $\mathbf{\Lambda}$ only needs to be done up to a multiplicative constant, as long as it precedes the optimization of α.

4.2.1 Optimization of the hyperparameter matrix

As a reminder, the hyperparameter matrix $\mathbf{\Lambda}$ filters how much each task is related to the others in the optimization problem. The higher $\Lambda^{tt'}$ is, the more Task t and Task t' will be considered jointly. Let us take a look at some extreme cases to understand better the role of $\mathbf{\Lambda}$:

- If $T = 1$, we are in a single-task setting and $\mathbf{\Lambda}$ is a scalar value. Therefore, the choice of $\mathbf{\Lambda}$ does not matter, as long as α is choosen wisely (as we will see that in Section 4.2.2). We can agree to set $\mathbf{\Lambda} = 1$, to neutralized the effect of $\mathbf{\Lambda}$ on the optimization problem. We will keep this value of 1 as the neutral value.

For instance, Λ^{tt} should always be equal to 1, because a task is always fully related to itself.

- If all tasks are equivalent (meaning that all the data could have actually been in the same task), then for all (t, t'), $\Lambda^{tt'}$ should be equal to 1, so that every task is related similarly to the other. Thus $\mathbf{\Lambda} = \mathbb{1}\mathbb{1}^\intercal$.

- If all tasks are completely different one from each other, then there should be no connection of the tasks in the optimization problem. To achieve that, $\Lambda^{tt'}$ should be equal to 0 for every $t \neq t'$. Thus $\mathbf{\Lambda} = \mathbf{I}_T$.

To summarize, $\Lambda^{tt'}$ should take a value between 0 and 1, depending on how much Task t and Task t' are related, 0 corresponding to completely uncorrelated tasks and 1 corresponding to identical tasks. It is worth to mention that if two tasks are negatively correlated (meaning that the two classes are inverted from one task to the other), then the value of $\Lambda^{tt'}$ should be 1, as the choice of $\tilde{\mathbf{y}}$ will already tackle the negative correlation. Therefore, the values of $\mathbf{\Lambda}$ should always be positive. Taking these remarks into account, we propose the following formula:

$$\Lambda^{tt'} = \frac{|\langle \boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t, \boldsymbol{\mu}_1^{t'} - \boldsymbol{\mu}_2^{t'} \rangle|}{\|\boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t\| \|\boldsymbol{\mu}_1^{t'} - \boldsymbol{\mu}_2^{t'}\|}. \tag{4.5}$$

As expected $\Lambda^{tt'}$ increases if the tasks are more correlated. This quantity is intuitively associated to the alignement between tasks. The vector $\mathbf{v}^t = \boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t$ is the vector orthogonal to the hyperplane that best discriminates the two classes. So $\Lambda^{tt'}$ is the absolute value of the normalized scalar product between the vectors $\mathbf{v}^t$ and $\mathbf{v}^{t'}$.

This quantity is simple, intuitive, and can be consistently estimated (see Section A.4.2). Most importantly, simulations comparing this choice to the optimal bound (given by information theory) suggests that this choice is close to optimal. Figure 4.2 compares:

- The naive choice $\mathbf{\Lambda} = \mathbb{1}_T \mathbb{1}_T^\intercal$.

- $\mathbf{\Lambda}$ given by equation (4.5), with the oracle knowledge of the $\boldsymbol{\mu}_j^t$.

- $\mathbf{\Lambda}$ given by equation (4.5), estimated from the data (see Section A.4.2 for more details).

- The optimal bound from information theory.

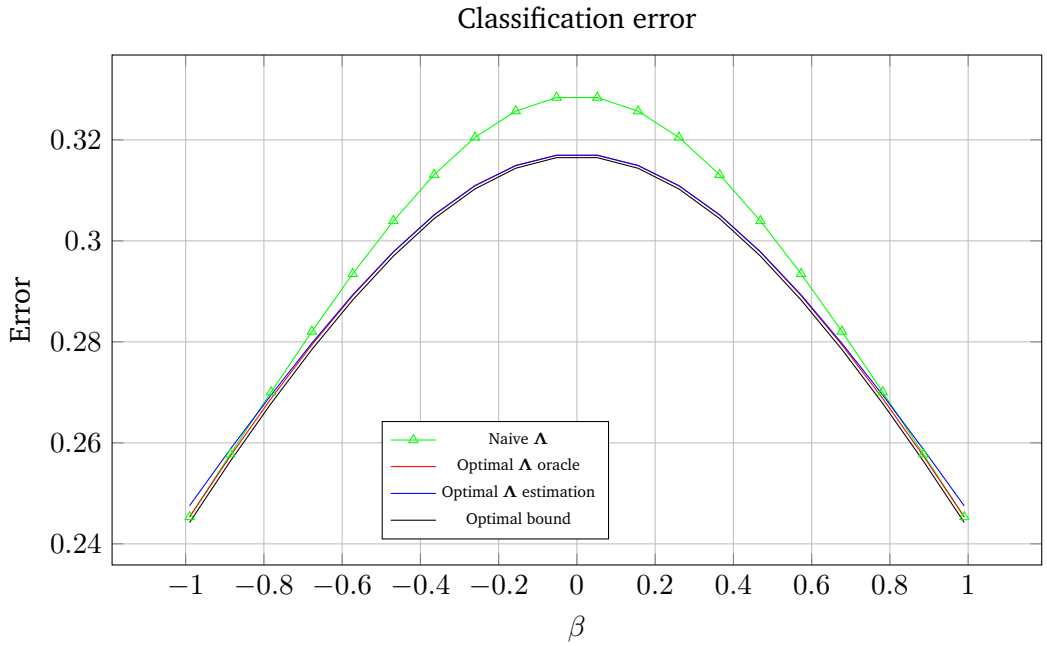

Fig. 4.2.: Classification error depending on correlation between tasks, for different choices of $\mathbf{\Lambda}$. Using $\Lambda^{(tt')}$ equal to squared correlation between task t and t' gives practically optimal results

4.2.2 Optimization of the regularization hyperparameter

One of the key results of Section 4.1 is that we can compute $\varepsilon_\star^t$ with the simple knowledge of the parameters of the problem. In particular, $\varepsilon_\star^t$ can be computed for different values of the regularization hyperparameter α, at a low computational cost. This means that a grid search over the interval $]\|\tilde{\mathbf{W}}\|, +\infty[$ gives us the best value of α to minimize the final classification error.

In most machine learning algorithms, the hyperparameter optimization is done through a process called cross-validation, which is costly both in computing power and in data. In that process, the algorithm is trained again for every possible value of parameter. Here, the statistical understanding of the algorithm allows to short-cut this process and train the algorithm just once.

To perform the minimization over $]\|\tilde{\mathbf{W}}\|, +\infty[$, we use the *golden-section search*, which is an effective method to find the minimum of a given unimodal function f [50]. It is a rather slow, but very robust method. It is analog to the bisection method, used to find the zero of a function. In the bisection method, one starts with a couple of point (x_1, x_2) which defines an interval $[x_1, x_2]$ in which their exists a point where the zero is achieved. The function is negative at one of this point, and positive at the other one, ensure that the zero actually belongs in the interval. Then,

the interval is splitted in two by a new point x_3, and depending whether $f(x_3)$ is positive or negative, a new interval is created out of the 3 points.

A key difference between the golden-section search and the bisection method is that a couple of point is not sufficient anymore, but we need to start instead with a *bracketing interval*, *i.e.,* a triplet (x_1, x_2, x_3) such that $f(x_2) < f(x_1)$ and $f(x_2) < f(x_1)$. The fact that the function is lower at x_2 than at the endpoints of the interval ensures that the minimum is indeed in the interval. Then the largest interval between $[x_1, x_2]$ and $[x_2, x_3]$ (let say $[x_2, x_3]$ to simplify) is splitted in two by a new point x_4, and depending whether $f(x_4)$ is larger or smaller than $f(x_2)$, a new smaller bracketing interval is made out of the 4 points :

- If $f(x_4) > f(x_2)$, then (x_1, x_2, x_4) is a bracketing interval.
- If $f(x_4) < f(x_2)$, then (x_2, x_4, x_3) is a bracketing interval.

The name "golden-section" comes from the fact that the ratio of the lengths $x_3 - x_2$ and $x_2 - x_1$ is either $\frac{\phi}{1+\phi}$ or $\frac{1+\phi}{\phi}$, where ϕ the golden number. This ratio maximizes the convergence speed of the method, similarly to the ratio $\frac{1}{2}$ which maximizes the convergence speed of the bisection method.

To initialize this method, one needs a bracketing interval. If $]\|\tilde{\mathbf{W}}\|, +\infty[$ is acyually a bracketing interval, the method cannot be initialized with an infinite value, so we propose the following algorithm to find a crude bracketing interval. Let us denote $x_0 = \|\tilde{\mathbf{W}}\|$ the left-endpoint of the search interval.

Algorithm 1 Initialization of the golden-section search

Input: Function $f : \alpha \mapsto \varepsilon_\star^t$
Output: A bracketing interval (x_1, x_2, x_3) of f
Initilize the bracketing triplet with $(x_1, x_2, x_3) = (1.5, 2, 3) * x_0$
while (x_1, x_2, x_3) is not a bracken=ting interval **do**

  if $f(x_1) < f(x_2) < f(x_3)$ **then**
    $(x_1, x_2, x_3) = (\frac{x_1+1}{2}, x_1, x_2)$
  else
    $(x_1, x_2, x_3) = (x_2, x_3, 2x_3 - 1)$
  end if
end while

Our initial interval is choosen such that $x_2 - x_1 = x_1 - x_0$ and $x_3 - x_2 = 2(x_2 - x_1)$. Each step of the algorithm either multiplies all the values by 2 or divides all the values by 2. This guarantees that the process will not be too long, even if the minimum is far from the left-endpoint of the search interval, or conversely if it is very close to the left-endpoint.

4.2.3 Computation of the regularization hyperparameter's lower bound

The grid search of α is performed over the interval $]\|\tilde{\mathbf{W}}\|, +\infty[$, which suggests that we know the value of $\|\tilde{\mathbf{W}}\|$. In fact, this value can be computed numerically, but implies the computation of the n eigenvalues of the matrix $\frac{\mathbf{Z}^\intercal \mathbf{A} \mathbf{Z}}{Tp}$, which is costly in computation power. If the largest eigenvalue of this matrix is asymptotically deterministic, in practice it fluctuates a little for every new realization of the random variables. In some cases, when the algorithm is mostly unsupervised, the optimal value of α is really close to $\|\tilde{\mathbf{W}}\|$, and a small fluctuation of $\|\tilde{\mathbf{W}}\|$ might jeopardize the whole algorithm. Indeed, when α is tuned, the quantity of interest that is optimized in practice is $\alpha - \|\tilde{\mathbf{W}}\|$, which appears in the resolvant $\mathbf{Q}$ to boost its first eigenvalues. As $\alpha - \|\tilde{\mathbf{W}}\|$ has to be tuned precisely in this case, a small change in the value of $\|\tilde{\mathbf{W}}\|$ can have dramatic consequences. Therefore, it would be safer to have a deterministic value for the lower bound of the interval over which the grid search is performed.

One of the key consequence of the choice of α is the outcome of the system of equations (4.2). The parameter matrix $\boldsymbol{\mathcal{A}}$ is the solution of a fixed point equation parametrized by α. Let us define the function:

$$
\begin{aligned}
f : \mathbb{R} \times \mathbb{R}^T &\to \mathbb{R}^T \\
(\alpha, \bar{\mathbf{y}}) &\mapsto \operatorname{diag}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}(\mathcal{D}_{\bar{\mathbf{y}}} - \boldsymbol{\Lambda})^{-1}\boldsymbol{\Lambda}) + \frac{\bar{\mathbf{y}} \odot \bar{\boldsymbol{\rho}}}{c} - \alpha T \mathbb{1}_T,
\end{aligned}
\tag{4.6}
$$

where $\operatorname{diag} : \mathbb{R}^{T \times T} \to \mathbb{R}^T$ is the function that returns the diagonal of a given matrix ($\operatorname{diag}(\mathbf{M})_t = \mathbf{M}_{tt}$), and $\bar{\mathbf{y}}$ is the vector defined by $\bar{\mathbf{y}}_t = \alpha(\bar{\delta}^t)^{-1}$. The function f is such that (4.2) is equivalent to $f(\alpha, \bar{\mathbf{y}}) = \mathbf{0}$. In Figure 4.3, we observe that the solution $\bar{\mathbf{y}}$ of $f(\alpha, \bar{\mathbf{y}}) = \mathbf{0}$, taken as a function of α, is smooth on a given interval, and that there is a discontinuity of the derivative, close to the bound $\|\tilde{\mathbf{W}}\|$, that is coincident with a collapse of the classification performance. Such value, if it can be computed, could be used as lower bound to the search of α. We will denote this quantity α_0.

The function f is such that (4.2) is equivalent to $f(\alpha, \bar{\mathbf{y}}) = \mathbf{0}$. Therefore, we can apply the implicit function theorem to f. That is, if $\frac{\partial f}{\partial \bar{\mathbf{y}}}$ is invertible, then there exists a function φ such that $\bar{\mathbf{y}} = \varphi(\alpha)$, where $\frac{\partial f}{\partial \bar{\mathbf{y}}}$ is the Jacobian matrix of f. Moreover, φ is in the same differentiability class than f. This means that, if φ is not $\mathcal{C}^1$ in α_0, then either f is not $\mathcal{C}^1$ in α_0, either $\frac{\partial f}{\partial \bar{\mathbf{y}}}(\alpha_0, \bar{\mathbf{y}})$ is not invertible. The value of α_0 we want to find is precisely characterized by the fact that φ is not $\mathcal{C}^1$ in α_0. As f is $\mathcal{C}^1$

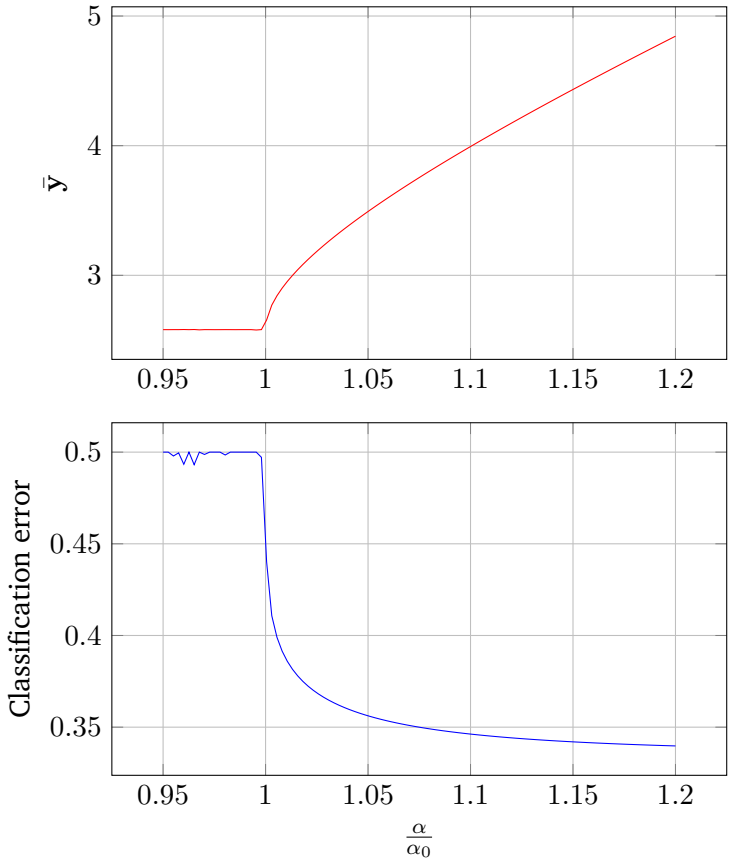

Fig. 4.3.: Joint evolution of $\bar{\mathbf{y}}$ and the classification error ε as functions of $\frac{\alpha}{\alpha_0}$, on the range $[0.95\alpha_0, 1.2\alpha_0]$ ($T = 1$, $p = 1000$, $n_{\ell 1} = n_{\ell 2} = n_{u1} = n_{u2} = 100$). The discontinuity of the derivative of $\bar{\mathbf{y}} = \phi(\alpha)$ (top figure) is coincident with a collapse of the classification performance (bottom figure).

on its domain, we deduce that $\frac{\partial f}{\partial \bar{\mathbf{y}}}(\alpha_0, \bar{\mathbf{y}})$ is not invertible. The condition "$\frac{\partial f}{\partial \bar{\mathbf{y}}}(\alpha, \bar{\mathbf{y}})$ is not invertible" is a necessary condition of $\alpha = \alpha_0$. Let us analyze this condition. First of all, we need to compute $\frac{\partial f}{\partial \bar{\mathbf{y}}}$:

$$\frac{\partial f_t}{\partial \bar{y}^{t'}}(\alpha, \bar{\mathbf{y}}) = -\left[\mathbf{\Lambda}(\mathcal{D}_{\bar{\mathbf{y}}} - \mathbf{\Lambda})^{-1}\mathbf{E}_{t't'}(\mathcal{D}_{\bar{\mathbf{y}}} - \mathbf{\Lambda})^{-1}\mathbf{\Lambda}\right]_{t,t} + \mathbb{1}_{t=t'}\frac{\bar{\rho}^t}{c} \tag{4.7}$$

The invertibility of $\frac{\partial f}{\partial \bar{\mathbf{y}}}(\alpha, \bar{\mathbf{y}})$ is hard to characterize. It would be easier to work instead with a 1-dimensional function $\tilde{f} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Then, $\frac{\partial \tilde{f}}{\partial \tilde{y}}(\alpha, \tilde{y})$ would be invertible if and only if $\frac{\partial \tilde{f}}{\partial \tilde{y}}(\alpha, \tilde{y}) \neq 0$. To reduce our problem to a 1-dimensional function, there are two things to do:

- Reduce the output of f to a unique scalar value. We can for example consider the function f_{t_0}, where t_0 is still to be determined.

- Reduce the T-dimensional variable $\bar{\mathbf{y}}$ to a scalar variable $\tilde{y}$. To this end, we will decompose $\bar{\mathbf{y}}$ as $\bar{\mathbf{y}} = \tilde{y}\bar{\mathbf{s}}$, assuming that we know $\bar{\mathbf{s}}$, meaning that we know in a sense the structure of the vector $\bar{\mathbf{y}}$.

Once we have our 1-dimensional function $\tilde{f}$, we can repeat the two following steps to compute $\bar{\mathbf{y}}$:

- If we have access to $\bar{\mathbf{s}}$, then we can compute $\tilde{y}$ such that $\frac{\partial \tilde{f}}{\partial \tilde{y}}(\alpha, \tilde{y})$ is not invertible.
- If $\tilde{y}$ is known, we can update $\bar{\mathbf{s}}$ by using the T equations from $f(\alpha, \bar{\mathbf{y}}) = \mathbf{0}$.

Before repeating these two steps, we need to choose a starting value to $\bar{\mathbf{s}}$. As a first approximation, we can use the fact that $\delta^t \simeq 0$ in the limit $\alpha \to \infty$, which leads to the following simplification:

$$\bar{y}^t = \alpha(\bar{\delta}^t)^{-1} = \frac{\alpha T c(1-\delta^t)}{\bar{\rho}^t} \simeq \frac{\tilde{y}}{\bar{\rho}^t} \tag{4.8}$$

In other terms, $\bar{\boldsymbol{\rho}}^{-1}$ drives the structure of $\bar{\mathbf{y}}$. We will therefore use $\bar{\mathbf{s}} = \bar{\boldsymbol{\rho}}^{-1}$ for the first iteration. Let us now describe more precisely each of the two steps mentionned above:

- Let us assume that we know $\bar{\mathbf{s}}$. We need to build a function $\tilde{f} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ such that (4.2) $\Leftrightarrow \tilde{f}(\alpha, \tilde{y}) = 0$. The function $\tilde{f}(\alpha, \tilde{y}) = f_{t_0}(\alpha, \tilde{y}\bar{\mathbf{s}})$ could work, but we need to choose wisely which value f_{t_0} of the output vector of f is the most relevant. Let us start by defining the new function:

$$\begin{aligned} &\tilde{f} : \mathbb{R} \times \mathbb{R} \to \mathbb{R} \\ &\quad (\alpha, \tilde{y}) \mapsto \left[\boldsymbol{\Lambda} + \boldsymbol{\Lambda}(\tilde{y}\mathcal{D}_{\bar{\mathbf{s}}} - \boldsymbol{\Lambda})^{-1}\boldsymbol{\Lambda}\right]_{t_0,t_0} + \tilde{y}\frac{\bar{s}^{t_0}\bar{\rho}^{t_0}}{c} - \alpha T, \end{aligned} \tag{4.9}$$

  Its (1-dimensional) Jacobian matrix can be computed:

$$\begin{aligned} \frac{\partial f}{\partial \tilde{y}}(\alpha, \tilde{y}) &= -\left[\boldsymbol{\Lambda}(\tilde{y}\mathcal{D}_{\bar{\mathbf{s}}} - \boldsymbol{\Lambda})^{-1}\mathcal{D}_{\bar{\mathbf{s}}}(\tilde{y}\mathcal{D}_{\bar{\mathbf{s}}} - \boldsymbol{\Lambda})^{-1}\boldsymbol{\Lambda}\right]_{t_0,t_0} + \frac{\bar{s}^{t_0}\bar{\rho}^{t_0}}{c} \\ &= -\bar{s}^{t_0}\left[\mathcal{D}_{\bar{\mathbf{s}}}^{-1}\boldsymbol{\Lambda}(\tilde{y}\mathbf{I}_T - \mathcal{D}_{\bar{\mathbf{s}}}^{-1}\boldsymbol{\Lambda})^{-2}\mathcal{D}_{\bar{\mathbf{s}}}^{-1}\boldsymbol{\Lambda}\right]_{t_0,t_0} + \frac{\bar{s}^{t_0}\bar{\rho}^{t_0}}{c} \\ &= -\bar{s}^{t_0}\left[\mathbf{M}(\tilde{y}\mathbf{I}_T - \mathbf{M})^{-2}\mathbf{M}\right]_{t_0,t_0} + \frac{\bar{s}^{t_0}\bar{\rho}^{t_0}}{c}, \end{aligned} \tag{4.10}$$

  where $\mathbf{M} = \mathcal{D}_{\bar{\mathbf{s}}}^{-1}\boldsymbol{\Lambda}$. Therefore,

$$\frac{\partial f}{\partial \tilde{y}}(\alpha, \tilde{y}) = 0 \Leftrightarrow \left[\mathbf{M}(\tilde{y}\mathbf{I}_T - \mathbf{M})^{-2}\mathbf{M}\right]_{t_0,t_0} = \frac{\bar{\rho}^{t_0}}{c}. \tag{4.11}$$

Schematically, as $\tilde{y}$ decreases, $\left[\mathbf{M}(\tilde{y}\mathbf{I}_T - \mathbf{M})^{-2}\mathbf{M}\right]_{t_0,t_0}$ grows, until it is larger than $\frac{\bar{\rho}^{t_0}}{c}$, therefore satistfying equation (4.11). Our t_0 of interest is the first one to satisfy (4.11) when $\tilde{y}$ decreases. The key value is the largest diagonal value of $\mathbf{M}$, which is also the largest diagonal value of $\mathbf{M}(\tilde{y}\mathbf{I}_T - \mathbf{M})^{-2}\mathbf{M}$ (it causes the term $(\tilde{y}\mathbf{I}_T - \mathbf{M})^{-2}$ to explode when $\tilde{y}$ becomes close to this diagonal value). This largest diagonal value is $\mathbf{M}_{t_0,t_0}$, where

$$t_0 = \arg\max_t \mathbf{M}_{t,t} = \arg\max_t (\bar{s}^t)^{-1} = \arg\max_t \bar{\rho}^t. \tag{4.12}$$

We will use this value of t_0. From there, if one has access to $\bar{\mathbf{s}}$, one can compute $\tilde{y}$ by solving (4.11).

- We need to update $\bar{\mathbf{s}}$ knowing $\tilde{y}$. First of all, we will use the fact that, in the first choice of $\bar{\mathbf{s}}$, $\bar{y}^{t_0} = \frac{\tilde{y}}{\bar{\rho}^{t_0}}$. We will enforce that in each update of $\bar{\mathbf{s}}$. We now need $T-1$ additional equations to build $\bar{\mathbf{s}}$. Let us consider the vector

$$\mathbf{v} = \text{diag}(\mathbf{\Lambda} + \mathbf{\Lambda}(\mathcal{D}_{\bar{\mathbf{y}}} - \mathbf{\Lambda})^{-1}\mathbf{\Lambda}) + \frac{\bar{\mathbf{y}} \odot \bar{\boldsymbol{\rho}}}{c}. \tag{4.13}$$

Then,

$$\begin{aligned} f(\alpha, \bar{\mathbf{y}}) = 0 \Leftrightarrow \mathbf{v} &= \alpha T \mathbb{1}_T \\ &\Rightarrow \forall\ 0 \leq t \leq T-1, \mathbf{v}_t = \mathbf{v}_{t+1}, \end{aligned} \tag{4.14}$$

which gives us the $T-1$ equations we need to build $\bar{\mathbf{s}}$ knowing $\tilde{y}$.

Once $\bar{\mathbf{y}}$ is known, we can compute α_0 by using $f(\alpha_0, \bar{\mathbf{y}}) = \mathbf{0}$, which leads to:

$$\alpha_0 = \frac{1}{T}\text{diag}(\mathbf{\Lambda} + \mathbf{\Lambda}(\mathcal{D}_{\bar{\mathbf{y}}} - \mathbf{\Lambda})^{-1}\mathbf{\Lambda}) + \frac{\bar{\mathbf{y}} \odot \bar{\boldsymbol{\rho}}}{c} \tag{4.15}$$

4.3 Improved algorithm and main limitations

We now see that the Theorem 21 provides us:

- An optimal choice $\tilde{\mathbf{y}}^\star$ of the label vector $\tilde{\mathbf{y}}$.
- The information we need to choose wisely the threshold ζ^t.
- The optimal classification error $\varepsilon_\star^t$ for a specific choice of ζ^t.

This leads to Algorithm 2, which summarizes the previous results and remarks. This algorithm is the outcome of the Chapters 3 and 4, and has benefited from the improvements enabled by the mathematical analysis performed throughout these chapters.

Algorithm 2 Improved algorithm

Input: labeled data $\mathbf{X}_\ell = [\mathbf{X}_\ell^1, \ldots, \mathbf{X}_\ell^T]$ and unlabeled data $\mathbf{X}_u = [\mathbf{X}_u^1, \ldots, \mathbf{X}_u^T]$
Output: Estimated class $\hat{j} \in \{1, 2\}$ for unlabeled data of a given target task t
Center data per task following (3.4)
Compute data matrices $\mathbf{Z}_\ell$ and $\mathbf{Z}_u$ from $\mathbf{X}_\ell$ and $\mathbf{X}_u$
Estimate matrix $\mathcal{M}$ (cf. Section A.4)
Create scores $\tilde{\mathbf{y}}^\star$ from (4.3) and Λ from (4.5).
Estimate the classification error $\varepsilon_\star^t$ according to (4.4) and optimize with respect to α using a grid search approach (cf. Section 4.2.2).
Compute classification scores $\mathbf{f}_u$ according to (3.15).
Output: $\hat{j}$ such that $f_i \underset{\hat{j}=1}{\overset{\hat{j}=2}{\gtrless}} \frac{m_1^t + m_2^t}{2}$.

However, despite these improvements, this algorithm still has some limitations that we will describe now.

4.3.1 Multiclass setting

This thesis focuses on a binary setting. There are some known methods to deal with a higher number of classes, such as *one-vs-all* and *one-vs-one* [51,52]. These methods can be roughly implemented in our case, but each one has some limitations. We do not pretend to analyze or implement these methods, but rather give an overview of what could be done, and which hurdles could be encountered. In this paragraph, $m > 2$ will denote the number of classes.

- **One-vs-all** : m binary classifiers are trained, each one separating one class $\mathcal{C}_j$ from the other $m - 1$ classes. Each new sample is then classified into the class with the highest score among the m classifiers. The computations of Theorem 21 can be adapted to these classifiers, but beyond that, this approach has several issues :
  - Each classifier has its own bias, making impossible a fair comparison between the output scores. To work well, this method implicitly makes the assumption that for all $1 \leq j \leq m$, the output distribution of the classifier $\mathcal{C}_j$-vs-all on samples from class $\mathcal{C}_j$ is the same. One way to tackle this problem is to normalize the distribution (*i.e.,* consider $\frac{f_i^t - m_j^t}{\sigma^t}$ instead

of f_i^t), but the algorithm then depends too heavily on the estimations of m_j^t and σ^t, affecting its robustness.

  - The optimal score vector $\tilde{\mathbf{y}}^\star$ does not have an explicit formula, as the quantity to maximize is not a quadratic form of $\tilde{\mathbf{y}}$ anymore (see Section A.2 of the appendix). As such, one needs to approximate the quantity to maximize, therefore losing the optimality.

- **One-vs-one** : $\frac{1}{2}m(m-1)$ binary classifiers are trained, each one separating one class $\mathcal{C}_j$ from an other class $\mathcal{C}_{j'}$. For every sample of unlabeled data, each one of these classifiers "votes" for the class the most relevant among the two that it compares. Then, the sample is attributed to the class collecting the most votes. This approach also comes with several issues :

  - To train a binary classifier $\mathcal{C}_j$-vs-$\mathcal{C}_{j'}$, one can get rid of labeled data from other classes, but not unlabeled data. As such, the unsupervised part of the algorithm will still make use of the unlabeled data of other classes and change the outcome of the score function, necessarily introducing a new bias. However, results of Theorem 21 can be adapted to this situation.

  - The number of one-vs-one classifiers to train is a quadratic function of the number of classes, while the number of one-vs-all classifiers was a linear function of the number of classes. With a high number of classes, the one-vs-one method is therefore much slower.

To our knowledge, one-vs-one is the only method able to keep the optimality we have in the binary case. The major issue is that a high number of classes makes the method heavy to use.

4.3.2 Lack of labeled data

Even if the model is designed to work for any number of labeled and unlabeled data, in practice it cannot do without a minimum number of labeled data. First of all, if there is no labeled data at all (as it would be the case in a purely unsupervised setting), the score vector $\mathbf{f}$ would be zero as the label vector $\mathbf{y}$ would be zero itself. Most importantly, labeled data is needed to perform the estimation of $\boldsymbol{\mathcal{M}}$ and $\boldsymbol{\Lambda}$. If labeled data is too scarce, the algorithm could be misleaded by the estimated values of $\boldsymbol{\mathcal{M}}$ and $\boldsymbol{\Lambda}$, implying poor performances.

4.4 Experiments

The experiments displayed in this section are done on both synthetic and real-world datasets. With Gaussian synthetic data, we simulate different learning settings, in order to convey some intuition on the different aspects of our model : multi-task learning, semi-supervised learning, uncertain labeling, floating labels... Meanwhile, the purpose of real-data experiments is to show that our algorithm is robust enough to keep its properties on dataset that are not isotropic Gaussian vectors anymore. The code used to genereate the following experiments is available at https://gricad-gitlab.univ-grenoble-alpes.fr/legervi/tsp

4.4.1 Multitask experiments

This section investigates the influence of task correlation on the performances of our algorithm. To do so, we consider a case of transfer learning between a source task and a target task. The source and target tasks are mixtures of two Gaussians: $\mathcal{N}(\pm\boldsymbol{\mu}, \mathbf{I}_p)$ for the source and $\mathcal{N}(\pm(\beta\boldsymbol{\mu} + \sqrt{1-\beta^2}\boldsymbol{\mu}^\perp), \mathbf{I}_p)$ for the target, where $\boldsymbol{\mu}^\perp$ is a vector orthogonal to the vector $\boldsymbol{\mu}$. This setting allows through β to control the similarity between both tasks. Specifically, for $\beta = 0$ the tasks are unrelated while for $\beta = 1$ they are identical. Note that $\beta = -1$ corresponds to a scenario where the classes of target and source tasks are reversed.

Figure 4.4 displays the empirical versus theoretical classification error of our algorithm as a function of β for the optimized method (which integrates the label and hyperparameter optimization) against the naive method which leaves the labels (± 1) untouched. The values of the optimal labels are displayed jointly in the top figure.

We also provide an optimal bound of performance established by information theory in [18]. This bound corresponds to the best performance achievable by any algorithm given the setting of our experiment. The figure clearly shows that our proposed optimized algorithm is extremely close to the information-theoretic optimum under our Assumptions 19 and 20. The close fit between theoretical and empirical performances confirms Proposition 25 even for finite-dimensional data.

We also observe a strong robustness of our method to negative transfer. Indeed, when β becomes negative, the error increases for the naive algorithm, while our proposed method is insensitive to the sign of β by a dynamic adaptation of the labels. The top figure shows the sign inversion of $\tilde{y}_1^2$ and $\tilde{y}_2^2$ when β becomes negative. In

the specific case where $\beta = -1$, our algorithm has the same performance as with $\beta = 1$, while the naive "conventional" approach gets worse than random guess. Besides, when β is close to 0, the labels of the source task have low magnitude, because there is no much information to get from source task.

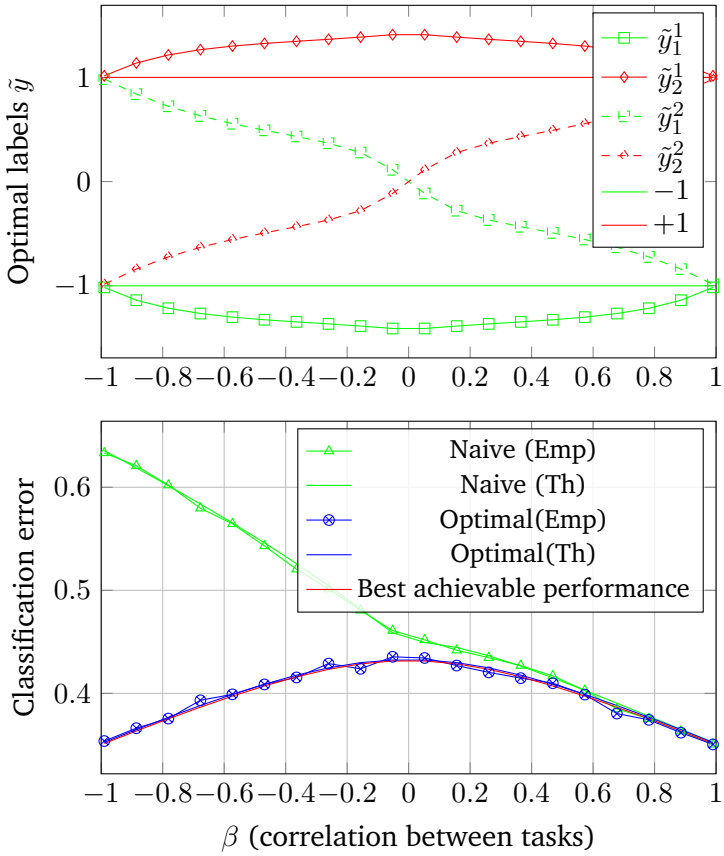

Fig. 4.4.: Joint evolution of optimal labeling and classification error as a function of correlation between tasks ($p = 200$, $n_\ell^1 = 100$, $n_\ell^2 = 1000$, $n_u^1 = n_u^2 = 250$). (**Top**) Optimal labels with normalization $\|\tilde{\mathbf{y}}\| = 1$. Optimal labels adapt themselves to avoid negative transfer (**Bottom**) Classification error for both naive and optimal algorithms. Our algorithm is close to optimal, while naive labels induce a negative transfer when tasks are related negatively.

4.4.2 Class imbalances

In the previous experiment, the number of data in each class was the same. However, our adaptive labeling is also useful to deal with class imbalances. Indeed, labels can be interpreted as weights, which should naturally be higher for underrepresented classes. We will consider a single-task learning scenario with a fixed amount of data. While the number of unlabeled data is equal for both classes, the number of labeled data is unbalanced.

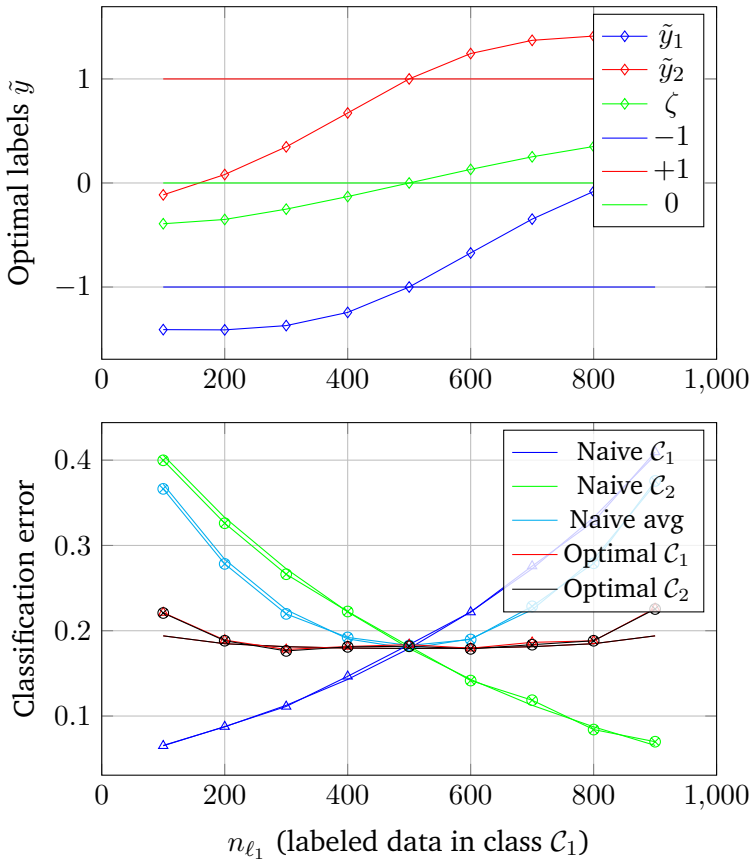

Fig. 4.5.: Joint evolution of optimal labeling and classification error as a function of the number of labeled data in class $\mathcal{C}_1$. The total number of labeled data is constant ($n_\ell = n_{\ell 1} + n_{\ell 2} = 1000$, $p = 200$, $n_{u1} = n_{u2} = 200$). (**Top**) Optimal labels with normalization $\|\tilde{\mathbf{y}}\| = 1$, and optimal threshold ζ (also normalized). Optimal labels adapt themselves to compensate the class imbalances (**Bottom**) Theoretical and empirical classification error for both naive and optimal labels and threshold. The overall error is better with our algorithm, while naive labels and threshold induce a high error for the most represented class.

As such, Figure 4.5 displays the classification error as a function of the number of labeled data in class $\mathcal{C}_1$ (bottom figure), jointly with the values of the optimal labels (top figure). We also display the value of the optimal threshold $\zeta = \frac{m_1+m_2}{2}$, which ensures that the probability of misclassifying each sample is the same whether this sample is in a class or in the other. As expected, the label value of the least represented class has a higher magnitude, as each labeled sample belonging to this class carries more information. Because of the imbalance, the means of the two classes are not symetrical about zero anymore, and the threshold value must be changed.

Because of the centering operation performed in (3.4), the mean $\boldsymbol{\mu}_j^t$ of the least represented class is higher in norm than the mean of the most represented class.

As a consequence, if the labels are not adapted to counter this phenomenon, the score mean m_j^t of the least represented class will be further away from zero than the score mean of the most represented class, as represented in Figure 4.6. In the naive setting, where the threshold has a default value of zero, the classification error for samples from the most represented class is much higher. As a consequence, the average classification error is higher overall. On the other hand, our choice of threshold easily tackles this problem.

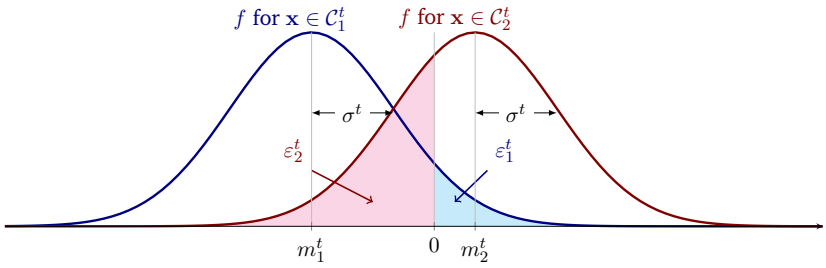

Fig. 4.6.: Schematic distribution of the score function f for samples of both classes for the experiments of Figure 4.5. The mean score m_2^t of the most represented class is closer to zero than the mean score m_1^t of the least represented class, leading to a higher classification error for samples from the most represented class.

4.4.3 Real data experiments

The purpose of this section is to check that our algorithm still performs well on real data, which does not necessarily follow Assumption 19. The data is generated with normalized VGG-features [53] of randomly BigGAN-generated images [54]. If GAN are known to produce concentrated vectors [30], the covariance is unlikely to be isotropic. However, thanks to the robustness of our method, we still have some reasonably good performances. In the following figures, only the empirical classification error is displayed, as the theoretical values are not relevant for this real data setting.

In the Figure 4.8, we consider a transfer-learning setting with 2 similar tasks (`doberman` vs `entlebucher` and `appenzeler` vs `rottweiler`, for which examples are displayed in Figure 4.7) and with classes purposely inverted between the tasks (similarly to the case $\beta = -1$ of Section 4.4.1). Naturally, the naive algorithm suffers from a huge negative transfer, while optimal labeling ensures a low error. The same tasks are used in Figure 4.9 to compare 1-task and 2-task learning (with our optimal algorithm). While the number of labeled data in the source task is constant, we add labeled data in the target task. The error of the single-task algorithm decreases, but remains higher than the multi-task algorithm, which benefits from the numerous

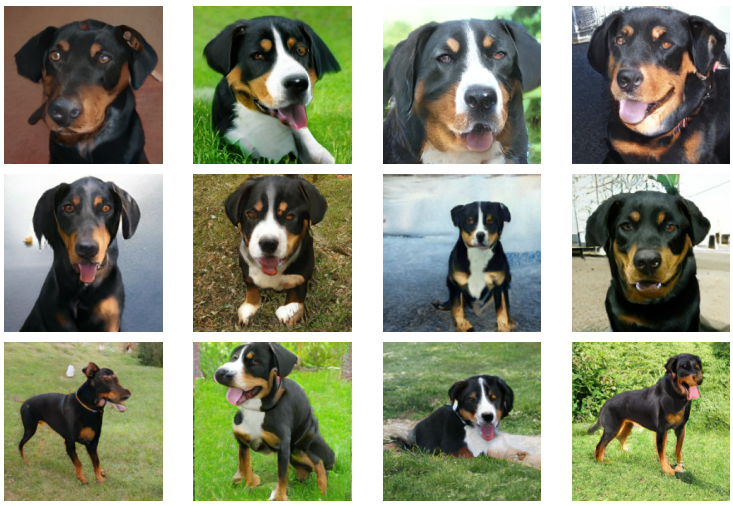

Fig. 4.7.: Examples of BigGAN-generated images used for the experiment. From left to right : *doberman* (Class $\mathcal{C}_1^1$), *entlebucher* (Class $\mathcal{C}_2^1$), *appenzeler* (Class $\mathcal{C}_1^2$) and *rottweiler* (Class $\mathcal{C}_2^2$). *entlebucher* and *appenzeler* are close, as well as *doberman* and *rottweiler*.

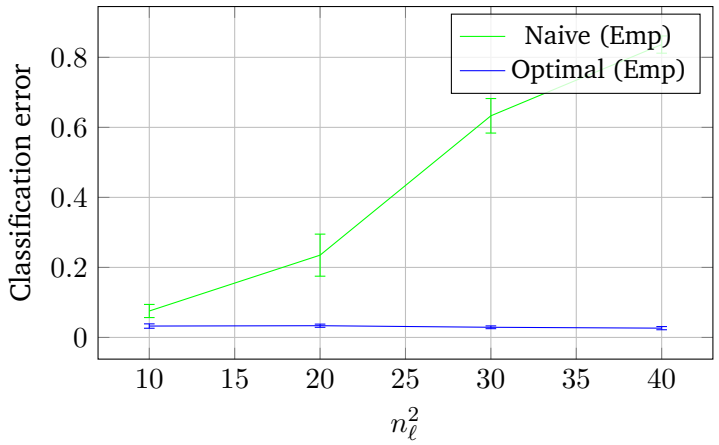

Fig. 4.8.: Empirical classification error, for both naive and optimal labels, as a function of the number of labeled data in source task, for BigGAN-generated images. (*doberman* vs *entlebucher* and *appenzeler* vs *rottweiler*, $p = 4096$, $n_\ell^1 = 20$, $n_u^1 = 200$ and $n_u^2 = 0$). The naive labels induce a huge negative transfer, while optimal labels keep the error at low level.

data of the second task. To simplify the interpretation, the fully-supervised algorithm is considered.

A third experiment is made with a semi-supervised setting, to which we compare the purely supervised method. Figure 4.10 shows the performances for the task `boxer` vs `greater swiss mountain dog` as a function of the number of unlabeled data. The error of the supervised method is logically constant, while the semi-supervised algorithm benefits from the new unlabeled samples.

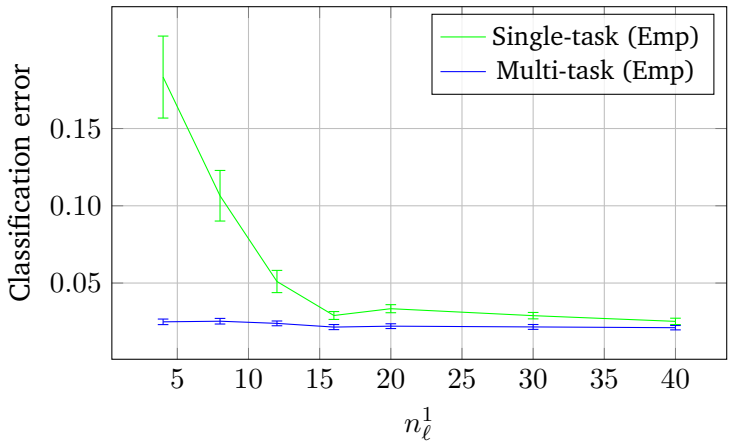

Fig. 4.9.: Empirical classification error, for 1-task and 2-task learning, as a function of the number of labeled data in target task, for BigGAN-generated images. (*doberman* vs *entlebucher* and *appenzeler* vs *rottweiler*, $p = 4096$, $n_\ell^2 = 100$, $n_u^1 = 200$ and $n_u^2 = 0$). The multi-task setting takes advantage of the labeled data from source task.

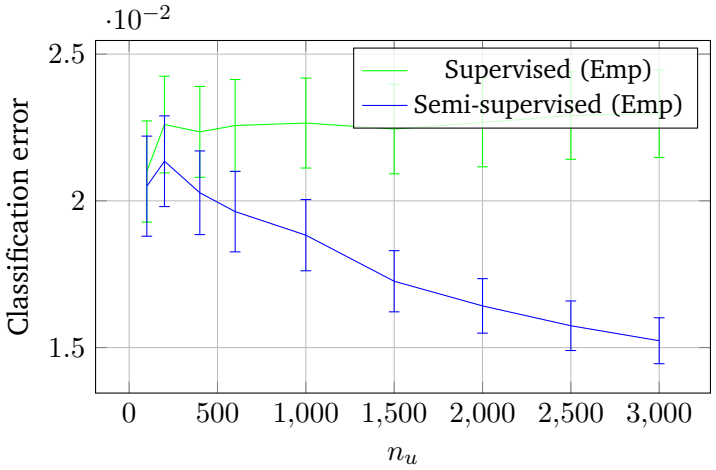

Fig. 4.10.: Empirical classification error for fully-supervised and semi-supervised algorithms, as a function of the number of unlabeled data. (BigGAN-generated images *boxer* vs *great swiss mountain dog*, $p = 4096$, $n_\ell = 10$). As expected, the error is constant with the fully-supervised algorithm, while the semi-supervised version benefits from additional unlabeled data.

In Figures 4.8 and 4.9, unlike in the experiments on synthetic data, the number of sample n is much lower than the number of features p. This is due to the fact that, from a signal-to-noise perspective, the task is much easier with our realistic datasets (even by generating really close classes such as `doberman` vs `entlebucher`) than it was with our simulations on synthetic data. In order to remain in a non-trivial setting, we have choosen a higher ratio $\frac{p}{n}$, which makes the task harder.

4.5 Concluding remarks

Thanks to its simplicity, the algorithm presented in this chapter allows an extensive mathematical analysis, which makes it at once robust, powerful and interpretable. Indeed, the comparison of the empirical error with the optimum-theoretic bound indicates that the algorithm reaches satisfying performances, and the basic assumptions made still allow it to perform well on real data. Our algorithm is therefore accessible to its users and is even able to bring them further knowledge. More precisely :

- the label vector $\tilde{y}$ informs the users on potential biases in the dataset, and helps them quantify how useful an additional task would be to solve the initial problem.
- the combined knowledge of the performance and the information-theoretic bound helps understand how difficult a problem is, and how far from optimal the associated performance lies.
- the experiments enlighten the users about the expected improvements of performance with additional labeled or unlabeled data.

5 Uncertain labeling

In Chapter 4, we performed a statistical analysis of our multi-task semi-supervised algorithm, allowing to improve the existing version of this algorithm. The data we considered were separated in two distinct categories, labeled data and unlabeled data, embracing the usual dichotomy between supervised and unsupervised algorithms. Following our idea to build a flexible and versatile classification algorithm, we would like to go beyond this strict dichotomy. Indeed, in the case of labeled data we have a perfect knowledge of the class of data samples, while in the case of unlabeled data we do not have any information at all. Such a model is very restrictive, and quite unrealistic. In the real life, labels can be controversial, biased and even erroneous. Therefore, our algorithm should be able to take into account data with uncertain labels. This would allow to model richer situations, and even to bring some new insights over the behavior of our algorithm. Thus, in this chapter, we aim to extend the results of Chapter 4 to the case of erroneous or uncertain data.

5.1 A new paradigm

Before diving into the modelization of uncertain labeling, we should present some of the cases we intend to deal with.

- Some labels can be controversial, meaning that there are labeled differently depending on the person giving this label. For example, in the case of medical data, one doctor could label a patient as healthy, while another one could label them as ill. More generaly, if the labeling is done by some experts, they might have different opinions over the same data point.

- Even without resorting to experts, some datasets are labeled with probabilities, because nobody knows with certainty to which class each data point belongs. For example, in the dataset FER+ (Facial Expression Recognition), each image has 10 different labels, corresponding to the votes of 10 crowd-sourced taggers. In the end, the dataset is labeled with a probability distribution [55].

- The labeling is sometimes simply unprecise, or subject to noise. For example, the famous dataset ImageNet is known to have a significant level of label noise, due to the fact that some samples can belong to multiple classes, while being labeled only in one class [56].

In order to model uncertain labeling, a first idea could be to give a weight to each label, this weight being higher when the labeling is done with more confidence. If there is a non-negligeable probability that a label is false, we give it a smaller weight, so that it does not jeopardize the whole algorithm if it is indeed wrongly labeled. However, this idea is not relevant in the case of controversial labeling. If two experts give two contradictory labels to the same data point, we cannot use this data point twice in our dataset. We must summarize the multiple labeling with a unique value. In our example of two contradictory labels, if these labels are respectively -1 and $+1$, the final labeling value could be 0, which is the mean between the two original labels.

Until now, the value of the label y_i^t of a given sample was either $\tilde{y}_1^t$ or $\tilde{y}_2^t$, depending wether the sample was belonging to the class $\mathcal{C}_1^t$ or $\mathcal{C}_2^t$. For any other case, one can choose to set $y_i^t = d_i^t \tilde{y}_1^t + (1 - d_i^t)\tilde{y}_2^t$, where $d_i^t \in [0, 1]$ quantifies how confidentely the sample is labeled in the class $\mathcal{C}^1$. Symmetrically, $1 - d_i^t \in [0, 1]$ quantifies how confidentely the sample is labeled in the class $\mathcal{C}^2$. We easily see that d_i^t can be interpreted as probability for the sample $\mathbf{x}_i^t$ to belong genuinely in $\mathcal{C}^1$.

Thus, let us consider now that instead of knowing which class the labeled sample belongs to, we have a couple (d_{i1}^t, d_{i2}^t) of pre-estimated probabilities that the sample belongs to a class or to the other. The case of the previous chapters is met if the couple is either $(1, 0)$ or $(0, 1)$, the associated label value being $\tilde{y}_1^t$ or $\tilde{y}_2^t$. For any other case, the value of y_i^t should be in the interval $]\tilde{y}_1^t, \tilde{y}_2^t[$. A natural choice seems to set $y_i^t = d_{i1}^t \tilde{y}_1^t + d_{i2}^t \tilde{y}_2^t$.

As a reminder, until now, unlabeled data were consider to have the label value $y_i^t = 0$. Therefore, a labeled datapoint with pre-estimated probabilities $(\frac{1}{\tilde{y}_1^t}, -\frac{1}{\tilde{y}_2^t})$ has the same label value as an unlabeled datapoint. Thus, the frontier between labeled and unlabeled data is blurred. However, we will still make a distinction between datapoints for which we want to compute an estimated class (*i.e.,* unlabeled data), and those for which we do not (*i.e.,* labeled data).

Now that we have in mind the fact that, in some cases, unlabeled data can be as informative as labeled data, there is no reason to take them into account differently in the unsupervised part of the algorithm. This justifies the choice of the solution with the relaxed fitting constraint in Chapter 3. As a reminder, this solution makes

use of all the data in the resolvent, and not only unlabeled data. This was one of the reason we chose this solution over the one with the strict fitting constraint.

5.2 Extended results

The main difference between Theorem 21 and the extended version we will state here is that labels are not necessarily equal anymore for datapoints in the same class. The equality $\forall\ \mathbf{x}_i^t \in \mathcal{C}_j^t, y_i^t = \tilde{y}_j^t$ does not hold anymore. Instead, we have $y_i^t = d_{i1}^t \tilde{y}_1^t + d_{i2}^t \tilde{y}_2^t$. To express that with a more convenient matrix formulation, we have $\mathbf{y}_\ell = \mathbf{D}\tilde{\mathbf{y}}$, with

$$\mathbf{D} = \sum_{t=1}^{T} \mathbf{E}_{tt}^{[T]} \otimes \begin{pmatrix} d_{11}^t & d_{12}^t \\ d_{21}^t & d_{22}^t \\ \vdots & \vdots \\ d_{n_\ell^t 1} & d_{n_\ell^t 2} \end{pmatrix} \tag{5.1}$$

The results of Theorem 21 have to be adapted. In particular, we have to introduce some statistics related to the couples of probabilities :

- $\bar{d}_{j_1,j_2}^t = \frac{1}{n_{\ell j_1}^t} \sum_{i' | x_{i'} \in \mathcal{C}_{j_1}^t} d_{i' j_2}^t$ the average probability for a genuine sample of class $\mathcal{C}_{j_1}^t$ to be labeled in class $\mathcal{C}_{j_1}^t$. In the ideal case, $\bar{d}_{j_1,j_2}^t = \mathbb{1}_{j_1 = j_2}$.
- $\tilde{d}_{j_1 j_2}^t = \frac{1}{n_\ell^t} \sum_{i | \mathbf{x}_i \in \mathcal{C}^t} d_{ij_1}^t d_{ij_2}^t$ which is a measure of the labeling uncertainty. If $j_1 \neq j_2$ (resp. $j_1 = j_2$), a high value of $\tilde{d}_{j_1 j_2}^t$ means a high (resp. low) uncertainty.

This leads to the following small-dimensional quantities, related to the matrix $\mathbf{D}$:

$$\bar{\mathbf{D}} = \sum_{t=1}^{T} \mathbf{E}_{tt} \otimes \begin{pmatrix} \bar{d}_{11}^t & \bar{d}_{12}^t \\ \bar{d}_{21}^t & \bar{d}_{22}^t \end{pmatrix},$$
$$\tilde{\mathbf{D}} = \sum_{t=1}^{T} \mathbf{E}_{tt} \otimes \begin{pmatrix} \tilde{d}_{11}^t & \tilde{d}_{12}^t \\ \tilde{d}_{21}^t & \tilde{d}_{22}^t \end{pmatrix}.$$

Theorem 26
Under Assumptions 19 and 20, and if labels are given by (5.1)*, for any unlabeled sample* $\mathbf{x} \in \mathcal{C}_j^t$*, and* f *being its associated score,*

$$f \to \mathcal{N}\left(m_j^t, \sigma^{t2}\right)$$

with $(1-\delta^t)m_j^t = \mathbf{a}_j^{t\intercal}\tilde{\mathbf{y}}$ *and* $(1-\delta^t)\sigma^t = \sqrt{\tilde{\mathbf{y}}^\intercal \mathbf{B}^t \tilde{\mathbf{y}}}$

$$
\begin{aligned}
\mathbf{a}_j^t &= \left(\mathbf{e}_{t,j}^{[2T]\intercal}\left(\boldsymbol{\Theta} - \frac{Tc\delta^t}{\rho^t}\boldsymbol{\Gamma}\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\right)^\intercal \\
\mathbf{B}^t &= \bar{\mathbf{D}}^\intercal\mathcal{D}_{\boldsymbol{\eta}}\left[2\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\left(\boldsymbol{\Theta} - \frac{Tc\delta^t}{\rho^t}\boldsymbol{\Gamma}\right) - \boldsymbol{\Gamma}^t\right]\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}} \\
&\quad + \bar{\mathbf{D}}^\intercal\mathcal{D}_{\boldsymbol{\eta}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\left(\boldsymbol{\Theta}\mathcal{D}_{\mathbf{r}^t}\boldsymbol{\Theta} + \bar{\boldsymbol{\Omega}}^t - \left(\frac{Tc\delta^t}{\rho^t}\right)^2\boldsymbol{\Gamma}^t\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}} \\
&\quad + \mathbf{T}^t \odot \tilde{\mathbf{D}}
\end{aligned}
$$

A proof of this theorem is displayed in Section A.5 of the appendix. Theorem 21 is a particular case of Theorem 26, with $\bar{\mathbf{D}} = \mathbf{I}_{2T}$ and $\tilde{\mathbf{D}} = \mathcal{D}_{\boldsymbol{\rho}\odot\boldsymbol{\eta}}\mathcal{D}^{-1}_{(\bar{\boldsymbol{\rho}}\odot\bar{\boldsymbol{\eta}})\otimes\mathbb{1}_2}$.

5.3 Experiments

As for now, the labeled data used in the experiments was assumed to be labeled without any mistake or imprecision. To simulate a case where labeled data suffers such imprecision, we will separate the labeled data in two categories :

- *reliable* data, labeled in a class with absolute certainty.
- *imprecise* data, for which there is a probability $r < 1$ that the data genuinely belongs to the class it has been labeled in (and therefore a probability $1 - r$ that it belongs in fact to the other class).

To simplify the setting, we assume that all the imprecise data has the same value r of reliability. n_i will denote the number of imprecise data, while n_r will denote the number of reliable data. In order to measure how useful imprecise data is to solve our problem, one could ask the following question : for a given value of n_r, if imprecise data is used instead of reliable data, how many samples n_i are needed to reach the same performance ? The first thing we notice, through Figure 5.1, is that the number of imprecise data needed to reach that performance is a linear function of the number of reliable data used in the first place.

Therefore, the ratio $\frac{n_r}{n_i}$ seems to be a relevant quantity to compute. It can be interpreted as a measure of the strength of imprecise data compared to reliable data. The higher this ratio is, the more information is brought by imprecise samples. $\frac{n_r}{n_i} = 0$ means that imprecise samples are not useful at all, while $\frac{n_r}{n_i} = 1$ means that imprecise samples are as useful as reliable samples. Figure 5.2 displays this ratio

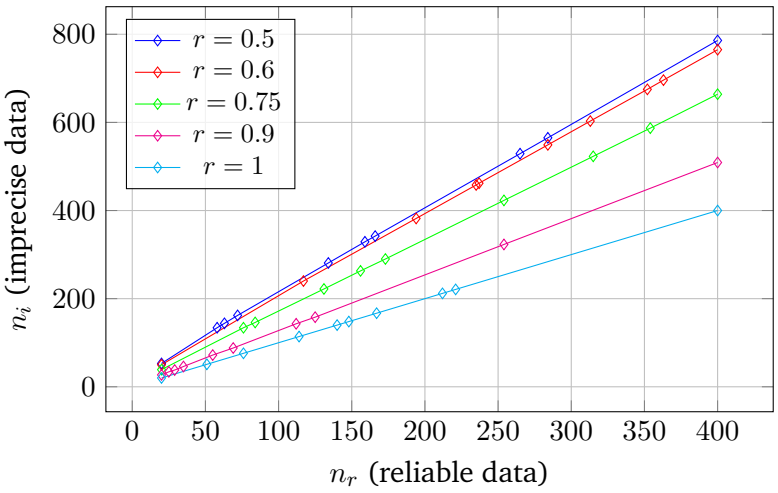

Fig. 5.1.: Number of imprecise data n_i needed to reach the same performance one had using n_r samples of reliable data, for different values of r (1-task, $p = 200$, $n_{u1} = n_{u2} = 200$). For each point, n_r is a random number between 20 and 400. The figure strongly suggests that n_i is a linear function of n_r.

for different values of reliability r and difficulty of the task (expressed through the quantity $D = \frac{1}{\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|}$).

5.4 Concluding remarks

As the algorithm built in Chapters 3 and 4 has been designed to that end, it extends naturally to the case of uncertain labeling described in this chapter. Theorem 26 is therefore in line with Theorem 21, yet with the addition of two new quantities describing the precision of the labeling. As showed in the experiments, the probabilistic labeling allows to link up labeled and unlabeled data. Beyond that, there are two main takeaways in this chapter:

- As one could expect, the more precise the labeling is, the better the algorithm will perform, but interestingly, this relation is not linear : an increase of precision is more precious when the precision is already high.
- The harder the task is, the more precious is the quality of the labeling. Unlabeled data is useful when the clusters are distinguable, but when they are too hard to separate, some labeled data is required.

In Chapter 4, the algorithm was compared to its optimal bound, computed in [18]. However, for the extended algorithm presented in this chapter, the optimal bound has not been computed yet. Without this bound, the quasi-optimality cannot be

assessed, and we lack some understanding of the algorithm's behavior. In particular, the non-linear relation between labeling precision and gain in performance could be understood better through the analysis of the optimal bound. This will be the object of the next chapter.

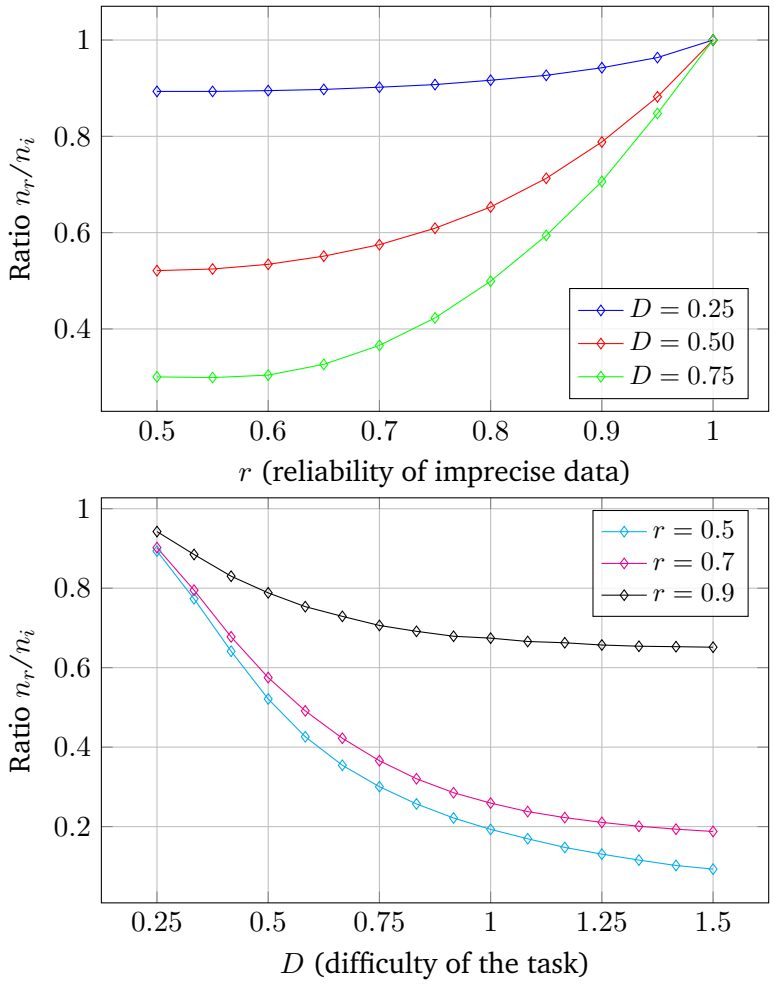

Fig. 5.2.: Ratio $\frac{n_r}{n_i}$ for different values of reliability (r) and difficulty of the task ($D = 1/\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|$). The higher the ratio is, the more effective is the contribution of imprecise samples to the task. Both figures show that the harder the task is, the least useful imprecise samples are. However, an increasing of r leads to significantely better results. (**Top**) Ratio $\frac{n_r}{n_i}$ as a function of the reliability of imprecise data, for different values of difficulty. (**Bottom**) Ratio $\frac{n_r}{n_i}$ as a function of the difficulty of the task, for different values of reliability.

6 Asymptotic Bayes risk

In Chapter 5, we presented how our algorithm can be extended in order to deal with uncertain labels. However, unlike the algorithm of Chapter 4, we do not have any optimal bound to compare it to, as the optimal bound has not been computed yet, to the best of our knowledge. The main goal of this chapter is therefore to derive the optimal bound in the case of uncertain labeling.

Such an optimal bound is called the Bayes risk, which is the minimal achievable probability of misclassification for a new data point. As said in Chapter 1, some Gaussian mixtures model, as the one used in this manuscript, have been analyzed with tools from statistical physics to derive the Bayes risk of a given problem, meaning that any possible algorithm cannot reach a better performance [17, 18].

This optimal bound can then bring further insight of the behavior of the algorithm, and lead to additional improvements. Therefore, the objectives of this chapter are twofolds :

- Compute the Bayes risk in the case of uncertain data labeling, inspired by the work of [18].

- Use the knowledge of this optimal bound to further understand the behavior of the algorithm from Chapter 4, which performances have been proven to be close to the optimal bound.

For simplicity reasons, the model presented in this chapter is a single-task model, but it is worth to note that most of the conclusions remain true in a multi-task setting, as the previous works it is based on are multi-task models ([18] and previous chapters).

The remainder of the chapter is organized as follows. Section 6.1 introduces the model, the assumptions and the aim of the next sections. Section 6.2 states our main theorem, and gives interpretation of this theorem. Section 6.3 gives a succinct proof of the main theorem. Finally, Section 6.4 displays simulations of both the optimal bound and the algorithm presented in Chapters 4 and 5.

6.1 Model and Main Objective

As in the previous chapter, we consider a semi-supervised binary classification task with training samples $\mathbf{X} = [\mathbf{X}_\ell, \mathbf{X}_u] \in \mathbb{R}^{p\times n}$ which consists of a set of n_ℓ labeled data samples $\mathbf{X}_\ell = \{\mathbf{x}_i\}_{i=1}^{n_\ell}$ and a set of n_u unlabeled data points $\mathbf{X}_u = \{\mathbf{x}_i\}_{i=n_\ell+1}^{n}$. Each labeled data point $\mathbf{x}_i$ has an associated couple (d_{i1}, d_{i2}) of *pre-estimated* probabilities that the vector belongs to one class or the other, such that $d_{i1} + d_{i2} = 1$. The goal of the classification task is to predict the genuine class of unlabeled data $\mathbf{X}_u$. In this context, we are interested in computing the Bayes risk of the classification task, *i.e.,* the minimal classification error achievable for each unlabeled sample $\mathbf{x}_i$ with the available data :

$$\inf_{\hat{y}_i} \mathbb{P}(\hat{y}_i \neq y_i)$$

where $\hat{y}_i = \mathbb{E}\left[y_i|\mathbf{X}\right]$ is the label prediction made for the sample $\mathbf{x}_i$.

Assumption 27 (On the data distribution)
The columns of the data matrix $\mathbf{X}$ *are independent Gaussian random variables. Specifically, the data samples* $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ *are i.i.d. observations such that* $\mathbf{x}_i \in \mathcal{C}_j \Leftrightarrow \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_j, \mathbf{I}_p)$ *where* $\mathcal{C}_j^t$ *denotes the Class* j*. We assume that the number of data in each class is the same. We further define the quantity* $\lambda = \frac{1}{4}\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|^2$*, which is called the* signal to noise ratio *(SNR).*

We study our model in a large dimensional setting, where the dimension and the amount of data have the same order of magnitude, which is practically the case with modern data.

Assumption 28 (Growth Rate)
As $n \to \infty$ *:*

- $p/n \to c > 0$
- $n_\ell/n \to \eta$

With our notations, in a single-task setting, and assuming that the probability couples of labeled data are either $(0, 1)$ or $(1, 0)$ (*i.e.,* the data is labeled with complete certainty), it has been proved in [18] that under the previous assumptions, as $p \to \infty$, the Bayes risk converges to

$$\mathcal{Q}(\sqrt{q_u}), \tag{6.1}$$

where $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} \, \mathrm{d}u$, and the couple (q_u, q_v) satisfies the following equations:

$$q_u = \lambda \frac{\frac{\lambda q_v}{c}}{1 + \frac{\lambda q_v}{c}} \tag{6.2}$$

$$q_v = \eta + (1 - \eta) F(q_u), \tag{6.3}$$

with $F(q) = \mathbb{E}\left[\tanh(\sqrt{q}\xi + q)\right]$, $\xi \sim \mathcal{N}(0, 1)$.

Our goal in the remainder of this article is to derive an equivalent result in the more general case where data is not labeled with certainty. Let us define, for each datapoint $\mathbf{x}_i$, $\varepsilon_i = d_{i2} - d_{i1} \in [-1, 1]$. This quantity is enough to charaterize the couple of probabilities, as $d_{i1} + d_{i2} = 1$. We observe that $|\varepsilon| = 1$ means that the data is labeled with certainty in a class, while $\varepsilon = 0$ means that the data is unlabeled.

To get equation (6.3), it is needed to compute the quantity $\hat{y}_i = \mathbb{E}\left[y_i | \mathbf{X}\right]$, which is the estimation of y_i with the available data $\mathbf{X}$. In the proof (presented in Section 6.3), a trick allows to compute this quantity as a function of q_u, and the labeling information is expressed through the prior distribution of y.

- For data labeled with certainty, it is known that $y_i = -1$ or $+1$, so the prior is either $\delta(t + 1)$ or $\delta(t - 1)$.
- For unlabeled data, the prior is uniform over $\{-1, +1\}$. Or equivalently, the distribution function is
$$\frac{1}{2}\delta(t + 1) + \frac{1}{2}\delta(t - 1).$$
- When the data is not labeled with certainty but with a probability couple (d_{i1}, d_{i2}), the prior distribution of y_i becomes
$$d_{i1}\delta(t + 1) + d_{i2}\delta(t - 1)$$

The following section states our main theorem using this last prior distribution.

6.2 Main Results

Theorem 29

Under the previous assumptions, as $p \to \infty$,

- *The Bayes risk converges to*

$$\mathcal{Q}(\sqrt{q_u}),$$

where $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} \, \mathrm{d}u$.

- *The overlaps* q_u, q_v *satisfy the following equations*

$$q_u = \lambda \frac{\frac{\lambda q_v}{c}}{1 + \frac{\lambda q_v}{c}} \tag{6.4}$$

$$q_v = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} F_{\varepsilon_i}(q_u) \tag{6.5}$$

with $F_\varepsilon(q) = \mathbb{E}\left[\psi_\varepsilon(q + \sqrt{q}\xi)\right]$, $\xi \sim \mathcal{N}(0,1)$ *and*

$$\psi_\varepsilon(t) = \frac{tanh(t) + \varepsilon^2 \left(1 - tanh(t) - tanh^2(t)\right)}{1 - \varepsilon^2 tanh^2(t)}.$$

A sketch of the proof of Theorem 29 is given in Section 6.3. The function F_ε is similar to the previous function F, but the expression of ψ_ε is not easy to understand as it is.

Remark 30
The function ψ_ε *can be put in the following (more convenient) form :*

$$\begin{aligned}\psi_\varepsilon(t) = {} & tanh(t) + \varepsilon^2 \left(1 - tanh(t)\right) \\ & - (1 - \varepsilon^2)(1 - tanh(t)) \sum_{k\geq 1} \varepsilon^{2k} tanh^{2k}(t).\end{aligned}$$

Remark 31
The function F_ε *can be approximated by :*

$$\tilde{F}_\varepsilon(q) = \mathbb{E}\left[\tilde{\psi}_\varepsilon(q + \sqrt{q}\xi)\right],$$

with $\xi \sim \mathcal{N}(0,1)$ *and*

$$\tilde{\psi}_\varepsilon(t) = tanh(t) + \varepsilon^2(1 - tanh(t)).$$

Figure 6.1 gives an idea of the quality of the approximation made in Remark 31. However, it is worth to note that its purpose is not to replace the original formula from Theorem 29, as that formula is already tractable and can be computed easily. Instead, Remark 31 intend to bring a simpler formula that conveys an understanding of the key role of quantity ε^2.

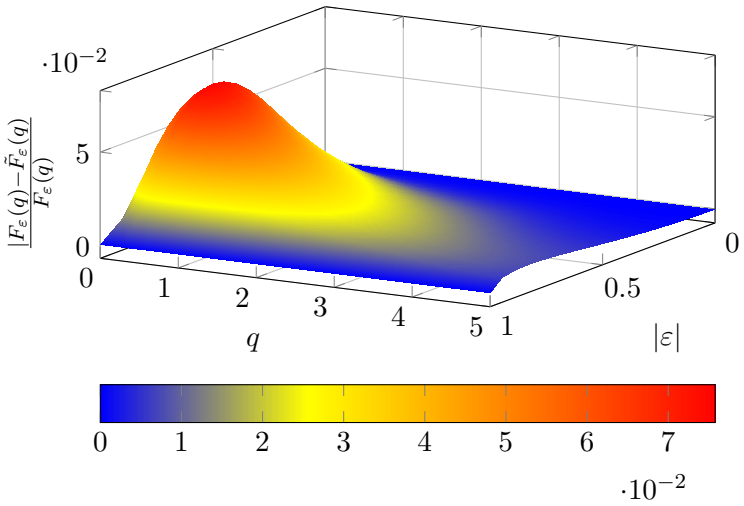

Fig. 6.1.: Relative error of the approximation $\tilde{F}_\varepsilon(q) \simeq F_\varepsilon(q)$. The error is at most 7%, and shrinks for either $\varepsilon = 0$, $\varepsilon = 1$ or large q.

As q_u and q_v are related to each other, one could also worry that a small error in equation (6.5) could lead to a completely different solution for q_u and q_v. Fortunately, if one replaces the solution $(q_u^\star, q_v^\star)$ of the system by another solution $(q_u^\star + \Delta q_u, q_v^\star + \Delta q_v)$, then we have $|\frac{\Delta q_u}{q_u^\star}| \leq |\frac{\Delta q_v}{q_v^\star}|$. This means that a small variation of q_v leads to an even smaller variation of q_u (see the details in Section A.6 of the appendix).

Corollary 32

With the previous approximation of the function F_ε, one can approximate the equation (6.5) *:*

$$q_v \simeq \bar{\varepsilon}^2 + (1 - \bar{\varepsilon}^2) F(q_u) \tag{6.6}$$

with

$$\begin{aligned} \bar{\varepsilon}^2 &= \frac{1}{n} \sum_{i=1} {\varepsilon_i}^2 \\ F(q) &= \mathbb{E}\left[tanh(q + \sqrt{q}\xi)\right] \\ \xi &\sim \mathcal{N}(0,1) \end{aligned}$$

Proof: The function $\tilde{F}_{\varepsilon_i}$ described in Remark 31 can be expressed with F :

$$\begin{aligned} \tilde{F}_{\varepsilon_i}(q) &= \mathbb{E}\left[\tanh(q + \sqrt{q}\xi) + {\varepsilon_i}^2 (1 - \tanh(q + \sqrt{q}\xi))\right] \\ &= {\varepsilon_i}^2 + \mathbb{E}\left[\tanh(q + \sqrt{q}\xi)\right](1 - {\varepsilon_i}^2) \\ &= {\varepsilon_i}^2 + (1 - {\varepsilon_i}^2) F(q) \end{aligned}$$

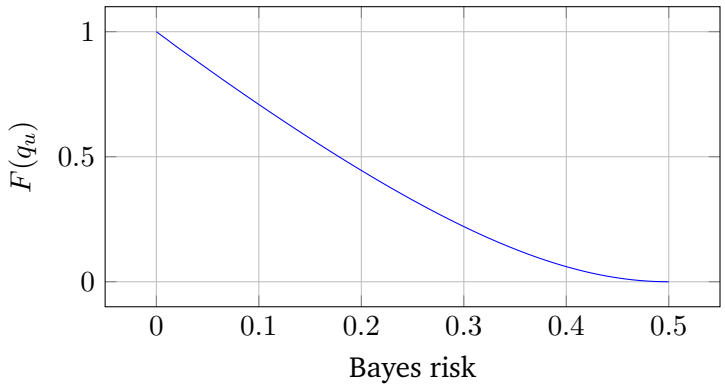

Fig. 6.2.: Usefulness of unlabeled data as a function of the Bayes risk of the task. Interestingly, the only criterion to determinate the effectiveness of unlabeled data is how solvable the task is. The lower the Bayes risk is, the more unlabeled data are useful to perform the task.

By mixing the results of Theorem 29 and Remark 31, one gets asymptotically

$$\begin{aligned} q_v &\simeq \frac{1}{n}\sum_{i=1}^{n} \tilde{F}_{\varepsilon_i}(q_u) \\ &= \frac{1}{n}\sum_{i=1}^{n} \left[{\varepsilon_i}^2 + (1-{\varepsilon_i}^2)F(q)\right] \\ &= \bar{\varepsilon}^2 + \left(1-\bar{\varepsilon}^2\right)F(q) \end{aligned}$$

Corollary 32 enables an easy interpretation of Theorem 29. Indeed, the value of q_v given by equation (6.6) is similar to equation (6.3), with $\eta = \bar{\varepsilon}^2$. One can check that :

- $\varepsilon^2 = 0 \leftrightarrow$ unlabeled data $\leftrightarrow \eta = 0$
- $\varepsilon^2 = 1 \leftrightarrow$ data labeled with certainty $\leftrightarrow \eta = 1$

If all samples are labeled with the same value ε, then it is equivalent to a task for which one would have a proportion ε^2 of data labeled with certainty and a proportion $1-\varepsilon^2$ of unlabeled data.

To go further, $F(q_u)$ can be understood as a quantity that expresses how useful unlabeled data are, relatively to labeled data. Indeed, if $F(q_u) = 1$, unlabeled data brings as much information as labeled data. Interestingly, this quantity $F(q_u)$ only depends on q_u, which itself related to the Bayes risk of the classification task. Thus, usefulness of unlabeled data only depends on how well the task can be performed. Figure 6.2 displays the quantity $F(q_u)$ as a function of Bayes risk.

6.3 Sketch of the proof

The proof of Theorem 29 is really similar to the one performed in [18], as (6.1) and (6.2) remain the same, but the main difference lays in the expression of q_v, that must be adapted. As in the original proof, $q_v = \langle \hat{\mathbf{y}}, \mathbf{y} \rangle$ is the *overlap* of the signal $\frac{1}{\sqrt{n}}\mathbf{y}|\mathbf{X}$, where $\mathbf{y} = (y_i)_i$, and we have asymptotically, through the law of large numbers :

$$q_v = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}\left[\hat{y}_i y_i\right] \tag{6.7}$$

where $\hat{y}_i = \mathbb{E}\left[y_i|\mathbf{X}\right]$ is the MMSE estimator of y_i.

Then, the following lemma, which is a corollary of Propositions 16 and 17 from Chapter 2, is one of the key of our proof.

Lemma 33
Estimating y_i from $\mathbf{X}$ is asymptotically equivalent to estimating the signal y_i from the output of a Gaussian channel with SNR q_u, where q_u the overlap of $\mu|\mathbf{X}$ is given by (6.4). *Let us consider the following Gaussian channel*

$$u_i = \sqrt{q} y_i + \xi_i,$$

with $q = q_u$ the SNR, y_i the signal and $\xi_i \sim \mathcal{N}(0,1)$. Then computing the overlap $\mathbb{E}\left[\hat{y}_i y_i\right]$ of y_i can be done equivalently with $\hat{y}_i = \mathbb{E}\left[y_i|\mathbf{X}\right]$ or with $\hat{y}_i = \mathbb{E}\left[y_i|u_i\right]$.

Thus, to compute the overlap $\mathbb{E}\left[\hat{y}_i y_i\right]$, one only needs to compute $\hat{y}_i = \mathbb{E}\left[y_i|u_i\right]$, which is easier to work with than $\mathbb{E}\left[y_i|\mathbf{X}\right]$. The signal y_i follows the distribution

$$y_i \sim \begin{cases} -1 \text{ with probability } d_{i1}, \\ +1 \text{ with probability } d_{i2}. \end{cases}$$

Therefore,

$$\mathbf{E}\left[y_i|u_i\right] = \frac{d_{i1}e^{\sqrt{q}u_i} - d_{i2}e^{-\sqrt{q}u_i}}{d_{i1}e^{\sqrt{q}u_i} + d_{i2}e^{-\sqrt{q}u_i}}.$$

Using $\varepsilon_i = d_{i1} - d_{i2}$, one gets $\mathbf{E}\left[y_i|u_i\right] = f_{\varepsilon_i}(\sqrt{q}u_i)$, with

$$f_\varepsilon(t) = \frac{\tanh(t) + \varepsilon}{1 + \varepsilon\tanh(t)}.$$

$$
\begin{aligned}
&\mathbf{E}\left[y_i \mathbf{E}\left[y_i|u_i\right]\right] = \mathbf{E}\left[y_i f_{\varepsilon_i}(qy_i + \sqrt{q}\xi_i)\right] \\
&= d_{i1}\mathbf{E}\left[f_{\varepsilon_i}(q + \sqrt{q}\xi_i)\right] - d_{i2}\mathbf{E}\left[f_{\varepsilon_i}(-q + \sqrt{q}\xi_i)\right] \\
&= d_{i1}\mathbf{E}\left[f_{\varepsilon_i}(q + \sqrt{q}\xi_i)\right] - d_{i2}\mathbf{E}\left[f_{\varepsilon_i}(-(q + \sqrt{q}\xi_i))\right] \\
&= \mathbf{E}\left[\psi_{\varepsilon_i}(q + \sqrt{q}\xi_i)\right],
\end{aligned}
$$

with $\psi_{\varepsilon_i}(t) = d_{i1} f_{\varepsilon_i}(t) - d_{i2} f_{\varepsilon_i}(-t)$.

More precisely,

$$
\begin{aligned}
\psi_{\varepsilon_i}(t) &= d_{i1}\frac{\tanh(t) + \varepsilon_i}{1 + \varepsilon_i \tanh(t)} - d_{i2}\frac{-\tanh(t) + \varepsilon_i}{1 - \varepsilon_i \tanh(t)} \\
&= (d_{i1} + d_{i2})\frac{\tanh(t) - {\varepsilon_i}^2 \tanh(t)}{1 - {\varepsilon_i}^2 \tanh^2(t)} \\
&+ (d_{i1} - d_{i2})\frac{\varepsilon_i(1 - \tanh^2(t))}{1 - {\varepsilon_i}^2 \tanh^2(t)} \\
&= \frac{\tanh(t) + {\varepsilon_i}^2\left(1 - \tanh(t) - \tanh^2(t)\right)}{1 - {\varepsilon_i}^2 \tanh^2(t)}.
\end{aligned}
$$

Therefore,

$$
\mathbf{E}\left[y_i \hat{y}_i\right] = \mathbf{E}\left[\psi_{\varepsilon_i}(q_u + \sqrt{q_u}\xi_i)\right], \tag{6.8}
$$

with

$$
\psi_{\varepsilon_i}(t) = \frac{\tanh(t) + {\varepsilon_i}^2\left(1 - \tanh(t) - \tanh^2(t)\right)}{1 - {\varepsilon_i}^2 \tanh^2(t)}.
$$

Combining (6.7) and (6.8), we obtain (6.5).

6.4 Simulations and Applications

The objective of this section is to confront theoretical results of Section 6.2 and the algorithm described in Chapters 4 and 5, which will be from now on refered to as *optimal algorithm*. The common idea of the following experiments is that the optimal algorithm is expected to behave similarly to the optimal bound studied in Section 6.2.

As we have seen in Section 6.2, different labeling settings can lead to the same value of $\bar{\varepsilon}^2$. Let us start with only unlabeled data and data labeled with certainty. Then,

$$\bar{\varepsilon}^2 = \eta, \tag{6.9}$$

and we obtain a classification error E. Now, let us assume that all the labeled data is labeled with the same confidence $\kappa < 1$. The total number of data n stays unchanged. For different values of κ, if one wants to achieve the same error E, then more labeled data will be needed as κ decreases. Indeed, reaching the same performance means obtaining the same q_u, and therefore q_v, as the task does not change beyond that. We recall that $q_v \simeq \bar{\varepsilon}^2 + (1 - \bar{\varepsilon}^2)F(q_u)$, and in our context, $F(q_u)$ does not change. Consequently, to obtain the same error E, $\bar{\varepsilon}^2$ must stay constant. Moreover, we have

$$\bar{\varepsilon}^2 = \frac{n_\ell}{n}(2\kappa - 1)^2 \tag{6.10}$$

By combining (6.9) and (6.10), one gets

$$n_\ell = \frac{\eta}{(2\kappa - 1)^2} n \tag{6.11}$$

For a given value of η, $(2\kappa - 1)^2$ must be no smaller than η, otherwise even a fully labeled dataset would not allow to reach $\bar{\varepsilon}^2 = \eta$.

Figure 6.3 displays both empirical and theoretical values of n_ℓ as a function of κ in this setting. The theoretical value is given by (6.11), and the empirical one is computed by incrementing n_ℓ (and decrementing n_u) until the error given by the optimal algorithm gets below E. The match between empirical and theoretical curves show that Corollary 32 helps to understand the behavior of the algorithm.

The other takeaway message of Section 6.2 is that the usefulness of unlabeled data, expressed through the quantity $F(q_u)$, only depends on the Bayes risk of the task, and therefore the final classification error of the optimal algorithm. In order to understand the contribution of unlabeled data, one could be interested in computing the reduction of the classification error by using the semi-supervised version of the optimal algorithm instead of the fully supervised one. With this aim in mind, we will consider the two following quantities:

- The absolute error reduction

$$\frac{E_{\text{sup}} - E_{\text{semi-sup}}}{E_{\text{sup}}}$$

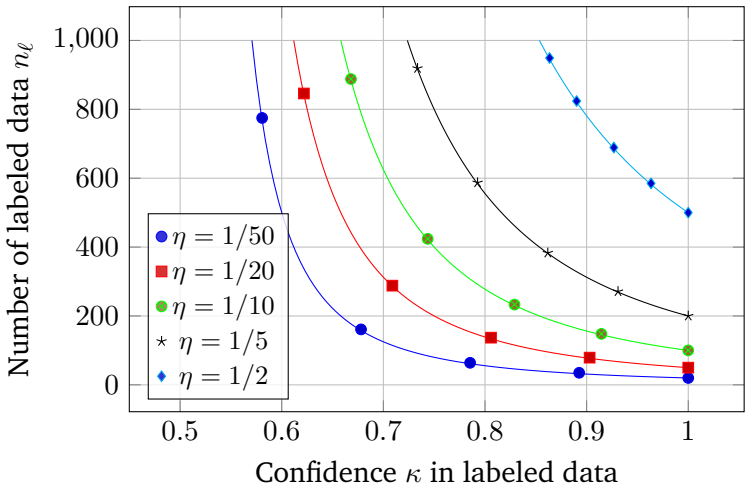

Fig. 6.3.: Number of labeled data n_ℓ needed to perform the same performance, as a function of the confidence in the data labeling, for different values of η ($n = 1000, p = 200, \lambda = 0.25$). The empirical values are displayed in dots, and theoretical prediction (built on the results of Section 6.2) in plain line. The least reliable the data is, the more data is needed to reach the same performance.

which is the tangible error reduction one can expect by adopting the semi-supervised method instead of the fully supervised one.

- The error reduction relatively to oracle bayes risk

$$\frac{E_{\text{sup}} - E_{\text{semi-sup}}}{E_{\text{sup}} - E_{\text{oracle}}}$$

which reflects how much of the way to oracle error has been done by adopting the semi-supervised method instead of the fully supervised one,

where E_{oracle} is the bayes risk one can expect when the centers of distributions $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are known. More precisely, oracle error is given by the formula

$$E_{\text{oracle}} = \mathcal{Q}(\sqrt{\lambda}).$$

Leaving out the parameter η, the main parameters that drive the final error are λ and c, as we can see in (6.4). Therefore, Figures 6.4 and 6.5 display the two kinds of error reduction presented above, respectively as functions of λ and c.

In Figure 6.4, it is clear that the error reduction is higher when λ grows, for both types of error reduction. Intuitively, a higher SNR means a lower final error, and consequently a higher contribution of unlabeled data to the classification.

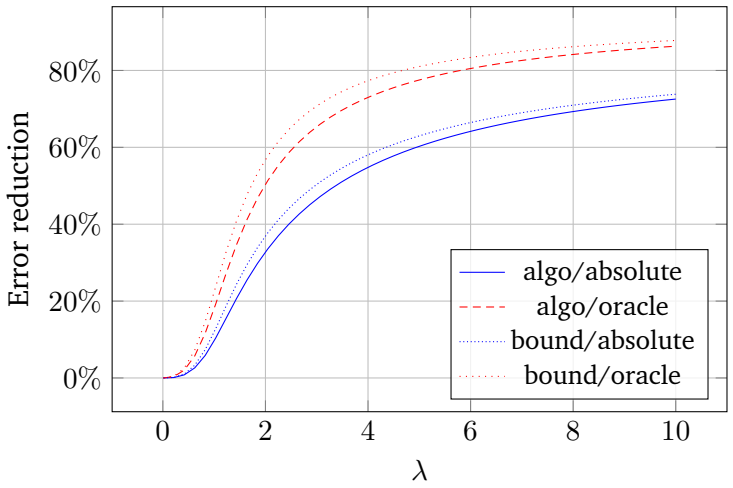

Fig. 6.4.: Percentage of error reduction by using the semi-supervised algorithm instead of the supervised one, as a function of the SNR λ ($n = p = 200, \eta = 0.2$). The easier the task is, the higher the semi-supervised contribution is, because the classification error is lower.

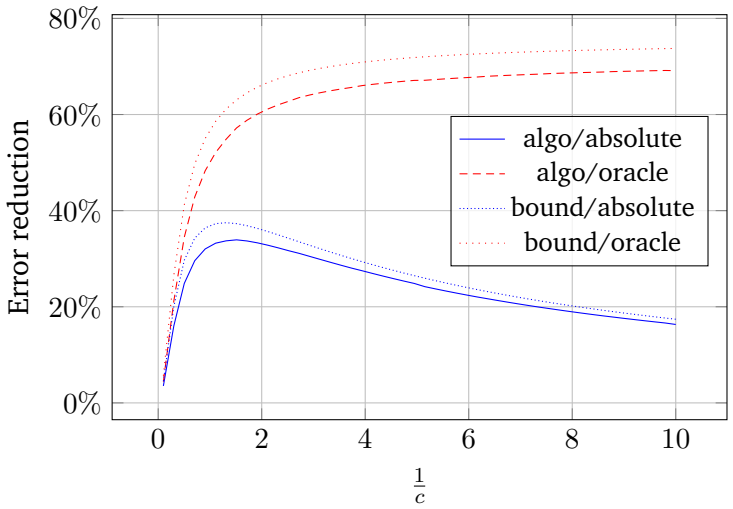

Fig. 6.5.: Percentage of error reduction by using the semi-supervised algorithm instead of the supervised one, as a function of the ratio $1/c = n/p$ ($\lambda = 2, p = 200, \eta = 0.2$). As $1/c$ grows, the semi-supervised algorithm is more and more effective comparatively to the supervised one, relatively to oracle error, because the classification error is lower and oracle error stays constant. However, if the oracle error ceases to be the reference, then the contribution of semi-supervised decreases for high values of $1/c$, because both algorithms edge closer to the oracle bound, which stays far from zero.

Meanwhile in Figure 6.5, the two types of error reduction do not behave similarly. In this case, the oracle error is constant, and both errors E_{sup} and $E_{\text{semi-sup}}$ get close to E_{oracle} as $\frac{1}{c}$ grows. Therefore, there is not much to gain by adopting the semi-supervised algorithm, as we are already close to the oracle bound with the

supervised one. However, the error reduction relatively to oracle still increases when $\frac{1}{c}$ grows. We see that the understanding of what remains to be improved plays a key role in Figure 6.5.

6.5 Concluding remarks

Figuring out the link between the performances of an algorithm and its optimal bound gives precious insight. In our case, the bound only depends on few parameters. By manipuling these parameters, we see that the performances of the algorithm stay close to the bound. This suggests that the algorithm is near optimal, or at least that it does not require additional complexification in order to be significantly improved. Therefore, if the algorithm gives poor performances while being close to the bound, it simply means that the problem is inherently too hard to solve. Thus, the interest of computing such optimal bounds is clear. By knowing in advance how far from optimal an algorithm is, one can avoid spending too much energy to solve a problem which turns out to be a dead-end.

Furthermore, the similarity of behavior between the algorithm and the bound allows to understand the algorithm from an other perspective. Indeed, results from Sections 6.2 and 6.4 provide a new understanding on when semi-supervised learning is truly useful, and when it is not.

7 Conclusion and perspectives

We proposed and analyzed a simple yet powerful learning scheme which combines semi-supervised and multi-task learning in the same framework. Our method is able to tackle the classical problem of negative transfer, is robust to data size imbalances, is able to deal with probabilistic labels... The key to the approach is the use of modern large dimensional random matrix results, which allows for optimal hyperparameter setting with absolutely no step of cross-validation required. This simple, intuitive, versatile, yet fully mathematically supported approach opens a new avenue of research in cost-efficient and technically tractable tools for "flexible machine learning".

The field of ML mostly relies on the availability of large amounts of data and computer power. Its success is an engineering achievement before being a scientific one. Indeed, most modern models are untractable by design, and succeed in their tasks for the same reason they will never be decipherable. Aside from this, the theoretical understanding of the learning process in ML algorithms is poor, and reduced to uninsightful results. At the opposite, we considered in this thesis a simple model, with a mere gaussian mixture and a linear classifier. Our model, which could be critized for being simplistic or unrealistic, offers on the other hand an insightful framework to work with.

Therefore, because we place ourselves aside from the current trend of ML, our method does not intend to compete with state-of-the-art algorithms, but rather guarantees sufficient performances. In accordance with the base "convivality" principles enunciated in the introduction, future investigations should not aim at any excessive complexification and improvement of the present method. Rather, we should seek for further simplification and accessibility: the random matrix framework remains indeed quite opaque to most, which still constitutes an accessibility hurdle we should try to overtake. Options lie in exploiting alternative tools from the mathematical, and even preferrably from the statistical physics, litterature in order to retrieve our technical results from a more direct and more intuitive path (which for instance the non-rigorous but much simpler replica methods may offer).

Our semi-supervised scheme, already extended to the case of uncertain labeling, could be further extended to the field of active learning. It consists in choosing

wisely the set of unlabeled data which should be labeled prioritarily in order to bring more information to the learning scheme [57]. It is motivated by the same paradigm as semi-supervised learning, which is that in most applications, unlabeled data is much easier to collect than labeled data. It relates to the idea mentionned in this manuscript that every data sample does not bring as much information, depending on the learning context. However, a hurdle to this approach is that our assumption of identical distribution for every sample would not hold if we were to consider a specific subset of unlabeled data.

The multi-task setting presented here could also be extended to a "multi-task learning on the edge" paradigm, which main idea is to keep computations locally with minimal information exchanges rather than centralizing the data in a single computing server. This kind of algorithm has already been studied in a fully supervised case [58], but what of the semi-supervised case ?

All of this aside, in our quest of convivial AI algorithms, the main pitfall would be to apply some principles superficially, without questionning in depth the field of AI. As such, there are two main existing direction of research that seem to act positively regarding the current and forthcoming socio-environmental crisis, but with mitigated results:

- A large part of the research consists in designing more efficent algorithms in order to decrease the global energy and computer power consumption. However, improvements in efficiency lead to the fact that more of the improved algorithms will be used, causing at the end of the day more energy and computer power consumption. This well-documented principle is known as the rebound effect [59]. Any approach that consists in designing low-consuming algorithms will enventually lead to a higher global energy and computer power consumption, which is the exact opposite of its claimed objective.
- A rather new field of research consists in assessing the environmental impact of a given AI algorithm [60, 61]. While being a useful tool to raise awareness among the people, it is often used to lighten up the global impact of AI, for instance by comparing it to another sector of activity with an even higher environmental impact. This phenomenon, known as "whataboutism", is a recognize factor of political inertness regarding climate change [62].

To avoid these pitfalls, we should stick to the notion of sufficient algorithms, which only match their purpose, and do not create new artificial needs.

Bibliography

[1] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608014002135

[2] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The Computational Limits of Deep Learning," arXiv, Tech. Rep. arXiv:2007.05558, Jul. 2022, arXiv:2007.05558 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/2007.05558

[3] D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.-M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, "Carbon Emissions and Large Neural Network Training."

[4] B. Perrigo, "The $2 per hour workers who made chatgpt safer," *TIME*, Jan. 2023. [Online]. Available: https://time.com/6247678/openai-chatgpt-kenya-workers/

[5] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0893608089900208

[6] I. Illich, *Tools for Conviviality*. Harper & Row, 1973.

[7] C. Louart, Z. Liao, and R. Couillet, "A random matrix approach to neural networks," *The Annals of Applied Probability*, vol. 28, no. 2, Apr. 2018. [Online]. Available: https://projecteuclid.org/journals/annals-of-applied-probability/volume-28/issue-2/A-random-matrix-approach-to-neural-networks/10.1214/17-AAP1328.full

[8] A. W. v. d. Vaart, *Asymptotic Statistics*, ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1998.

[9] E. P. Wigner, "On the distribution of the roots of certain symmetric matrices," *Annals of Mathematics*, vol. 67, p. 325, 1958. [Online]. Available: https://api.semanticscholar.org/CorpusID:123682463

[10] V. A. Marchenko and L. A. Pastur, “Distribution of eigenvalues for some sets of random matrices,” *Mathematics of The Ussr-sbornik*, vol. 1, pp. 457–483, 1967. [Online]. Available: https://api.semanticscholar.org/CorpusID:31589842

[11] B. Scholkopf and A. Smola, *Learning with Kernels: support vector machines, regularization, optimization, and beyond*, 2001.

[12] R. Couillet and Z. Liao, *Random Matrix Methods for Machine Learning*. Cambridge University Press, 2022.

[13] J. Sohl-Dickstein, “The boundary of neural network trainability is fractal,” 2024. [Online]. Available: https://arxiv.org/abs/2402.06184

[14] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[15] T. Lesieur, C. D. Bacco, J. E. Banks, F. Krzakala, C. Moore, and L. Zdeborová, “Phase transitions and optimal algorithms in high-dimensional gaussian mixture clustering,” *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 601–608, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:14737624

[16] M. Lelarge and L. Miolane, “Fundamental limits of symmetric low-rank matrix estimation,” *Probability Theory and Related Fields*, vol. 173, no. 3, pp. 859–929, Apr. 2019. [Online]. Available: https://doi.org/10.1007/s00440-018-0845-x

[17] ——, “Asymptotic bayes risk for gaussian mixture in a semi-supervised setting,” 2019.

[18] M.-T. Nguyen and R. Couillet, “Asymptotic bayes risk of semi-supervised multi-task learning on gaussian mixture,” 2023.

[19] M. E. Tipping and C. M. Bishop, “Probabilistic Principal Component Analysis,” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 61, no. 3, pp. 611–622, 01 2002. [Online]. Available: https://doi.org/10.1111/1467-9868.00196

[20] N. G. Polson and S. L. Scott, “Data augmentation for support vector machines,” *Bayesian Analysis*, vol. 6, pp. 1–23, 2011. [Online]. Available: https://api.semanticscholar.org/CorpusID:16001780

[21] C. Bishop, “Bayesian PCA,” in *Advances in Neural Information Processing Systems*, M. Kearns, S. Solla, and D. Cohn, Eds., vol. 11. MIT Press, 1998. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/1998/file/c88d8d0a6097754525e02c2246d8d27f-Paper.pdf

[22] F. Krzakala, M. Mézard, F. Sausset, Y. F. Sun, and L. Zdeborová, “Statistical-physics-based reconstruction in compressed sensing,” *Phys. Rev. X*, vol. 2, p. 021005, May 2012. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.2.021005

[23] Z. Bai and J. Silverstein, *Spectral Analysis of Large Dimensional Random Matrices*, 01 2010.

[24] E. Titchmarsh, *The Theory of Functions*. Oxford University Press, 1939. [Online]. Available: https://books.google.fr/books?id=g2K4AAAAIAAJ

[25] N. Akhiezer and I. Glazman, *Theory of Linear Operators in Hilbert Space*, ser. Dover Books on Mathematics. Dover Publications, 1993. [Online]. Available: https://books.google.fr/books?id=GTWMqiuvOAQC

[26] P. L. C. Heinz H. Bauschke, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer Cham, 2017.

[27] M. Ledoux, *The concentration of measure phenomenon*. American Mathematical Society, 2001.

[28] N. El Karoui, “Concentration of measure and spectra of random matrices: Applications to correlation matrices, elliptical distributions and beyond,” *The Annals of Applied Probability*, vol. 19, no. 6, Dec. 2009. [Online]. Available: http://dx.doi.org/10.1214/08-AAP548

[29] C. Louart and R. Couillet, “Concentration of Measure and Large Random Matrices with an application to Sample Covariance Matrices,” Feb. 2019, working paper or preprint. [Online]. Available: https://hal.science/hal-02020287

[30] M. E. A. Seddik, C. Louart, M. Tamaazousti, and R. Couillet, “Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures,” in *International Conference on Machine Learning*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:210868271

[31] R. Caruana, S. Baluja, and T. Mitchell, “Using the Future to "Sort Out" the Present: Rankprop and Multitask Learning for Medical Risk Evaluation.”

[32] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, and A. Galstyan, "Multitask learning and benchmarking with clinical time series data," *Scientific Data*, vol. 6, no. 1, Jun. 2019. [Online]. Available: http://dx.doi.org/10.1038/s41597-019-0103-9

[33] S. Ruder, "An overview of multi-task learning in deep neural networks," Tech. Rep., 06 2017.

[34] B. R. Paredes, A. Argyriou, N. Berthouze, and M. Pontil, "Exploiting unrelated tasks in multi-task learning," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, N. D. Lawrence and M. Girolami, Eds., vol. 22. La Palma, Canary Islands: PMLR, 21–23 Apr 2012, pp. 951–959. [Online]. Available: https://proceedings.mlr.press/v22/romera12.html

[35] M. Tiomoko, R. Couillet, and H. Tiomoko, "Large dimensional analysis and improvement of multi task learning," *arXiv preprint arXiv:2009.01591*, 2020.

[36] M. Tiomoko, H. T. Ali, and R. Couillet, "Deciphering and Optimizing Multi-Task Learning: a Random Matrix Approach," Sep. 2020. [Online]. Available: https://openreview.net/forum?id=Cri3xz59ga

[37] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. The MIT Press, 09 2006. [Online]. Available: https://doi.org/10.7551/mitpress/9780262033589.001.0001

[38] B. Shahshahani and D. Landgrebe, "The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 5, pp. 1087–1095, 1994.

[39] F. G. Cozman and I. Cohen, "Risks of semi-supervised learning: How unlabeled data can degrade performance of generative classifiers," in *Semi-Supervised Learning*, 2006. [Online]. Available: https://api.semanticscholar.org/CorpusID:63547716

[40] S. Ben-David, T. Lu, and D. Pál, "Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning," in *Annual Conference Computational Learning Theory*, 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:7670149

[41] X. Mai and R. Couillet, "A random matrix analysis and improvement of semi-supervised learning for large dimensional data," *The Journal of Machine Learning Research*, vol. 19, no. 1, pp. 3074–3100, 2018.

[42] ——, "Consistent semi-supervised graph regularization for high dimensional data," *Journal of Machine Learning Research*, vol. 22, no. 94, pp. 1–48, 2021. [Online]. Available: http://jmlr.org/papers/v22/19-081.html

[43] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems 16 (NIPS 2003)*, 2003.

[44] K. Avrachenkov, A. Mishenin, P. Gonçalves, and M. Sokol, "Generalized Optimization Framework for Graph-based Semi-supervised Learning," in *Proceedings of the 2012 SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, Apr. 2012, pp. 966–974.

[45] T. Joachims, "Transductive Learning via Spectral Graph Partitioning," 2003.

[46] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions," 2003.

[47] B. Ghojogh and M. Crowley, "Unsupervised and supervised principal component analysis: Tutorial," 2022.

[48] M. Tiomoko, R. Couillet, and F. Pascal, "PCA-based multi-task learning: a random matrix approach," in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 23–29 Jul 2023, pp. 34 280–34 300. [Online]. Available: https://proceedings.mlr.press/v202/tiomoko23a.html

[49] I. M. Johnstone, "On the distribution of the largest eigenvalue in principal components analysis," *The Annals of Statistics*, vol. 29, no. 2, pp. 295–327, 2001. [Online]. Available: http://www.jstor.org/stable/2674106

[50] D. Pejic and M. Arsic, *Minimization and Maximization of Functions: Golden-Section Search in One Dimension*. Cham: Springer International Publishing, 2019, pp. 55–90. [Online]. Available: https://doi.org/10.1007/978-3-030-13803-5_3

[51] C. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, NY, 2006.

[52] A. Rocha and S. K. Goldenstein, “Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 289–302, 2014.

[53] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2015.

[54] A. Brock, J. Donahue, and K. Simonyan, “Large scale gan training for high fidelity natural image synthesis,” 2019.

[55] E. Barsoum, C. Zhang, C. Canton-Ferrer, and Z. Zhang, “Training deep networks for facial expression recognition with crowd-sourced label distribution,” *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, 2016.

[56] S. Yun, S. J. Oh, B. Heo, D. Han, J. Choe, and S. Chun, “Re-labeling imagenet: from single to multi-labels, from global to localized labels,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2340–2350.

[57] B. Settles, “Active learning literature survey,” 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:324600

[58] S. Fakhry, R. Couillet, and M. Tiomoko, “Multi-task learning on the edge: cost-efficiency and theoretical optimality,” 2021. [Online]. Available: https://arxiv.org/abs/2110.04639

[59] A. Druckman, M. Chitnis, S. Sorrell, and T. Jackson, “Missing carbon reductions? exploring rebound and backfire effects in uk households,” *Energy Policy*, vol. 39, no. 6, pp. 3572–3581, Jun. 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301421511002473

[60] A.-L. Ligozat, J. Lefèvre, A. Bugeau, and J. Combaz, “Unraveling the hidden environmental impacts of ai solutions for environment,” 2022. [Online]. Available: https://arxiv.org/abs/2110.11822

[61] A. Berthelot, E. Caron, M. Jay, and L. Lefèvre, “Estimating the environmental impact of generative-ai services using an lca-based methodology,” *Procedia CIRP*, vol. 122, pp. 707–712, 2024, 31st CIRP Conference on Life Cycle Engineering. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827124001173

[62] W. F. Lamb, G. Mattioli, S. Levi, J. T. Roberts, S. Capstick, F. Creutzig, J. C. Minx, F. Müller-Hansen, T. Culhane, and J. K. Steinberger, "Discourses of climate delay," *Global Sustainability*, vol. 3, p. e17, 2020.

[63] A. Kammoun, M. Kharouf, W. Hachem, and J. Najim, "A central limit theorem for the sinr at the lmmse estimator output for large-dimensional signals," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5048–5063, 2009.

List of Figures

Appendix

A

A.1 Solution of the optimization problem

First of all, we need to clarify why (3.5) is equivalent to (3.6).

$$
\begin{aligned}
&\sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} \hat{\omega}_{ii'}^{tt'} \left(f_i^t - f_{i'}^{t'} \right)^2 \\
&= \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} \left[\hat{\omega}_{ii'}^{tt'} \left((f_i^t)^2 - (f_{i'}^{t'})^2 \right) - 2 f_i^t \hat{\omega}_{ii'}^{tt'} f_{i'}^{t'} \right] \\
&= \sum_{t,t'=1}^{T} \Lambda^{tt'} \left(\sum_{i=1}^{n^t} (f_i^t)^2 \underbrace{\sum_{i'=1}^{n^{t'}} \hat{\omega}_{ii'}^{tt'}}_{=0} + \sum_{i'=1}^{n^{t'}} (f_{i'}^{t'})^2 \underbrace{\sum_{i=1}^{n^t} \hat{\omega}_{ii'}^{tt'}}_{=0} \right) - 2 \sum_{t,t'=1}^{T} \Lambda^{tt'} \sum_{i=1}^{n^t} \sum_{i'=1}^{n^{t'}} f_i^t \hat{\omega}_{ii'}^{tt'} f_{i'}^{t'} \\
&= -2 \sum_{t,t'=1}^{T} \Lambda^{tt'} \mathbf{f}^{t\intercal} \hat{\mathbf{W}}^{tt'} \mathbf{f}^{t'} = -2 \mathbf{f}^{\intercal} \hat{\mathbf{W}} \mathbf{f}
\end{aligned}
$$

Therefore, under the condition $\mathbf{f}_\ell = \mathbf{y}_\ell$

$$
(3.5) \Leftrightarrow \min_{\mathbf{f}^1, \ldots, \mathbf{f}^T} -2 \mathbf{f}^{\intercal} \hat{\mathbf{W}} \mathbf{f} + 2 \alpha_u \| \mathbf{f}_u \|^2 \Leftrightarrow (3.6)
$$

Similarly, equation (3.7) and equation (3.11) are equivalent.

A.1.1 Strict fitting constraint

The optimization problem we need to solve is

$$
\min_{\mathbf{f} \in \mathbb{R}^n} \alpha \| \mathbf{f}_u \|^2 - \mathbf{f}^{\intercal} \tilde{\mathbf{W}} \mathbf{f} \Leftrightarrow \min_{\mathbf{f}_u \in \mathbb{R}^{n_u}} \mathbf{f}_u{}^{\intercal} \left(\alpha \mathbf{I}_{n_u} - \tilde{\mathbf{W}}_{uu} \right) \mathbf{f}_u - 2 \mathbf{f}_u{}^{\intercal} \tilde{\mathbf{W}}_{u\ell} \mathbf{y}_\ell
$$

The problem is convex as long as $\alpha > \|\tilde{\mathbf{W}}_{uu}\|$, ensuring $\alpha\mathbf{I}_{n_u} - \tilde{\mathbf{W}}_{uu}$ to be positive definite. We assume that this condition is satisfied. The quantity to minimize is a quadratic form of $\mathbf{f}$, so the solution of the optimization problem satisfies:

$$\min_{\mathbf{f}\in\mathbb{R}^n} \alpha\|\mathbf{f}_u\|^2 - \mathbf{f}^\intercal\tilde{\mathbf{W}}\mathbf{f} \Leftrightarrow \left(\alpha\mathbf{I}_{n_u} - \tilde{\mathbf{W}}_{uu}\right)\mathbf{f}_u - \tilde{\mathbf{W}}_{u\ell}\mathbf{y}_\ell = 0$$
$$\Leftrightarrow \mathbf{f}_u = \left(\mathbf{I}_{n_u} - \frac{\tilde{\mathbf{W}}_{uu}}{\alpha}\right)^{-1}\frac{\tilde{\mathbf{W}}_{u\ell}}{\alpha}\mathbf{y}_\ell \Leftrightarrow \mathbf{f}_u = \left(\mathbf{I}_{n_u} - \frac{\mathbf{Z}_u{}^\intercal\mathbf{A}\mathbf{Z}_u}{Tp}\right)^{-1}\frac{\mathbf{Z}_u{}^\intercal\mathbf{A}\mathbf{Z}_\ell}{Tp}\mathbf{y}_\ell$$

Indeed,

$$\frac{\tilde{\mathbf{W}}}{\alpha} = \sum_{t,t'=1}^{T}\tilde{\Lambda}^{tt'}\mathbf{E}_{tt'}\otimes\hat{\mathbf{W}}^{tt'} = \frac{1}{Tp}\sum_{t,t'=1}^{T}\tilde{\Lambda}^{tt'}\mathbf{E}_{tt'}\otimes(\mathbf{X}^{t\intercal}\mathbf{X}^{t'})$$
$$= \frac{1}{Tp}\sum_{t,t'=1}^{T}(\mathbf{E}_{tt}\otimes\mathbf{X}^t)^\intercal(\tilde{\mathbf{\Lambda}}\otimes\mathbf{I}_p)(\mathbf{E}_{t't'}\otimes\mathbf{X}^{t'}) = \frac{\mathbf{Z}^\intercal\mathbf{A}\mathbf{Z}}{Tp}$$

A.1.2 Relaxed fitting constraint

The optimization problem we need to solve is

$$\min_{\mathbf{f}\in\mathbb{R}^n} \alpha\|\mathbf{f}_\ell - \mathbf{y}_\ell\|^2 + \alpha\|\mathbf{f}_u\|^2 - \mathbf{f}^\intercal\tilde{\mathbf{W}}\mathbf{f}$$

By using the convention $\mathbf{y}_u = 0$, we can merge the first two terms:

$$\min_{\mathbf{f}\in\mathbb{R}^n} \alpha\|\mathbf{f} - \mathbf{y}\|^2 - \mathbf{f}^\intercal\tilde{\mathbf{W}}\mathbf{f} \Leftrightarrow \min_{\mathbf{f}\in\mathbb{R}^n} \mathbf{f}^\intercal\left(\alpha\mathbf{I}_n - \tilde{\mathbf{W}}\right)\mathbf{f} - 2\alpha\mathbf{f}^\intercal\mathbf{y}$$

The problem is convex as long as $\alpha > \|\tilde{\mathbf{W}}\|$, ensuring $\alpha\mathbf{I}_n - \tilde{\mathbf{W}}$ to be positive definite. We assume that this condition is satisfied. The quantity to minimize is a quadratic form of $\mathbf{f}$, so the solution of the optimization problem satisfies:

$$\min_{\mathbf{f}\in\mathbb{R}^n} \alpha\|\mathbf{f} - \mathbf{y}\|^2 - \mathbf{f}^\intercal\tilde{\mathbf{W}}\mathbf{f} \Leftrightarrow \left(\alpha\mathbf{I}_n - \tilde{\mathbf{W}}\right)\mathbf{f} - \alpha\mathbf{y} = 0$$
$$\Leftrightarrow \mathbf{f} = \left(\mathbf{I}_n - \frac{\tilde{\mathbf{W}}}{\alpha}\right)^{-1}\mathbf{y} \Leftrightarrow \mathbf{f} = \left(\mathbf{I}_n - \frac{\mathbf{Z}^\intercal\mathbf{A}\mathbf{Z}}{Tp}\right)^{-1}\mathbf{y}.$$

Indeed,

$$\frac{\tilde{\mathbf{W}}}{\alpha} = \sum_{t,t'=1}^{T} \tilde{\Lambda}^{tt'} \mathbf{E}_{tt'} \otimes \hat{\mathbf{W}}^{tt'} = \frac{1}{Tp} \sum_{t,t'=1}^{T} \tilde{\Lambda}^{tt'} \mathbf{E}_{tt'} \otimes (\mathbf{X}^{t\intercal} \mathbf{X}^{t'})$$
$$= \frac{1}{Tp} \sum_{t,t'=1}^{T} (\mathbf{E}_{tt} \otimes \mathbf{X}^{t})^{\intercal} (\tilde{\mathbf{\Lambda}} \otimes \mathbf{I}_p)(\mathbf{E}_{t't'} \otimes \mathbf{X}^{t'}) = \frac{\tilde{\mathbf{Z}}^{\intercal} \tilde{\mathbf{Z}}}{Tp},$$

where $\tilde{\mathbf{Z}} = \mathbf{A}^{\frac{1}{2}} \mathbf{Z}$.

If $\alpha_\ell \neq \alpha_u$, the solution is instead:

$$\left(\mathbf{M} - \tilde{\mathbf{W}}\right) \mathbf{f} - \mathbf{M}\mathbf{y} = 0$$
$$\Leftrightarrow \mathbf{f} = \left(\mathbf{M} - \tilde{\mathbf{W}}\right)^{-1} \mathbf{M}\mathbf{y} \Leftrightarrow \mathbf{f} = \left(\mathbf{M} - \frac{\tilde{\mathbf{Z}}^{\intercal} \tilde{\mathbf{Z}}}{Tp}\right)^{-1} \mathbf{M}\mathbf{y}$$
$$\Leftrightarrow \mathbf{f} = \mathbf{y} + \frac{1}{Tp} \mathbf{M}^{-1} \tilde{\mathbf{Z}}^{\intercal} \underbrace{\left(\mathbf{I}_{Tp} - \frac{\tilde{\mathbf{Z}} \mathbf{M} \tilde{\mathbf{Z}}^{\intercal}}{Tp}\right)^{-1}}_{=\mathbf{Q}} \tilde{\mathbf{Z}}\mathbf{y},$$

where

$$\mathbf{M} = \begin{pmatrix} \alpha_\ell \mathbf{I}_{n_\ell} & \mathbf{0} \\ \mathbf{0} & \alpha_u \mathbf{I}_{n_u} \end{pmatrix} \qquad \text{and} \qquad \mathbf{A} = \mathbf{\Lambda} \otimes \mathbf{I}_p.$$

Once restricted to unlabeled data, we have

$$\mathbf{f}_u = \frac{1}{Tp} \frac{1}{\alpha_u} \mathbf{Z}_u^{\intercal} \mathbf{A}^{\frac{1}{2}} \mathbf{Q} \mathbf{A}^{\frac{1}{2}} \mathbf{Z}_\ell \mathbf{y}_\ell, \tag{A.1}$$

so we understand that there is only one difference with equation (3.14), which turns out to be in the resolvant $\mathbf{Q}$. In this resolvant, the matrix $\mathbf{M}$ acts like a weight filter over samples : labeled samples are considered with a weight α_ℓ, while unlabeled samples are considered with a weight α_u. However, the resolvant corresponds to the unsupervised part of the algorithm. As a consequence, labeled and unlabeled samples should not be considered differently as it comes to $\mathbf{Q}$. Thus, we make the choice $\alpha_\ell = \alpha_u = \alpha$.

A.2 Proof of Proposition 24

$\zeta^t \in [m_1^t, m_2^t]$ so there exists $\lambda \in [0, 1]$ such that:

$$\zeta^t = \lambda m_1^t + (1 - \lambda) m_2^t$$

$$\varepsilon_1^t = \mathcal{Q}\left(\frac{\zeta^t - m_1^t}{\sigma^t}\right) = \mathcal{Q}\left((1-\lambda)\frac{m_2^t - m_1^t}{\sigma^t}\right)$$
$$\varepsilon_2^t = \mathcal{Q}\left(\frac{m_2^t - \zeta^t}{\sigma^t}\right) = \mathcal{Q}\left(\lambda\frac{m_2^t - m_1^t}{\sigma^t}\right)$$

$\mathcal{Q}$ is a decreasing function, so in order to minimize ε_1^t and ε_2^t, one needs to maximize $\frac{m_2^t - m_1^t}{\sigma^t}$, or equivalently the following quantity:

$$\left(\frac{m_2^t - m_1^t}{\sigma^t}\right)^2 = \frac{(\tilde{\mathbf{y}}^\intercal(\mathbf{a}_2^t - \mathbf{a}_1^t))^2}{\tilde{\mathbf{y}}^\intercal \mathbf{B}^t \tilde{\mathbf{y}}} \tag{A.2}$$

Then the optimal scores is the solution of

$$\arg\max_{\tilde{\mathbf{y}}} \frac{(\tilde{\mathbf{y}}^\intercal(\mathbf{a}_2^t - \mathbf{a}_1^t))^2}{\tilde{\mathbf{y}}^\intercal \mathbf{B}^t \tilde{\mathbf{y}}}.$$

To solve this maximization problem, we need the following lemma.

Lemma 34
Let $\mathbf{B} \in \mathbb{R}^{m\times m}$ an invertible positive matrix and $\mathbf{a} \in \mathbb{R}^m$. Then:

$$\arg\max_{\mathbf{v}} \frac{(\mathbf{v}^\intercal \mathbf{a})^2}{\mathbf{v}^\intercal \mathbf{B} \mathbf{v}} = \{\lambda \mathbf{B}^{-1}\mathbf{a}, \forall \lambda \in \mathbb{R}\}.$$

Proof: With the change of variable $\mathbf{u} = \mathbf{B}^{\frac{1}{2}}\mathbf{v}$, the problem can be written as

$$\arg\max_{\mathbf{u}} \frac{(\mathbf{u}^\intercal \mathbf{B}^{-\frac{1}{2}}\mathbf{a})^2}{\mathbf{u}^\intercal \mathbf{u}}.$$

If $\mathbf{u}$ is a solution to this problem, then $\lambda\mathbf{u}$ is also a solution, for any $\lambda \in \mathbb{R}$. It is therefore sufficient to find a solution $\mathbf{u}$ such that $\|\mathbf{u}\| = \|\mathbf{B}^{-\frac{1}{2}}\mathbf{a}\|$.

$$\arg\max_{\substack{\mathbf{u} \\ \text{s.t.}\|\mathbf{u}\| = \|\mathbf{B}^{-\frac{1}{2}}\mathbf{a}\|}} \frac{(\mathbf{u}^\intercal \mathbf{B}^{-\frac{1}{2}}\mathbf{a})^2}{\mathbf{u}^\intercal \mathbf{u}} = \arg\max_{\substack{\mathbf{u} \\ \text{s.t.}\|\mathbf{u}\| = \|\mathbf{B}^{-\frac{1}{2}}\mathbf{a}\|}} (\mathbf{u}^\intercal \mathbf{B}^{-\frac{1}{2}}\mathbf{a})^2 = \pm \mathbf{B}^{-\frac{1}{2}}\mathbf{a}$$

Through the change of variable, and using the fact that $\lambda\mathbf{u}$ is also a solution, we have $\mathbf{v} = \lambda \mathbf{B}^{-1}\mathbf{a}, \forall \lambda \in \mathbb{R}$.

Thanks to this lemma, the optimal scores are (up to a multiplicative constant)

$$\arg\max_{\tilde{\mathbf{y}}} \frac{(\tilde{\mathbf{y}}^\intercal(\mathbf{a}_2^t - \mathbf{a}_1^t))^2}{\tilde{\mathbf{y}}^\intercal \mathbf{B}^t \tilde{\mathbf{y}}} = (\mathbf{B}^t)^{-1}(\mathbf{a}_2^t - \mathbf{a}_1^t).$$

Note that to ensure the positiveness of $\frac{m_2^t - m_1^t}{\sigma^t}$, the multiplicative constant must be positive.

A.3 Proof of Proposition 25

To minimize $\varepsilon^t = \frac{\varepsilon_1^t + \varepsilon_2^t}{2}$, the optimal threshold is $\zeta^t = \frac{m_1^t + m_2^t}{2}$.

$$\varepsilon^t = \frac{1}{2}\mathcal{Q}\left(\frac{\zeta^t - m_1^t}{\sigma^t}\right) + \frac{1}{2}\mathcal{Q}\left(\frac{m_2^t - \zeta^t}{\sigma^t}\right) = \mathcal{Q}\left(\frac{m_2^t - m_1^t}{2\sigma^t}\right)$$

As stated in Proposition 24, the minimum value of ε^t is achieved with $\tilde{\mathbf{y}}^\star = (\mathbf{B}^t)^{-1}(a_2^t - a_1^t)$, which leads to

$$\begin{aligned} m_2^t - m_1^t &= (a_2^t - a_1^t)^\intercal \mathbf{B}^t)^{-1}(a_2^t - a_1^t) \\ \sigma^t &= \sqrt{(a_2^t - a_1^t)^\intercal (\mathbf{B}^t)^{-1}(a_2^t - a_1^t)} \end{aligned}$$

Finally, we have

$$\varepsilon_\star^t = \mathcal{Q}\left(\frac{1}{2}\sqrt{(a_2^t - a_1^t)^\intercal (\mathbf{B}^t)^{-1}(a_2^t - a_1^t)}\right)$$

A.4 Estimation of useful quantities

A.4.1 Estimation of the data matrix

To estimate the data matrix $\boldsymbol{\mathcal{M}} = \mathbf{M}^\intercal\mathbf{M}$, one only needs to estimate quantities such as $\theta = \boldsymbol{\mu}_1^\intercal \boldsymbol{\mu}_2$. To estimate these quantities, we use the following unbiaised estimator:

$$\hat{\theta} = \frac{1}{n_1 n_2} \mathbb{1}_{n_1}^\intercal \mathbf{X}_1^\intercal \mathbf{X}_2 \mathbb{1}_{n_2}$$

where $(\mathbf{X}_j)_{\cdot,i} \sim \mathcal{N}(\boldsymbol{\mu}_j, \mathbf{I}_p)$

$$\hat{\theta} = \frac{1}{n_1 n_2} \mathbb{1}_{n_1}^\intercal \left(\mathbb{1}_{n_1}\boldsymbol{\mu}_1^\intercal + \mathbf{V}_1^\intercal\right)\left(\boldsymbol{\mu}_2 \mathbb{1}_{n_2}^\intercal + \mathbf{V}_2\right)\mathbb{1}_{n_2}$$

where $(\mathbf{V}_j)_{\cdot,i} \sim \mathcal{N}(0, \mathbf{I}_p)$
To compute the law of $\hat{\theta}$, we will use the two following facts:

Lemma 35

$$\frac{1}{\sqrt{n_j}}\mathbf{V}_j \mathbb{1}_{n_j} \sim \mathcal{N}(0, \mathbf{I}_p)$$

Lemma 36

If $\mathbf{u}, \mathbf{v} \sim \mathcal{N}(0, \mathbf{I}_p)$ *and are independent, then*

$$\frac{1}{\sqrt{p}}\mathbf{u}^\intercal \mathbf{v} \sim \mathcal{N}(0, 1)$$

$$\begin{aligned}
\hat{\theta} &= \frac{1}{n_1 n_2} \mathbb{1}_{n_1}^\intercal \left(\mathbb{1}_{n_1}\boldsymbol{\mu}_1^\intercal + \mathbf{V}_1^\intercal\right)\left(\boldsymbol{\mu}_2 \mathbb{1}_{n_2}^\intercal + \mathbf{V}_2\right)\mathbb{1}_{n_2} \\
&= \left(\boldsymbol{\mu}_1 + \frac{1}{n_1}\mathbf{V}_1 \mathbb{1}_{n_1}\right)^\intercal \left(\boldsymbol{\mu}_2 + \frac{1}{n_2}\mathbf{V}_2 \mathbb{1}_{n_2}\right) \\
&= \left(\boldsymbol{\mu}_1 + \frac{1}{\sqrt{n_1}}\mathbf{u}_1\right)^\intercal \left(\boldsymbol{\mu}_2 + \frac{1}{\sqrt{n_2}}\mathbf{u}_2\right) \\
&\text{with} \quad \mathbf{u}_1, \mathbf{u}_2 \sim \mathcal{N}(0, \mathbf{I}_p) \\
&= \boldsymbol{\mu}_1^\intercal \boldsymbol{\mu}_2 + \frac{1}{\sqrt{n_2}}\boldsymbol{\mu}_1^\intercal \mathbf{u}_2 + \frac{1}{\sqrt{n_1}}\mathbf{u}_1^\intercal \boldsymbol{\mu}_2 + \frac{1}{\sqrt{n_1 n_2}}\mathbf{u}_1^\intercal \mathbf{u}_2 \\
&= \boldsymbol{\mu}_1^\intercal \boldsymbol{\mu}_2 + \frac{\|\boldsymbol{\mu}_1\|}{\sqrt{n_2}} z_1 + \frac{\|\boldsymbol{\mu}_2\|}{\sqrt{n_1}} z_2 + \sqrt{\frac{p}{n_1 n_2}} z_3 \\
&\text{with} \quad z_1, z_2, z_3 \sim \mathcal{N}(0, 1).
\end{aligned}$$

Therefore the estimator is consistent, and converges with an $\mathcal{O}\left(\frac{1}{\sqrt{n}}\right)$ speed, under Assumption 20. For the specific case of $\theta = \boldsymbol{\mu}_1^\intercal \boldsymbol{\mu}_1$, one can use the previous results, but needs to divide the samples in two separate subsets, in order to keep the property of independence between the samples.

A.4.2 Estimation of the hyperparameter matrix

We recall that the hyperparameter matrix $\boldsymbol{\Lambda}$ is Similarly, the quantity $\langle \boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t, \boldsymbol{\mu}_1^{t'} - \boldsymbol{\mu}_2^{t'} \rangle$ from equation (4.5) can be estimated by:

$$\hat{\theta} = \left(\frac{1}{n_1^t}\mathbf{X}_1^t \mathbb{1}_{n_1^t} - \frac{1}{n_2^t}\mathbf{X}_2^t \mathbb{1}_{n_2^t}\right)^\intercal \left(\frac{1}{n_1^{t'}}\mathbf{X}_1^{t'} \mathbb{1}_{n_1^{t'}} - \frac{1}{n_2^{t'}}\mathbf{X}_2^t \mathbb{1}_{n_2^{t'}}\right)$$

To estimate the normalizing term $\|\boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t\| = \sqrt{(\boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t)^\intercal(\boldsymbol{\mu}_1^t - \boldsymbol{\mu}_2^t)}$, one should once again divide the samples in two separate subsets to keep the independence between the samples.

A.5 Proof of Theorems 21 and 26

Theorem 21 is a particular case of Theorem 26, with $\bar{\mathbf{D}} = \mathbf{I}_{2T}$ and $\tilde{\mathbf{D}} = \mathcal{D}_{\boldsymbol{\rho}\odot\boldsymbol{\eta}}\mathcal{D}^{-1}_{(\bar{\boldsymbol{\rho}}\odot\bar{\boldsymbol{\eta}})\otimes\mathbb{1}_2}$. Therefore, we only need to prove Theorem 26.

We recall that

$$\mathbf{f}_u = \frac{1}{Tp}\mathbf{Z}_u^\intercal\mathbf{A}^{\frac{1}{2}}\underbrace{\left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1}}_{=\mathbf{Q}}\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell\mathbf{y}_\ell,$$

so the score f_i of an unlabeled sample is

$$f_i = \frac{1}{Tp}\mathbf{z}_i^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{Q}\mathbf{A}^{\frac{1}{2}}\mathbf{Z}_\ell\mathbf{y}_\ell = \frac{1}{Tp}\sum_{i'}\mathbf{z}_i^\intercal\mathbf{A}^{\frac{1}{2}}\mathbf{Q}\mathbf{A}^{\frac{1}{2}}\mathbf{z}_{i'}y_{i'}. \tag{A.3}$$

The convergence in distribution of the statistics of the classification score f is identical to the central limit theorem derived in [63], where the authors proved the Gaussian distribution of objects of the form $\mathbf{x}^\intercal(\mathbf{X}\mathbf{X}^\intercal + \theta\mathbf{I}_p)^1\mathbf{x}$, even for covariances which are more complex than in our case. Our output scores clearly fall under the scope of these results. With the same arguments, any linear combination of the scores converges to a normal distribution. Thus, according to Cramér–Wold theorem, $\mathbf{f}_u$ is a Gaussian vector. As long as the score vector is a Gaussian vector, the proof boils down to compute the mean and variance of every score, and ensure the pairwise uncorrelation of the scores. To compute the means and variances of scores, we will make use of the notion of deterministic equivalent [12]. A deterministic equivalent $\bar{\mathbf{Q}} \in \mathbb{R}^{m\times m}$ of a given random matrix $\mathbf{Q} \in \mathbb{R}^{m\times m}$ is defined by the fact that, for deterministic matrices $\mathbf{B} \in \mathbb{R}^{m\times m}$ and vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$ of unit norms (operator and Euclidean, respectively), we have, as $m \to \infty$:

- $\frac{1}{m}\operatorname{Tr}\mathbf{B}(\mathbf{Q} - \bar{\mathbf{Q}}) \xrightarrow{a.s.} 0$
- $\mathbf{u}^\intercal(\mathbf{Q} - \bar{\mathbf{Q}})\mathbf{v} \xrightarrow{a.s.} 0$

The notation $\mathbf{Q} \leftrightarrow \bar{\mathbf{Q}}$ denotes that $\bar{\mathbf{Q}}$ is a deterministic equivalent of $\mathbf{Q}$.

The proof is organized as follows:

- A first order deterministic equivalent (Section A.5.1) is needed to compute the mean of f_i (Section A.5.2)
- A second order deterministic equivalent (Section A.5.3) is needed to compute the variance of f_i (Section A.5.4)
- The correlation between two distinct scores is computed, proving uncorrelatedness between each pair of scores (Section A.5.5).

In the following proof, we will need to distinguish the data matrix $\mathbf{Z}$ and its centered version $\mathring{\mathbf{Z}}$, as the centering breaks the independance between samples. This effect of centering is sumed up in the following lemma.

Lemma 37
For a data matrix $\mathbf{X}$ defined as in Assumption 2, and $\mathring{\mathbf{X}} = \mathbf{X}^t\mathbf{P}$ being its centered version, we have asymtotically:

$$\mathbb{E}\left[\mathring{\mathbf{X}}\mathring{\mathbf{X}}^\intercal\right] = (1-\frac{1}{n})\mathbb{E}\left[\mathbf{X}\mathbf{X}^\intercal\right] - \frac{1}{n}\mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{X}]^\intercal \tag{A.4}$$

$$\begin{cases} \mathbb{E}\left[\mathbf{X}^\intercal\mathbf{X}\right] = \mathbb{E}[\mathbf{X}]^\intercal\mathbb{E}[\mathbf{X}] + \operatorname{Tr}(\mathbf{\Sigma})\mathbf{I}_n, \\ \mathbb{E}\left[\mathring{\mathbf{X}}^\intercal\mathring{\mathbf{X}}\right] = \mathbb{E}[\mathring{\mathbf{X}}]^\intercal\mathbb{E}[\mathring{\mathbf{X}}] + \operatorname{Tr}(\mathbf{\Sigma})\mathbf{I}_n - \frac{1}{n}\operatorname{Tr}(\mathbf{\Sigma})\mathbb{1}_n\mathbb{1}_n^\intercal \end{cases} \tag{A.5}$$

While the matrix $\mathbf{X}\mathbf{X}^\intercal$ is practically unaffected by the centering, the matrix $\mathbf{X}^\intercal\mathbf{X}$ is changed, not only by the centering of the mean matrix $\mathbb{E}[\mathbf{X}]^\intercal\mathbb{E}[\mathbf{X}]$, but also with the appearance of the bias term $-\frac{1}{n}\operatorname{Tr}(\mathbf{\Sigma})\mathbb{1}\mathbb{1}^\intercal$. With these results in mind, $\mathbf{X}^t, \boldsymbol{\mu}_j^t, \mathbf{Z}$ will still denote their centered versions, as it was the case until now.

A.5.1 First order deterministic equivalent

We recall that

$$\mathbf{Q} = \left(\mathbf{I}_{Tp} - \frac{\mathbf{A}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^\intercal\mathbf{A}^{\frac{1}{2}}}{Tp}\right)^{-1} = \left(\mathbf{I}_{Tp} - \frac{\tilde{\mathbf{Z}}\tilde{\mathbf{Z}}^\intercal}{Tp}\right)^{-1},$$

where $\tilde{\mathbf{Z}} = \mathbf{A}^{\frac{1}{2}}\mathbf{Z}$. We start by "guessing" that a deterministic equivalent must be of the form $\bar{\mathbf{Q}} = \mathbf{F}^{-1}$. Then:

$$\mathbf{Q} - \bar{\mathbf{Q}} = \mathbf{Q}\left(\mathbf{F} - \mathbf{I}_{Tp} + \frac{1}{Tp}\tilde{\mathbf{Z}}\tilde{\mathbf{Z}}^\intercal\right)\bar{\mathbf{Q}},$$

using the fact that $\mathbf{A}^{-1} - \mathbf{B}^{-1} = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A})\mathbf{B}^{-1}$, for any invertible matrices $\mathbf{A}$ and $\mathbf{B}$.

The columns of $\tilde{\mathbf{Z}}$ are the vectors $\tilde{\mathbf{z}}_i$ such that:

$$\tilde{\mathbf{z}}_i = (\mathbf{C}^t)^{\frac{1}{2}}\mathbf{z}_i^0 + \mathbf{A}^{\frac{1}{2}}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t),$$

where $\mathbf{C}^t = \mathbf{A}^{\frac{1}{2}}(\mathbf{E}_{tt} \otimes \mathbf{I}_p)\mathbf{A}^{\frac{1}{2}}$ and $\mathbf{z}_i^0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{Tp})$. Then $\mathbf{Q} - \bar{\mathbf{Q}}$ rewrites:

$$\mathbf{Q} - \bar{\mathbf{Q}} = \mathbf{Q}\left(\mathbf{F} - \mathbf{I}_{Tp} + \frac{1}{Tp}\sum_{i=1}^{n}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\right)\bar{\mathbf{Q}},$$

For $\bar{\mathbf{Q}}$ to be a deterministic equivalent, we must have, for any deterministic matrix $\mathbf{B} \in \mathbb{R}^{Tp\times Tp}$:

$$\begin{aligned}&\frac{1}{Tp}\operatorname{Tr}[\mathbf{B}(\mathbf{Q} - \bar{\mathbf{Q}})] \xrightarrow{a.s.} 0\\ \Leftrightarrow &\frac{1}{Tp}\operatorname{Tr}[\mathbf{B}\mathbf{Q}(\mathbf{F} - \mathbf{I}_{Tp})\bar{\mathbf{Q}}] + \frac{1}{(Tp)^2}\sum_{i=1}^{n}\tilde{\mathbf{z}}_i^\intercal\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i \xrightarrow{a.s.} 0\end{aligned} \tag{A.6}$$

Let us study the quantity $\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i$. Using Sherman-Morrison formula, it can be proved that:

$$\mathbf{Q} = \mathbf{Q}_{-i} + \frac{1}{Tp}\frac{\mathbf{Q}_{-i}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}}{1 - \frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}\tilde{\mathbf{z}}_i} \tag{A.7}$$

with $\mathbf{Q}_{-i} = \left(\mathbf{I}_{Tp} - \frac{\tilde{\mathbf{Z}}_{-i}\tilde{\mathbf{Z}}_{-i}^\intercal}{Tp}\right)^{-1}$, the notation $\tilde{\mathbf{Z}}_{-i}$ standing for the matrix $\tilde{\mathbf{Z}}$ with the i-th column removed.

In particular:

$$\tilde{\mathbf{z}}_i^\intercal\mathbf{Q} = \frac{\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}}{1 - \frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}\tilde{\mathbf{z}}_i}, \tag{A.8}$$

which means that

$$\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i = \frac{1}{Tp}\frac{\tilde{\mathbf{z}}_i^\intercal\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}_{-i}\tilde{\mathbf{z}}_i}{1 - \frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}\tilde{\mathbf{z}}_i} \tag{A.9}$$

Lemma 38
For any deterministic matrix $\mathbf{B} \in \mathbb{R}^{Tp\times Tp}$:

$$\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal\mathbf{B}\tilde{\mathbf{z}}_i - \frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{B}] \xrightarrow{a.s.} 0,$$

where $\mathbf{C}_j^t = \mathbf{A}^{\frac{1}{2}}(\mathbf{E}_{tt} \otimes (\mathbf{I}_p + \boldsymbol{\mu}_j^t\boldsymbol{\mu}_j^\intercal))\mathbf{A}^{\frac{1}{2}}$.

Proof:

$$\begin{aligned}\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal \mathbf{B}\tilde{\mathbf{z}}_i = &\frac{1}{Tp}\mathbf{z}_i^{0\intercal}(\mathbf{C}^t)^{\frac{1}{2}}\mathbf{B}(\mathbf{C}^t)^{\frac{1}{2}}\mathbf{z}_i^0 \\ &+ \frac{1}{Tp}\mathbf{z}_i^{0\intercal}(\mathbf{C}^t)^{\frac{1}{2}}(\mathbf{B}+\mathbf{B}^\intercal)\mathbf{A}^{\frac{1}{2}}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t) \\ &+ \frac{1}{Tp}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t)^\intercal \mathbf{A}^{\frac{1}{2}}\mathbf{B}\mathbf{A}^{\frac{1}{2}}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t)\end{aligned}$$

Considering successively the 3 terms:

- $\frac{1}{Tp}\mathbf{z}_i^{0\intercal}(\mathbf{C}^t)^{\frac{1}{2}}\mathbf{B}(\mathbf{C}^t)^{\frac{1}{2}}\mathbf{z}_i^0 - \frac{1}{Tp}\operatorname{Tr}[\mathbf{C}^t\mathbf{B}] \xrightarrow{a.s.} 0$
- $\frac{1}{Tp}\mathbf{z}_i^{0\intercal}(\mathbf{C}^t)^{\frac{1}{2}}(\mathbf{B}+\mathbf{B}^\intercal)\mathbf{A}^{\frac{1}{2}}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t)$ has zero mean, and its variance is of order $O(\frac{1}{p^2})$, therefore it can be proven that it converges almost surely to 0.
- $\frac{1}{Tp}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t)^\intercal \mathbf{A}^{\frac{1}{2}}\mathbf{B}\mathbf{A}^{\frac{1}{2}}(\mathbf{e}_t \otimes \boldsymbol{\mu}_j^t) = \frac{1}{Tp}\operatorname{Tr}[\mathbf{A}^{\frac{1}{2}}(\mathbf{E}_{tt} \otimes \boldsymbol{\mu}_j^t\boldsymbol{\mu}_j^{t\intercal})\mathbf{A}^{\frac{1}{2}}\mathbf{B}]$

Adding the first and the third term, we get $\frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{B}]$, which concludes the proof.

Therefore, by applying Lemma 38 to equation (A.9):

$$\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i - \frac{1}{Tp}\frac{\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}_{-i}]}{1-\frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{Q}_{-i}]}0.$$

It can be proven (see [12]) that $\frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{Q}_{-i}] \simeq \frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{Q}]$. Using the definition of a deterministic equivalent, we also have $\frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\mathbf{Q}] \xrightarrow{a.s.} \frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}]$. Finally, using the fact that $\mathbf{C}_j^t$ is a small-rank perturbation of $\mathbf{C}^t$, we have $\frac{1}{Tp}\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}] - \frac{1}{Tp}\operatorname{Tr}[\mathbf{C}^t\bar{\mathbf{Q}}] \xrightarrow{a.s.} 0$. Let us define the following quantities:

$$\tilde{\delta}_j^t = \frac{\rho_j^t}{Tc}\frac{1}{1-\delta^t} \quad \text{and} \quad \delta^t = \frac{1}{Tp}\operatorname{Tr}\left(\mathbf{C}^t\bar{\mathbf{Q}}\right).$$

Therefore, the previous equation rewrites as:

$$\frac{1}{Tp}\tilde{\mathbf{z}}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i = \frac{1}{Tp}\frac{\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}]}{1-\delta^t}$$

$$\begin{aligned}\frac{1}{(Tp)^2}\sum_{i=1}^n \tilde{\mathbf{z}}_i^\intercal \bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}\tilde{\mathbf{z}}_i &= \frac{1}{Tp}\sum_{i=1}^n \frac{1}{Tp}\frac{\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}]}{1-\delta^t} \\ = \frac{1}{Tp}\sum_{t,j}\frac{n_j^t}{Tp}\frac{\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}]}{1-\delta^t} &= \frac{1}{Tp}\sum_{t,j}\tilde{\delta}_j^t\operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{B}\mathbf{Q}]\end{aligned}$$

Going back to equation (A.6), we have the following condition that would ensure $\bar{\mathbf{Q}} \leftrightarrow \mathbf{Q}$:

$$\frac{1}{Tp}\operatorname{Tr}[\mathbf{BQ}(\mathbf{F}-\mathbf{I}_{Tp})\bar{\mathbf{Q}}] + \frac{1}{Tp}\sum_{t,j}\tilde{\delta}_j^t \operatorname{Tr}[\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{BQ}] \xrightarrow{a.s.} 0$$
$$\Leftrightarrow \frac{1}{Tp}\operatorname{Tr}\left[\mathbf{BQ}\left(\mathbf{F}-\mathbf{I}_{Tp}+\sum_{t,j}\tilde{\delta}_j^t\mathbf{C}_j^t\right)\bar{\mathbf{Q}}\right] \xrightarrow{a.s.} 0$$

Therefore, we must have:

$$\bar{\mathbf{Q}} = \mathbf{F}^{-1} = \left(\mathbf{I}_{Tp} - \sum_{t,j}\tilde{\delta}_j^t\mathbf{C}_j^t\right)^{-1}$$

Let's compute more explicitly the quantity δ^t. Let us define $\bar{\delta}^t = \tilde{\delta}_1^t + \tilde{\delta}_2^t$, and $\bar{\mathbf{M}} = \mathbf{A}^{\frac{1}{2}}[\mathbf{e}_1^{[T]}\otimes\boldsymbol{\mu}_1^1, \mathbf{e}_1^{[T]}\otimes\boldsymbol{\mu}_2^1, \ldots, \mathbf{e}_T^{[T]}\otimes\boldsymbol{\mu}_2^T]$. Then,

$$\begin{aligned}
&\sum_{t,j}\tilde{\delta}_j^t\mathbf{C}_j^t = \sum_{t,j}\tilde{\delta}_j^t\mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{tt}\otimes\left(\mathbf{I}_p + \boldsymbol{\mu}_j^t\boldsymbol{\mu}_j^{t\intercal}\right)\right)\mathbf{A}^{\frac{1}{2}}\\
=&\sum_t \bar{\delta}^t\mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}\\
+&\sum_{t,j}\tilde{\delta}_j^t\mathbf{A}^{\frac{1}{2}}\left(\mathbf{e}_t^{[T]}\otimes\boldsymbol{\mu}_j^t\right)\left(\mathbf{e}_t^{[T]}\otimes\boldsymbol{\mu}_j^t\right)^\intercal\mathbf{A}^{\frac{1}{2}}\\
=&\mathbf{A}^{\frac{1}{2}}\left(\mathcal{D}_{\bar{\boldsymbol{\delta}}}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}} + \bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\\
=&\left(\tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\mathcal{D}_{\bar{\boldsymbol{\delta}}}\tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\right)\otimes\mathbf{I}_p + \bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal.
\end{aligned}$$

If $\bar{\mathbf{Q}}_0 = \left(\mathbf{I}_T - \tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\mathcal{D}_{\bar{\boldsymbol{\delta}}}\tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\right)^{-1}\otimes\mathbf{I}_p$, using Woodbury identity,

$$\begin{aligned}
\bar{\mathbf{Q}} &= \left(\left(\mathbf{I}_T - \tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\mathcal{D}_{\bar{\boldsymbol{\delta}}}\tilde{\boldsymbol{\Lambda}}^{\frac{1}{2}}\right)\otimes\mathbf{I}_p - \bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\right)^{-1}\\
&= \left(\bar{\mathbf{Q}}_0^{-1} - \bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\right)^{-1} = \left(\bar{\mathbf{Q}}_0^{-1} - \mathbf{U}\mathbf{U}^\intercal\right)^{-1}\\
&= \bar{\mathbf{Q}}_0 + \bar{\mathbf{Q}}_0\mathbf{U}\left(\mathbf{I}_{2T} - \mathbf{U}^\intercal\bar{\mathbf{Q}}_0\mathbf{U}\right)^{-1}\mathbf{U}^\intercal\bar{\mathbf{Q}}_0.
\end{aligned}$$

The second term is a matrix of rank $2T << Tp$ and can be neglected in the trace. Then,

$$\operatorname{Tr}\left(\mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}\bar{\mathbf{Q}}\right) = \operatorname{Tr}\left(\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}\bar{\mathbf{Q}}_0\mathbf{A}^{\frac{1}{2}}\right)$$

$$\begin{aligned}\mathbf{A}^{\frac{1}{2}}\bar{\mathbf{Q}}_0\mathbf{A}^{\frac{1}{2}} &= \left[\tilde{\mathbf{\Lambda}}^{\frac{1}{2}}\left(\mathbf{I}_T - \tilde{\mathbf{\Lambda}}^{\frac{1}{2}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\tilde{\mathbf{\Lambda}}^{\frac{1}{2}}\right)^{-1}\tilde{\mathbf{\Lambda}}^{\frac{1}{2}}\right]\otimes\mathbf{I}_p \\ &= \underbrace{\left[\tilde{\mathbf{\Lambda}} + \tilde{\mathbf{\Lambda}}\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}^{-1} - \tilde{\mathbf{\Lambda}}\right)^{-1}\tilde{\mathbf{\Lambda}}\right]}_{=\boldsymbol{\mathcal{A}}}\otimes\mathbf{I}_p.\end{aligned}$$

Finally,

$$\delta^t = \frac{1}{Tp}\operatorname{Tr}\left(\mathbf{E}_{tt}\boldsymbol{\mathcal{A}}\right)\operatorname{Tr}(\mathbf{I}_p) = \frac{1}{T}\boldsymbol{\mathcal{A}}_{tt}.$$

We can then conclude that the $\{\delta^t\}_t$ are the solution of the following system of equations:

$$\begin{cases} \forall t,\ \delta^t = \frac{1}{T}\boldsymbol{\mathcal{A}}_{tt}, \\ \boldsymbol{\mathcal{A}} = \tilde{\mathbf{\Lambda}} + \tilde{\mathbf{\Lambda}}\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}^{-1} - \tilde{\mathbf{\Lambda}}\right)^{-1}\tilde{\mathbf{\Lambda}}. \end{cases}$$

A.5.2 Computation of the mean

$$\mathbb{E}[f_i] = \frac{1}{Tp}\sum_{i'}\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'}\right]y_{i'}.$$

Using equation (A.8), we have:

$$\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'}\right] = \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i}\tilde{\mathbf{z}}_{i'}\right]}{1-\delta^t}.$$

And thanks to equation (A.5),

$$\begin{aligned}&\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i,-i'}\tilde{\mathbf{z}}_{i'}\right] \\ &= \mathbb{E}\left[\tilde{\mathbf{z}}_i\right]^\intercal\bar{\mathbf{Q}}\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}\right] - \frac{1}{n^t}\mathbb{E}\left[\operatorname{Tr}(\mathbf{C}^t\mathbf{Q})\right]\mathbb{1}_{t=t'} \\ &= \mathbf{e}_{t,j}^{[2T]\intercal}\bar{\mathbf{M}}^\intercal\bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t',j'}^{[2T]} - \frac{Tp}{n^t}\delta^t\mathbf{e}_{t,j}^{[2T]\intercal}(\mathbf{I}_T\otimes\mathbb{1}_2\mathbb{1}_2^\intercal)\mathbf{e}_{t',j'}^{[2T]},\end{aligned}$$

with

$$\mathbf{C}^t := \mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}.$$

So finally

$$\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal \mathbf{Q} \tilde{\mathbf{z}}_{i'}\right] \simeq \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal \mathbf{Q}_{-i,-i'} \tilde{\mathbf{z}}_{i'}\right]}{(1-\delta^t)(1-\delta^{t'})}$$
$$= \frac{\mathbf{e}_{t,j}^{[2T]\intercal}}{1-\delta^t}\left(\bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}} \bar{\mathbf{M}} - \frac{Tc}{\rho^t}\delta^t \mathbf{I}_T \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal\right) \frac{\mathbf{e}_{t',j'}^{[2T]}}{1-\delta^{t'}}.$$

Going back to the quantity we want to estimate,

$$\mathbb{E}[f_i] = \frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\tilde{\mathbf{z}}_i^\intercal \mathbf{Q} \tilde{\mathbf{z}}_{i'}\right] y_{i'}$$
$$= \frac{\mathbf{e}_{t,j}^{[2T]\intercal}}{1-\delta^t}\left(\bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}} \bar{\mathbf{M}} - \frac{Tc}{\rho^t}\delta^t \mathbf{I}_T \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal\right) \sum_{i'} \frac{y_{i'}}{Tp} \frac{\mathbf{e}_{t',j'}^{[2T]}}{1-\delta^{t'}},$$

where we have

$$\sum_{i'} \frac{y_{i'}}{Tp} \frac{\mathbf{e}_{t',j'}^{[2T]}}{1-\delta^{t'}} = \sum_{t',j'} \frac{n_{j'}^{t'}}{Tp(1-\delta^{t'})} \frac{n_{\ell j'}^{t'}}{n_{j'}^{t'}} \frac{\mathbf{e}_{t',j'}^{[2T]}}{n_{\ell j'}^{t'}} \sum_{i'|x_{i'} \in \mathcal{C}_{j'}^{t'}} (d_{i'1}^{t'} \tilde{y}_1^{t'} + d_{i'2}^{t'} \tilde{y}_2^{t'})$$
$$= \sum_{t',j'} \tilde{\delta}_{j'}^{t'} \eta_{j'}^{t'} \left[\frac{\tilde{y}_1^{t'}}{n_{\ell j'}^{t'}} \sum_{i'|x_{i'} \in \mathcal{C}_{j'}^{t'}} d_{i'1}^{t'} + \frac{\tilde{y}_2^{t'}}{n_{\ell j'}^{t'}} \sum_{i'|x_{i'} \in \mathcal{C}_{j'}^{t'}} d_{i'2}^{t'}\right] \mathbf{e}_{t',j'}^{[2T]}$$
$$= \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}},$$

with $\bar{d}_{j_1,j_2}^t = \frac{1}{n_{\ell j_1}^t} \sum_{i'|x_i \in \mathcal{C}_{j_1}^t} d_{i'j_2}^t$ and

$$\bar{\mathbf{D}} = \sum_{t=1}^{T} \mathbf{E}_{tt} \otimes \begin{pmatrix} \bar{d}_{11}^t & \bar{d}_{12}^t \\ \bar{d}_{21}^t & \bar{d}_{22}^t \end{pmatrix}.$$

So we have

$$\mathbb{E}[f_i] = \frac{\mathbf{e}_{t,j}^{[2T]\intercal}}{1-\delta^t}\left(\bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}} \bar{\mathbf{M}} - \frac{Tc}{\rho^t}\delta^t \mathbf{I}_T \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal\right) \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}}.$$

Let us recall the small-dimensional quantity

$$\boldsymbol{\Theta}_0 = (\boldsymbol{\mathcal{A}} \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal) \odot \mathbf{M}^\intercal \mathbf{M} = \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}_0 \bar{\mathbf{M}}$$

$$
\begin{aligned}
\bar{\mathbf{Q}}\bar{\mathbf{M}} &= \left(\bar{\mathbf{Q}}_0^{-1} - \bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\right)^{-1}\bar{\mathbf{M}} \\
&= \left(\mathbf{I}_{Tp} - \bar{\mathbf{Q}}_0\bar{\mathbf{M}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\right)^{-1}\bar{\mathbf{Q}}_0\bar{\mathbf{M}} \\
&= \bar{\mathbf{Q}}_0\bar{\mathbf{M}}\left(\mathbf{I}_{2T} - \mathcal{D}_{\tilde{\boldsymbol{\delta}}}\bar{\mathbf{M}}^\intercal\bar{\mathbf{Q}}_0\bar{\mathbf{M}}\right)^{-1} \\
&= \bar{\mathbf{Q}}_0\bar{\mathbf{M}}\left(\mathbf{I}_{2T} - \mathcal{D}_{\tilde{\boldsymbol{\delta}}}\boldsymbol{\Theta}_0\right)^{-1}
\end{aligned}
$$

Let us define

$$
\boldsymbol{\Theta} := \bar{\mathbf{M}}^\intercal\bar{\mathbf{Q}}\bar{\mathbf{M}} = \boldsymbol{\Theta}_0\left(\mathbf{I}_{2T} - \mathcal{D}_{\tilde{\boldsymbol{\delta}}}\boldsymbol{\Theta}_0\right)^{-1}
$$

$$
\mathbb{E}[f_i] = \frac{\mathbf{e}_{t,j}^{[2T]\intercal}}{1-\delta^t}\left(\boldsymbol{\Theta} - \frac{Tc}{\rho^t}\delta^t\mathbf{I}_T \otimes \mathbb{1}_2\mathbb{1}_2^\intercal\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}
$$

As $\boldsymbol{\Gamma} = \mathbf{I}_T \otimes \mathbb{1}_2\mathbb{1}_2^\intercal$ and $\gamma^t = \frac{Tc\delta^t}{\rho^t}$, we can conclude that:

$$
m_j^t = \mathbb{E}[f_i] = \frac{\mathbf{e}_{t,j}^{[2T]\intercal}}{1-\delta^t}\left(\boldsymbol{\Theta} - \gamma^t\boldsymbol{\Gamma}\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}. \tag{A.10}
$$

A.5.3 Second order deterministic equivalent

In the following, we will need a deterministic equivalent for $\mathbf{Q}\mathbf{C}_j^t\mathbf{Q}$, which is not merely $\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}$. We will perform an analysis similar to the one performed in A.5.1. Instead of working with the condition $\frac{1}{m}\operatorname{Tr}\mathbf{B}(\mathbf{Q}-\bar{\mathbf{Q}}) \xrightarrow{a.s.} 0$, we will use the condition $\mathbf{u}^\intercal(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{v} \xrightarrow{a.s.} 0$, where $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{Tp}$ are of unit norm.

$$
\begin{aligned}
&\mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}\mathbf{C}_j^t\mathbf{Q}\mathbf{v}\right] - \mathbf{u}^\intercal\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{v} \\
&= \mathbb{E}\left[\mathbf{u}^\intercal(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{C}_j^t\mathbf{Q}\mathbf{v}\right] + \underbrace{\mathbb{E}\left[\mathbf{u}^\intercal\bar{\mathbf{Q}}\mathbf{C}_j^t(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{v}\right]}_{\to 0} \\
&= \mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}(\bar{\mathbf{Q}}^{-1} - \mathbf{Q}^{-1})\bar{\mathbf{Q}}\mathbf{C}_j^t\mathbf{Q}\mathbf{v}\right] \\
&= \mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}\left(-\sum_{t',j'}\tilde{\delta}_{j'}^{t'}\mathbf{C}_{j'}^{t'} + \frac{\tilde{\mathbf{Z}}\tilde{\mathbf{Z}}^\intercal}{Tp}\right)\bar{\mathbf{Q}}\mathbf{C}_j^t\mathbf{Q}\mathbf{v}\right] \\
&:= -A_1 + A_2
\end{aligned}
$$

Using the shortcut notation $\mathbf{w}_i = \mathbf{Q}_{-i}\tilde{\mathbf{z}}_i$,

$$
\begin{aligned}
A_2 &= \mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q} \frac{\tilde{\mathbf{Z}}\tilde{\mathbf{Z}}^\intercal}{Tp} \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}\mathbf{v}\right] \\
&= \frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}\mathbf{v}\right] \\
&= \frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \frac{\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}}{1-\delta^{t'}}\tilde{\mathbf{z}}_{i'}^\intercal \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}\mathbf{v}\right] \\
&= \underbrace{\frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \frac{\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}}{1-\delta^{t'}}\tilde{\mathbf{z}}_{i'}^\intercal \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}_{-i'}\mathbf{v}\right]}_{:=B_1} \\
&+ \underbrace{\frac{1}{(Tp)^2}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \frac{\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}}{1-\delta^{t'}}\tilde{\mathbf{z}}_{i'}^\intercal \bar{\mathbf{Q}} \mathbf{C}_j^t \frac{\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i'}}{1-\delta^{t'}}\mathbf{v}\right]}_{:=B_2},
\end{aligned}
$$

where the last two equalities are obtained through (A.8) and (A.7) respectively. B_1 and A_1 cancel each other out:

$$
\begin{aligned}
B_1 &= \frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \frac{\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}}{1-\delta^{t'}}\tilde{\mathbf{z}}_{i'}^\intercal \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}_{-i'}\mathbf{v}\right] \\
&= \frac{1}{Tp}\sum_{i'} \mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q}_{-i'} \mathbb{E}\left[\frac{\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal}{1-\delta^{t'}}\right] \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}_{-i'}\mathbf{v}\right] \\
&= \mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q}\left(\sum_{i'} \frac{1}{Tp}\frac{\mathbf{C}_{j'}^{t'}}{1-\delta^{t'}}\right) \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}\mathbf{v}\right] \\
&= \mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q}\left(\sum_{t',j'} \tilde{\delta}_{j'}^{t'} \mathbf{C}_{j'}^{t'}\right) \bar{\mathbf{Q}} \mathbf{C}_j^t \mathbf{Q}\mathbf{v}\right] = A_1
\end{aligned}
$$

Finally we have

$$\begin{aligned}
&\mathbb{E}\left[\mathbf{u}^\intercal \mathbf{Q}\mathbf{C}_j^t\mathbf{Q}\mathbf{v}\right] - \mathbf{u}^\intercal\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{v} = B_2\\
&= \frac{1}{(Tp)^2}\sum_{i'}\frac{1}{(1-\delta^{t'})^2}\mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal\bar{\mathbf{Q}}\mathbf{C}_j^t\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}_{-i'}\mathbf{v}\right]\\
&= \frac{1}{(Tp)^2}\sum_{i'}\frac{1}{(1-\delta^{t'})^2}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right)\mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}_{-i'}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}_{-i'}\mathbf{v}\right]\\
&= \frac{1}{(Tp)^2}\sum_{i'}\frac{1}{(1-\delta^{t'})^2}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right)\mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}\mathbf{C}_{j'}^{t'}\mathbf{Q}\mathbf{v}\right]\\
&= \sum_{t',j'}\frac{n_{j'}^{t'}}{(Tp)^2(1-\delta^{t'})^2}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right)\mathbb{E}\left[\mathbf{u}^\intercal\mathbf{Q}\mathbf{C}_{j'}^{t'}\mathbf{Q}\mathbf{v}\right]
\end{aligned}$$

Until now we have the following (not deterministic) equivalent:

$$\mathbf{Q}\mathbf{C}_j^t\mathbf{Q} \leftrightarrow \bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}} + \sum_{t',j'}\frac{n_{j'}^{t'}}{(Tp)^2(1-\delta^{t'})^2}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right)\mathbf{Q}\mathbf{C}_{j'}^{t'}\mathbf{Q} \tag{A.11}$$

First of all, as $\mathbf{C}_j^t$ is a small-rank perturbation of $\mathbf{C}^t$ and $\bar{\mathbf{Q}}$ is a small-rank perturbation of $\bar{\mathbf{Q}}_0$:

$$\begin{aligned}
&\frac{1}{Tp}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right) = \frac{1}{Tp}\operatorname{Tr}\left(\bar{\mathbf{Q}}_0\mathbf{C}^t\bar{\mathbf{Q}}_0\mathbf{C}^{t'}\right)\\
&= \frac{1}{Tp}\operatorname{Tr}\left(\bar{\mathbf{Q}}_0\mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}\bar{\mathbf{Q}}_0\mathbf{A}^{\frac{1}{2}}\left(\mathbf{E}_{t't'}\otimes\mathbf{I}_p\right)\mathbf{A}^{\frac{1}{2}}\right)\\
&= \frac{1}{Tp}\operatorname{Tr}\left(\left(\mathbf{E}_{tt}\otimes\mathbf{I}_p\right)\left(\boldsymbol{\mathcal{A}}\otimes\mathbf{I}_p\right)\left(\mathbf{E}_{t't'}\otimes\mathbf{I}_p\right)\left(\boldsymbol{\mathcal{A}}\otimes\mathbf{I}_p\right)\right)\\
&= \frac{1}{T}\operatorname{Tr}\left(\mathbf{E}_{tt}\boldsymbol{\mathcal{A}}\mathbf{E}_{t't'}\boldsymbol{\mathcal{A}}\right) = \frac{\boldsymbol{\mathcal{A}}_{tt'}^2}{T}
\end{aligned}$$

Let us define

$$\begin{aligned}
\bar{\mathbf{S}}^{tt'} &= \frac{1}{Tc(1-\delta^{t'})^2}\frac{1}{Tp}\operatorname{Tr}\left(\bar{\mathbf{Q}}\mathbf{C}^t\bar{\mathbf{Q}}\mathbf{C}^{t'}\right) = \frac{\boldsymbol{\mathcal{A}}_{tt'}^2}{T^2c(1-\delta^{t'})^2}\\
\mathbf{S}^{tt'} &= \frac{1}{Tc(1-\delta^{t'})^2}\frac{1}{Tp}\mathbb{E}\left[\operatorname{Tr}\left(\mathbf{Q}\mathbf{C}^t\mathbf{Q}\mathbf{C}^{t'}\right)\right].
\end{aligned}$$

One can deduce from (A.11) that $\mathbf{S} = \bar{\mathbf{S}} + \bar{\mathbf{S}}\mathcal{D}_{\bar{\boldsymbol{\rho}}}\mathbf{S}$, so $\mathbf{S} = \bar{\mathbf{S}}\left(\mathbf{I}_T - \mathcal{D}_{\bar{\boldsymbol{\rho}}}\bar{\mathbf{S}}\right)^{-1}$. Similarly, it follows that:

$$\mathbf{Q}\mathbf{C}_j^t\mathbf{Q} \leftrightarrow \bar{\mathbf{Q}}\mathbf{C}_j^t\bar{\mathbf{Q}} + \sum_{t',j'}\rho_{j'}^{t'}\mathbf{S}^{tt'}\bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\bar{\mathbf{Q}}.$$

A.5.4 Computation of the variance

The second order moment can be computed as:

$$\begin{aligned}\mathbf{E}[f_i^2] &= \frac{1}{(Tp)^2}\sum_{i',i''}\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i''}\right]y_{i'}y_{i''}\\ &= \underbrace{\frac{1}{(Tp)^2}\sum_{i'}\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'}\right]y_{i'}^2}_{:=C_1}\\ &+ \underbrace{\frac{1}{(Tp)^2}\sum_{i'\neq i''}\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i''}\right]y_{i'}y_{i''}}_{:=C_2}.\end{aligned}$$

- Computation of C_1

$$\begin{aligned}&(1-\delta^t)^2(1-\delta^{t'})^2\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'}\right]\\ &= \mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal\mathbf{Q}_{-i,-i'}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i,-i'}\tilde{\mathbf{z}}_{i'}\right]\\ &= \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}_{-i,-i'}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\intercal\mathbf{Q}_{-i,-i'}\tilde{\mathbf{z}}_{i'}\tilde{\mathbf{z}}_{i'}^\intercal\right)\right]\\ &= \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t\mathbf{Q}\mathbf{C}^{t'}\right)\right]\end{aligned}$$

Therefore:

$$\begin{aligned}C_1 &= \frac{1}{(Tp)^2}\sum_{i'}\frac{\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t\mathbf{Q}\mathbf{C}^{t'}\right)\right]}{(1-\delta^t)^2(1-\delta^{t'})^2}y_{i'}^2\\ &= \sum_{i'}\frac{\mathbf{S}^{tt'}}{n(1-\delta^t)^2}y_{i'}^2\\ &= \sum_{t'}\frac{\rho^{t'}\mathbf{S}^{tt'}}{(1-\delta^t)^2}\eta^{t'}\underbrace{\left(\frac{1}{n_\ell^{t'}}\sum_{i'|\mathbf{x}_{i'}\in\mathcal{C}^{t'}}\left(d_{i'1}^{t'}\tilde{y}_1^{t'}+d_{i'2}^{t'}\tilde{y}_2^{t'}\right)^2\right)}_{=\begin{pmatrix}\tilde{y}_1^{t'} & \tilde{y}_2^{t'}\end{pmatrix}\begin{pmatrix}\tilde{d}_{11}^{t'} & \tilde{d}_{12}^{t'}\\ \tilde{d}_{21}^{t'} & \tilde{d}_{22}^{t'}\end{pmatrix}\begin{pmatrix}\tilde{y}_1^{t'}\\ \tilde{y}_2^{t'}\end{pmatrix}},\end{aligned}$$

with $\tilde{d}_{j_1j_2}^t = \frac{1}{n_\ell^t}\sum_{i|\mathbf{x}_i\in\mathcal{C}^t}d_{ij_1}^t d_{ij_2}^t$. If we further define

$$\begin{aligned}\tilde{\mathbf{D}} &= \sum_{t=1}^T\mathbf{E}_{tt}\otimes\begin{pmatrix}\tilde{d}_{11}^t & \tilde{d}_{12}^t\\ \tilde{d}_{21}^t & \tilde{d}_{22}^t\end{pmatrix}\\ \mathbf{T}^t &= \frac{1}{(1-\delta^t)^2}\mathcal{D}_{\bar{\boldsymbol{\rho}}\odot\bar{\boldsymbol{\eta}}\odot\mathbf{S}^{t\cdot}}\otimes\mathbb{1}_2\mathbb{1}_2^\intercal,\end{aligned}$$

then,

$$C_1 = \tilde{\mathbf{y}}^\intercal \left(\mathbf{T}^t \odot \tilde{\mathbf{D}} \right) \tilde{\mathbf{y}}.$$

- Computation of C_2
  Using A.8, we have

$$\begin{aligned}
\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q} \tilde{\mathbf{z}}_i \tilde{\mathbf{z}}_i^\intercal \mathbf{Q} \tilde{\mathbf{z}}_{i''}\right] &= \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i,-i'} \tilde{\mathbf{z}}_i \tilde{\mathbf{z}}_i^\intercal \mathbf{Q}_{-i,-i''} \tilde{\mathbf{z}}_{i''}\right]}{(1-\delta^{t'})(1-\delta^t)^2(1-\delta^{t''})} \\
&= \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i'} \mathbf{C}_j^t \mathbf{Q}_{-i''} \tilde{\mathbf{z}}_{i''}\right]}{(1-\delta^{t'})(1-\delta^t)^2(1-\delta^{t''})},
\end{aligned}$$

and using A.7, we have

$$\begin{aligned}
&\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i'} \mathbf{C}_j^t \mathbf{Q}_{-i''} \tilde{\mathbf{z}}_{i''}\right] \\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \mathbf{C}_j^t \mathbf{Q}_{-i''} \tilde{\mathbf{z}}_{i''}\right] + \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \tilde{\mathbf{z}}_{i''}\right]}{Tp(1-\delta^{t''})} \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t \mathbf{Q}\mathbf{C}^{t''}\right)\right] \\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \mathbf{C}_j^t \mathbf{Q}_{-i'',-i'} \tilde{\mathbf{z}}_{i''}\right] + \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \tilde{\mathbf{z}}_{i''}\right]}{Tp(1-\delta^{t'})} \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t \mathbf{Q}\mathbf{C}^{t'}\right)\right] \\
&+ \frac{\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \tilde{\mathbf{z}}_{i''}\right]}{Tp(1-\delta^{t''})} \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t \mathbf{Q}\mathbf{C}^{t''}\right)\right] \\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \mathbf{C}_j^t \mathbf{Q}_{-i'',-i'} \tilde{\mathbf{z}}_{i''}\right] + \frac{\mathbf{e}_{t',j'}^{[2T]\intercal}(\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma}) \mathbf{e}_{t'',j''}^{[2T]}}{Tp(1-\delta^{t'})} \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t \mathbf{Q}\mathbf{C}^{t'}\right)\right] \\
&+ \frac{\mathbf{e}_{t',j'}^{[2T]\intercal}(\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma}) \mathbf{e}_{t'',j''}^{[2T]}}{Tp(1-\delta^{t''})} \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t \mathbf{Q}\mathbf{C}^{t''}\right)\right].
\end{aligned}$$

Using A.5,

$$\begin{aligned}
&\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^\intercal \mathbf{Q}_{-i',-i''} \mathbf{C}_j^t \mathbf{Q}_{-i'',-i'} \tilde{\mathbf{z}}_{i''}\right] \\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i'}\right]^\intercal \mathbb{E}\left[\mathbf{Q}_{-i',-i''} \mathbf{C}_j^t \mathbf{Q}_{-i'',-i'}\right] \mathbb{E}\left[\tilde{\mathbf{z}}_{i''}\right] - \frac{1}{n^{t'}} \mathbb{E}\left[\mathrm{Tr}(\mathbf{Q}\mathbf{C}_j^t \mathbf{Q}\mathbf{C}^{t'}\right] \mathbb{1}_{t'=t''} \\
&= \mathbf{e}_{t',j'}^{[2T]\intercal} \bar{\mathbf{M}}^\intercal \mathbb{E}\left[\mathbf{Q}\mathbf{C}_j^t \mathbf{Q}\right] \bar{\mathbf{M}} \mathbf{e}_{t'',j''}^{[2T]} - \frac{1}{n^{t'}} \mathbb{E}\left[\mathrm{Tr}(\mathbf{Q}\mathbf{C}_j^t \mathbf{Q}\mathbf{C}^{t'}\right] \mathbf{e}_{t',j'}^{[2T]\intercal} (\mathbf{I}_T \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal) \mathbf{e}_{t'',j''}^{[2T]}.
\end{aligned}$$

Then, we have

$$
\begin{aligned}
&\mathbb{E}\left[\tilde{\mathbf{z}}_{i'}^{\intercal}\mathbf{Q}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^{\intercal}\mathbf{Q}\tilde{\mathbf{z}}_{i''}\right]\\
&=\frac{\mathbf{e}_{t',j'}^{[2T]\intercal}\bar{\mathbf{M}}^{\intercal}\mathbb{E}\left[\mathbf{Q}\mathbf{C}_j^t\mathbf{Q}\right]\bar{\mathbf{M}}\mathbf{e}_{t'',j''}^{[2T]}}{(1-\delta^{t'})(1-\delta^t)^2(1-\delta^{t''})}-\frac{1}{n^{t'}}\frac{\mathbb{E}\left[\mathrm{Tr}(\mathbf{Q}\mathbf{C}_j^t\mathbf{Q}\mathbf{C}^{t'}\right]\mathbf{e}_{t',j'}^{[2T]\intercal}\mathbf{\Gamma}\mathbf{e}_{t'',j''}^{[2T]}}{(1-\delta^{t'})(1-\delta^t)^2(1-\delta^{t''})}\\
&+\frac{\mathbf{e}_{t',j'}^{[2T]\intercal}(\mathbf{\Theta}-\gamma^t\mathbf{\Gamma})\mathbf{e}_{t'',j''}^{[2T]}}{Tp(1-\delta^{t'})^2(1-\delta^t)^2(1-\delta^{t''})}\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t\mathbf{Q}\mathbf{C}^{t'}\right)\right]\\
&+\frac{\mathbf{e}_{t',j'}^{[2T]\intercal}(\mathbf{\Theta}-\gamma^t\mathbf{\Gamma})\mathbf{e}_{t'',j''}^{[2T]}}{Tp(1-\delta^{t'})(1-\delta^t)^2(1-\delta^{t''})^2}\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{Q}\mathbf{C}^t\mathbf{Q}\mathbf{C}^{t''}\right)\right]\\
&=\frac{\mathbf{e}_{t',j'}^{[2T]\intercal}}{1-\delta^{t'}}\underbrace{\frac{\bar{\mathbf{M}}^{\intercal}\mathbb{E}\left[\mathbf{Q}\mathbf{C}_j^t\mathbf{Q}\right]\bar{\mathbf{M}}}{(1-\delta^t)^2}}_{:=\boldsymbol{\mathcal{U}}_1^t}\frac{\mathbf{e}_{t'',j''}^{[2T]}}{1-\delta^{t''}}-Tc\mathbf{S}^{tt'}\mathbf{e}_{t',j'}^{[2T]\intercal}\underbrace{\frac{\mathbf{\Gamma}}{(1-\delta^t)^2}}_{:=\boldsymbol{\mathcal{U}}_2^t}\frac{Tc}{\rho^{t'}}\mathbf{e}_{t'',j''}^{[2T]}\\
&+Tc\mathbf{e}_{t',j'}^{[2T]\intercal}\left(\underbrace{\frac{(\mathbf{\Theta}-\gamma^t\mathbf{\Gamma})}{(1-\delta^t)^2}}_{:=\boldsymbol{\mathcal{U}}_3^t}\frac{\mathbf{S}^{tt'}}{1-\delta^{t''}}+\frac{\mathbf{S}^{tt''}}{1-\delta^{t'}}\boldsymbol{\mathcal{U}}_3^t\right)\mathbf{e}_{t'',j''}^{[2T]}.
\end{aligned}
$$

Therefore:

$$
\begin{aligned}
C_2&=\left(\sum_{i'}\frac{y_{i'}}{Tp}\frac{\mathbf{e}_{t',j'}^{[2T]}}{1-\delta^{t'}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_1^t\left(\sum_{i''}\frac{y_{i''}}{Tp}\frac{\mathbf{e}_{t'',j''}^{[2T]}}{1-\delta^{t''}}\right)-\left(\sum_{i'}\frac{y_{i'}}{n}\mathbf{e}_{t',j'}^{[2T]}\mathbf{S}^{tt'}\right)^{\intercal}\boldsymbol{\mathcal{U}}_2^t\left(\sum_{i''}\frac{y_{i''}}{n_{j''}^{t''}}\mathbf{e}_{t'',j''}^{[2T]}\right)\\
&+\left(\sum_{i'}\frac{y_{i'}}{n}\mathbf{e}_{t',j'}^{[2T]}\mathbf{S}^{tt'}\right)^{\intercal}\boldsymbol{\mathcal{U}}_3^t\left(\sum_{i''}\frac{y_{i''}}{Tp}\frac{\mathbf{e}_{t'',j''}^{[2T]}}{1-\delta^{t''}}\right)+\left(\sum_{i'}\frac{y_{i'}}{Tp}\frac{\mathbf{e}_{t',j'}^{[2T]}}{1-\delta^{t'}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_3^t\left(\sum_{i''}\frac{y_{i''}}{n}\mathbf{e}_{t'',j''}^{[2T]}\mathbf{S}^{tt''}\right)\\
&=\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_1^t\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)-\left(\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_2^t\left(\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)\\
&+\left(\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_3^t\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)+\left(\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)^{\intercal}\boldsymbol{\mathcal{U}}_3^t\left(\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}\right)\\
&=\tilde{\mathbf{y}}^{\intercal}\bar{\mathbf{D}}^{\intercal}\mathcal{D}_{\boldsymbol{\eta}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\boldsymbol{\mathcal{U}}_1^t\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}-\tilde{\mathbf{y}}^{\intercal}\bar{\mathbf{D}}^{\intercal}\mathcal{D}_{\boldsymbol{\eta}}\boldsymbol{\mathcal{U}}_2^t\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}+2\tilde{\mathbf{y}}^{\intercal}\bar{\mathbf{D}}^{\intercal}\mathcal{D}_{\boldsymbol{\eta}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\boldsymbol{\mathcal{U}}_3^t\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}
\end{aligned}
$$

Let us decompose

$$
\begin{aligned}
&\bar{\mathbf{M}}^{\intercal}\bar{\mathbf{Q}}_0\mathbf{C}_j^t\bar{\mathbf{Q}}_0\bar{\mathbf{M}}\\
&=\bar{\mathbf{M}}^{\intercal}\bar{\mathbf{Q}}_0\bar{\mathbf{M}}\mathbf{e}_{t,j}^{[2T]}\mathbf{e}_{t,j}^{[2T]\intercal}\bar{\mathbf{M}}^{\intercal}\bar{\mathbf{Q}}_0\bar{\mathbf{M}}+\bar{\mathbf{M}}^{\intercal}\bar{\mathbf{Q}}_0\mathbf{A}^{\frac{1}{2}}(\mathbf{E}_{tt}\otimes\mathbf{I}_p)\mathbf{A}^{\frac{1}{2}}\bar{\mathbf{Q}}_0\bar{\mathbf{M}}\\
&=\mathbf{\Theta}_0\mathbf{e}_{t,j}^{[2T]}\mathbf{e}_{t,j}^{[2T]\intercal}\mathbf{\Theta}_0+\left(\underbrace{\boldsymbol{\mathcal{A}}\mathbf{E}_{tt}\boldsymbol{\mathcal{A}}}_{:=\boldsymbol{\mathcal{V}}^t}\otimes\mathbb{1}_2\mathbb{1}_2^{\intercal}\right)\odot\mathbf{M}^{\intercal}\mathbf{M}\\
&=\mathbf{\Theta}_0\mathbf{e}_{t,j}^{[2T]}\mathbf{e}_{t,j}^{[2T]\intercal}\mathbf{\Theta}_0+\underbrace{\left(\boldsymbol{\mathcal{V}}^t\otimes\mathbb{1}_2\mathbb{1}_2^{\intercal}\right)\odot\mathbf{M}^{\intercal}\mathbf{M}}_{:=\mathbf{\Omega}_0^t}.
\end{aligned}
$$

Similarly,

$$\begin{aligned}
&\bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}_0 \left(\mathbf{C}_j^t + \sum_{t',j'} \rho_{j'}^{t'} \mathbf{S}^{tt'} \mathbf{C}_{j'}^{t'} \right) \bar{\mathbf{Q}}_0 \bar{\mathbf{M}} \\
&= \boldsymbol{\Theta}_0 \left(\mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} + \mathcal{D}_{\mathbf{r}^t} \right) \boldsymbol{\Theta}_0 + \underbrace{\left(\bar{\boldsymbol{\mathcal{V}}}^t \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal \right) \odot \mathbf{M}^\intercal \mathbf{M}}_{:= \bar{\boldsymbol{\Omega}}_0^t},
\end{aligned}$$

with $\mathbf{r}^t = \boldsymbol{\rho} \odot (\mathbf{S}^t \otimes \mathbb{1}_2 \mathbb{1}_2^\intercal)$ and $\bar{\boldsymbol{\mathcal{V}}}^t = \boldsymbol{\mathcal{V}}^t + \sum_{t'} \rho^{t'} \mathbf{S}^{tt'} \boldsymbol{\mathcal{V}}^{t'}$. Using the previous results, we have:

$$\begin{aligned}
&\bar{\mathbf{M}}^\intercal \mathbb{E}[\mathbf{Q} \mathbf{C}_j^t \mathbf{Q}] \bar{\mathbf{M}} = \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}} \left(\mathbf{C}_j^t + \sum_{t',j'} \rho_{j'}^{t'} \mathbf{S}^{tt'} \mathbf{C}_{j'}^{t'} \right) \bar{\mathbf{Q}} \bar{\mathbf{M}} \\
&= \boldsymbol{\Theta} \left(\mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} + \mathcal{D}_{\mathbf{r}^t} \right) \boldsymbol{\Theta} + \underbrace{\left(\mathbf{I}_{2T} - \boldsymbol{\Theta}_0 \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \right)^{-1} \bar{\boldsymbol{\Omega}}_0^t \left(\mathbf{I}_{2T} - \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \boldsymbol{\Theta}_0 \right)^{-1}}_{:= \bar{\boldsymbol{\Omega}}^t}.
\end{aligned}$$

Going back to C_2,

$$\begin{aligned}
C_2 = \frac{1}{(1-\delta^t)^2} \tilde{\mathbf{y}}^\intercal \bar{\mathbf{D}}^\intercal \mathcal{D}_{\boldsymbol{\eta}} \Big[&\mathcal{D}_{\tilde{\boldsymbol{\delta}}} \boldsymbol{\Theta} \left(\mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} + \mathcal{D}_{\mathbf{r}^t} \right) \boldsymbol{\Theta} \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \\
&+ \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \bar{\boldsymbol{\Omega}}^t \mathcal{D}_{\tilde{\boldsymbol{\delta}}} - \boldsymbol{\Gamma}^t \mathcal{D}_{\mathbf{r}^t} + 2 \mathcal{D}_{\tilde{\boldsymbol{\delta}}} (\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma}) \mathcal{D}_{\mathbf{r}^t} \Big] \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}}.
\end{aligned}$$

We recall that

$$(m_j^t)^2 = \frac{1}{(1-\delta^t)^2} \tilde{\mathbf{y}}^\intercal \bar{\mathbf{D}}^\intercal \mathcal{D}_{\boldsymbol{\eta}} \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \boldsymbol{\mathcal{U}} \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}},$$

with

$$\begin{aligned}
\boldsymbol{\mathcal{U}} &= \left(\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma} \right) \mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} \left(\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma} \right) \\
&= \boldsymbol{\Theta} \mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} \boldsymbol{\Theta} - \gamma^{t2} \boldsymbol{\Gamma} \mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} \boldsymbol{\Gamma} \\
&= \boldsymbol{\Theta} \mathbf{e}_{t,j}^{[2T]} \mathbf{e}_{t,j}^{[2T]\intercal} \boldsymbol{\Theta} - \gamma^{t2} \underbrace{\mathbf{E}^{tt} \otimes (\mathbb{1}_2 \mathbb{1}_2^\intercal)}_{:= \boldsymbol{\Gamma}^t}.
\end{aligned}$$

$$\begin{aligned}
C_2 - {m_j^t}^2 = &\frac{1}{(1-\delta^t)^2} \tilde{\mathbf{y}}^\intercal \bar{\mathbf{D}}^\intercal \mathcal{D}_{\boldsymbol{\eta}} \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \left(\boldsymbol{\Theta} \mathcal{D}_{\mathbf{r}^t} \boldsymbol{\Theta} + \bar{\boldsymbol{\Omega}}^t - \gamma^{t2} \boldsymbol{\Gamma}^t \right) \mathcal{D}_{\tilde{\boldsymbol{\delta}}} \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}} \\
&+ \frac{1}{(1-\delta^t)^2} \tilde{\mathbf{y}}^\intercal \bar{\mathbf{D}}^\intercal \mathcal{D}_{\boldsymbol{\eta}} \left[2 \mathcal{D}_{\tilde{\boldsymbol{\delta}}} (\boldsymbol{\Theta} - \gamma^t \boldsymbol{\Gamma}) - \boldsymbol{\Gamma}^t \right] \mathcal{D}_{\mathbf{r}^t} \mathcal{D}_{\boldsymbol{\eta}} \bar{\mathbf{D}} \tilde{\mathbf{y}}
\end{aligned}$$

As $Var(q_i) = C_1 + C_2 - {m_j^t}^2$, we finally have:

$$\sigma^{t2} = \frac{1}{(1-\delta^t)^2}\tilde{\mathbf{y}}^\intercal\left(\mathbf{T}^t \odot \tilde{\mathbf{D}}\right)\tilde{\mathbf{y}} + \frac{1}{(1-\delta^t)^2}\tilde{\mathbf{y}}^\intercal\bar{\mathbf{D}}^\intercal\mathcal{D}_{\boldsymbol{\eta}}\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\left(\boldsymbol{\Theta}\mathcal{D}_{\mathbf{r}^t}\boldsymbol{\Theta} + \bar{\boldsymbol{\Omega}}^t - {\gamma^t}^2\boldsymbol{\Gamma}^t\right)\mathcal{D}_{\tilde{\boldsymbol{\delta}}}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}$$
$$+ \frac{1}{(1-\delta^t)^2}\tilde{\mathbf{y}}^\intercal\bar{\mathbf{D}}^\intercal\mathcal{D}_{\boldsymbol{\eta}}\left[2\mathcal{D}_{\tilde{\boldsymbol{\delta}}}(\boldsymbol{\Theta} - \gamma^t\boldsymbol{\Gamma}) - \boldsymbol{\Gamma}^t\right]\mathcal{D}_{\mathbf{r}^t}\mathcal{D}_{\boldsymbol{\eta}}\bar{\mathbf{D}}\tilde{\mathbf{y}}.$$

A.5.5 Computation of the correlation between scores

$$\mathbf{E}[f_{i_1}f_{i_2}] = \frac{1}{(Tp)^2}\sum_{i'_1,i'_2}\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right]y_{i'_1}y_{i'_2}$$

The term $\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right]$ has the same order of magnitude $O(1)$ no matter if $i'_1 = i'_2$ or $i'_1 \neq i'_2$. As there are n couples of identical indices over the total number of couples n^2, we can neglect the couples of identical indices:

$$\mathbf{E}[f_{i_1}f_{i_2}] = \frac{1}{(Tp)^2}\sum_{i'_1\neq i'_2}\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right]y_{i'_1}y_{i'_2}.$$

For f_{i_1} and f_{i_2} to be uncorrelated, we need to have $\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right] = \mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\right]\mathbb{E}\left[\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right]$. Using successively equations (A.7) and (A.8),

$$\begin{aligned}
&(1-\delta^{t_1})(1-\delta^{t'_1})(1-\delta^{t_2})(1-\delta^{t'_2})\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right]\\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}_{-i_1,i'_1}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}_{-i_2,i'_2}\tilde{\mathbf{z}}_{i_2}\right]\\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}_{-i_2,i'_2}\tilde{\mathbf{z}}_{i_2}\right] \qquad \text{(A.12)}\\
&+ \frac{1}{Tp}\frac{1}{1-\delta^{t_2}}\mathbb{E}[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i_2}\tilde{\mathbf{z}}_{i_2}^\intercal\underbrace{\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i'_1}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}_{-i_2,i'_2}}_{\mathbf{B}}\tilde{\mathbf{z}}_{i_2}]
\end{aligned}$$

As $\mathbf{B}$ and $\tilde{\mathbf{z}}_{i_2}$ are independant, we can apply lemma 38 inside the expectation of the second term, which then rewrites as:

$$\begin{aligned}
&\frac{1}{Tp}\frac{1}{1-\delta^{t_2}}\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i'_2}\,\mathrm{Tr}\left(\mathbf{C}_{j_2}^{t_2}\mathbf{B}\right)\right]\\
&= \frac{1}{Tp}\frac{1}{1-\delta^{t_2}}\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i'_2}\tilde{\mathbf{z}}_{i'_2}^\intercal\mathbf{Q}_{-i_2,i'_2}\mathbf{C}_{j_2}^{t_2}\mathbf{Q}_{-i_1,i'_1,i_2}\tilde{\mathbf{z}}_{i'_1}\right]
\end{aligned}$$

We see that the second term of (A.12) is of order $O(\frac{1}{p})$, while the first term is of order $O(1)$. Therefore this second term is negligeable, which means that we can

replace $\mathbf{Q}_{-i_1,i_1'}$ by $\mathbf{Q}_{-i_1,i_1',i_2}$ without consequences. More generally, we can remove the indices i_2, i_2' of $\mathbf{Q}_{-i_1,i_1'}$ and the indices i_1, i_1' of $\mathbf{Q}_{-i_2,i_2'}$:

$$\begin{aligned}
&(1-\delta^{t_1})(1-\delta^{t_1'})(1-\delta^{t_2})(1-\delta^{t_2'})\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_1'}\tilde{\mathbf{z}}_{i_2'}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right] \\
&= \mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal \mathbf{Q}_{-i_1,i_1',i_2,i_2'}\tilde{\mathbf{z}}_{i_1'}\tilde{\mathbf{z}}_{i_2'}^\intercal \mathbf{Q}_{-i_1,i_1',i_2,i_2'}\tilde{\mathbf{z}}_{i_2}\right] \\
&= \mathbb{E}\left[\mathbf{e}_{t_1}^\intercal \bar{\mathbf{M}}^\intercal \mathbf{Q}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \mathbf{Q}\bar{\mathbf{M}}\mathbf{e}_{t_2}\right].
\end{aligned}$$

Furthermore, the equivalent of equation (A.11) gives:

$$\begin{aligned}
&\mathbf{Q}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \mathbf{Q} \leftrightarrow \bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}} \\
&+ \sum_{t',j'} \frac{n_{j'}^{t'}}{(Tp)^2(1-\delta^{t'})^2}\operatorname{Tr}\left(\bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}\mathbf{C}_{j'}^{t'}\right)\mathbf{Q}\mathbf{C}_{j'}^{t'}\mathbf{Q}
\end{aligned}$$

Once again, the second term is negligeable because the trace is of order $O(1)$, so the whole term is of order $O(\frac{1}{p})$, and we have:

$$\begin{aligned}
&\mathbb{E}\left[\mathbf{e}_{t_1}^\intercal \bar{\mathbf{M}}^\intercal \mathbf{Q}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \mathbf{Q}\bar{\mathbf{M}}\mathbf{e}_{t_2}\right] \\
=&\mathbb{E}\left[\mathbf{e}_{t_1}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_2}\right] \\
=&\mathbf{e}_{t_1}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_1'}\mathbf{e}_{t_2'}^\intercal \bar{\mathbf{M}}^\intercal \bar{\mathbf{Q}}\bar{\mathbf{M}}\mathbf{e}_{t_2}.
\end{aligned}$$

Finally, we have

$$\mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_1'}\tilde{\mathbf{z}}_{i_2'}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right] = \mathbb{E}\left[\tilde{\mathbf{z}}_{i_1}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_1'}\right]\mathbb{E}\left[\tilde{\mathbf{z}}_{i_2'}^\intercal \mathbf{Q}\tilde{\mathbf{z}}_{i_2}\right],$$

which proves uncorrelatedness.

A.6 Perturbation of the overlap equations

Let $(q_u^\star + \Delta q_u, q_v^\star + \Delta q_v)$ a solution of equation (6.4), where $(q_u^\star, q_v^\star)$ is the solution of the system of equations from Theorem 29. Δq_v represents the approximation error of $q_v^\star$ through the approximation $\tilde{F}_\varepsilon(q_v) \simeq F_\varepsilon(q_v)$. To make sure that this approximation does not jeopardize the solution of the system of equations of Theorem 29, we are interested in computing the quantity $|\frac{\Delta q_u}{q_u^\star}|$ as a function $|\frac{\Delta q_v}{q_v^\star}|$. To simplify the expression of q_u, we will denote $\tilde{\lambda} = \frac{\lambda}{c}$.

If $q_u = q_u^\star + \Delta q_u$ is the solution of (6.4) with $q_v = q_v^\star + \Delta q_v$,

$$\begin{aligned} q_u &= \lambda \frac{\tilde{\lambda}(q_v^\star + \Delta q_v)}{1 + \tilde{\lambda}(q_v^\star + \Delta q_v)} \\ &= \lambda \frac{\tilde{\lambda}(q_v^\star + \Delta q_v)}{1 + \tilde{\lambda} q_v^\star} \frac{1}{1 + \underbrace{\frac{\tilde{\lambda}\Delta q_v}{1 + \tilde{\lambda} q_v^\star}}_{:=u}} \\ &= (q_u^\star + \lambda u)\frac{1}{1+u} \\ &= q_u^\star + (\lambda - q_u^\star)\frac{u}{1+u} \\ &= q_u^\star + \underbrace{(\lambda - q_u^\star)\frac{\tilde{\lambda}\Delta q_v}{1 + \tilde{\lambda} q_v}}_{=\Delta q_u}. \end{aligned}$$

Indeed,

$$\frac{u}{1+u} = \frac{\frac{\tilde{\lambda}\Delta q_v}{1+\tilde{\lambda} q_v^\star}}{1 + \frac{\tilde{\lambda}\Delta q_v}{1+\tilde{\lambda} q_v^\star}} = \frac{\tilde{\lambda}\Delta q_v}{1 + \tilde{\lambda} q_v^\star + \tilde{\lambda}\Delta q_v} = \frac{\tilde{\lambda}\Delta q_v}{1 + \tilde{\lambda} q_v}.$$

Going back to the quantity of interest,

$$\frac{\Delta q_u}{q_u^\star} = \frac{\left(1 - \frac{\tilde{\lambda} q_v^\star}{1+\tilde{\lambda} q_v^\star}\right)\frac{\tilde{\lambda}\Delta q_v}{1+\tilde{\lambda} q_v}}{\frac{\tilde{\lambda} q_v^\star}{1+\tilde{\lambda} q_v^\star}} = \frac{\frac{1}{1+\tilde{\lambda} q_v^\star}\frac{\tilde{\lambda}\Delta q_v}{1+\tilde{\lambda} q_v}}{\frac{\tilde{\lambda} q_v^\star}{1+\tilde{\lambda} q_v^\star}} = \frac{1}{1+\tilde{\lambda} q_v}\frac{\Delta q_v}{q_v^\star}.$$

As $|\frac{1}{1+\tilde{\lambda} q_v}| \leq 1$, we conclude that $|\frac{\Delta q_u}{q_u^\star}| \leq |\frac{\Delta q_v}{q_v^\star}|$.