Deciphering Lasso-based Classification Through a Large Dimensional Analysis of the Iterative Soft-Thresholding Algorithm

Malik Tiomoko [1] Ekkehard Schnoor [2] Mohamed El Amine Seddik [3] Igor Colin [1] Aladin Virmaux [1]

Abstract

This paper proposes a theoretical analysis of a Lasso-based classification algorithm. Leveraging on a realistic regime where the dimension of the data p and their number n are of the same order of magnitude, the theoretical classification error is derived as a function of the data statistics. As a result, insights into the functioning of the Lasso in classification and its differences with competing algorithms are highlighted. Our work is based on an original novel analysis of the iterative soft-thresholding algorithm (ISTA), which may be of independent interest beyond the particular problem studied here and may be adapted to similar iterative schemes. A theoretical optimization of the model's hyperparameters is also provided, which allows for the data- and time-consuming cross-validation to be avoided. Finally, several applications on synthetic and real data are provided to validate the theoretical study and justify its impact in the design and understanding of algorithms of practical interest.

1. Introduction

The Lasso. The Lasso (Least Absolute Shrinkage and Selection Operator, (Tibshirani, 1996)) is one of the most well-known tools in statistics and signal processing. By employing ℓ_1-regularization, it imposes sparsity on the solution sought by selecting only a limited number of features of interest for the task at hand. Furthermore, the Lasso and its various related variants are of interest for machine learning, particularly in the field of classification when only a few features are relevant for predicting the class memberships.

On the theoretical analysis of the Lasso. Although the Lasso is a classical and widespread tool, its precise performance, strengths and limitations for classification tasks are subject to surprisingly few theoretical studies. In particular, the choice of the hyperparameter (*i.e.,* the regularization parameter) has, for the time being, remained restricted to a cross validation which may be time and data consuming. Furthermore, the difficulty of predicting in advance the performance obtained by the classification algorithm and the lack of statistical interpretation make it data-dependent. In this article, we address these issues from a theoretical perspective through a novel analysis of ISTA, which may be extended to similar iterative algorithms. Specifically, based on a mixture of concentrated random vector assumption on the data, among which we find images generated by Generative Adversarial Nets (GANs), and considering a statistical learning regime where the dimension of the data evolves linearly with the number of samples, this paper derives the exact theoretical classification error as a function of the underlying model parameters and the data statistics. Even though the theoretical analysis is performed in an asymptotic setting, we confirm our findings on both real and synthetic datasets of finite sample and feature size, thereby allowing their application also to situations of practical interest.

State of the art. Although widely used in practice, the theoretical study of Lasso has been subject to very few studies on the exact performance characterization in large dimensions. Much of the literature has focused on the related problem of Compressive Sensing, where one aims to reconstruct sparse vectors from only few linear measurements (Candès & Tao, 2005; Candès et al., 2006), typically providing non-asymptotic bounds (e.g. on the necessary number of measurements). Our study complements this theory by providing not bounds but exact performances. Regarding the exact performance, large areas of literature have focused on risk analysis in the large random matrices regime (where the dimension p and the number of data n are of the same order of magnitude): Approximate Message Passing (Bayati & Montanari, 2011) and analysis based on the Convex Gaussian Minmax Theorem (Thrampoulidis et al., 2015). These two approaches, although very robust and theoretically stable, are restricted to regression cases

[1] Huawei Noah's Ark Lab, Paris, France [2] Chair for Mathematics of Information Processing, RWTH Aachen University, Germany [3] Mathematical and Algorithmic Sciences Laboratory, Huawei Technologies France. Correspondence to: Malik Tiomoko <malik.tiomoko@huawei.com>, Ekkehard Schnoor <schnoor@mathc.rwth-aachen.de>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

and the theoretical formulas obtained are generally difficult to interpret given a classification problem. In this work, we are more interested in the latter while providing a set of intuitions for the Lasso-based classification algorithms. We should interestingly mention that ISTA, being an iterative process, can likely be analyzed theoretically also in a **classification** context using Statistical Physics techniques in particular Approximate Message Passing (AMP) beyond the regression case predominant in literature. However, the tools required by AMP (Gerbelot & Berthier, 2021; Mezard & Montanari, 2009; Baker et al., 2020) and the one used in this article (Random Matrix Theory and in particular the *leave-one-out approach*), although being similar, have differences. The two methods are indeed conceptually similar in the considered regime (commensurable dimension and sample size) and their common objective (large dimensional analysis). However, the two methods have differences, notably concerning the assumptions on the data (concentrated random vector in the case considered in the present article and Gaussian in the AMP case). Furthermore, the analytical tools (deterministic equivalent in particular used in Random Matrix Theory) do not have explicit equivalents. The universality of the random matrix results for the distribution of the data combined with the intuitive interpretation of the cavity method's proof make them two different but complementary methods. Despite the differences between the two methods, it is possible to establish strong interconnections, a line of research that we find extremely important, but goes beyond the study of the Lasso (but is also valid for the comparison of studies on linear models in general, among others).

Our contributions. Therefore, the main contributions brought by this work are listed as follows:

- The exact classification error of the Lasso problem is theoretically derived using the so-called *leave-one-out approach* to handle the strong statistical dependencies.
- As a consequence, insights into the Lasso-based classification are obtained. We propose a new approach to hyperparameter optimization avoiding cross-validation.
- A range of applications is proposed to attest the relevance of the theoretical study and the robustness of the concentrated random vector assumptions on real data.

Outline of the paper. The paper is structured as follows. In Section 2, the Lasso-based classification algorithm is introduced and the data modelling for the high-dimensional analysis is further presented. Section 3 provides our main theoretical findings and some insights into the functioning of the Lasso-based classification algorithm from a statistical perspective. Section 4 proposes one application of the theoretical result in the context of hyperparameter selection.

Related work. Sparse representations have attracted much attention from researchers in the areas of signal processing, image processing, computer vision and pattern recognition (Mallat, 1999; Elad et al., 2010; Lu & Li, 2014; Foucart & Rauhut, 2013; Baraniuk et al., 2011), with their high potential to represent some phenomenon with as few variables as possible. Several papers (Donoho, 2006; Candès et al., 2006) laid the foundation of Compressive Sensing by demonstrating the rationale of reconstructing sparse signals from fewer samples than required by Shannon (Shannon, 1948). Recently, sparse representations received a lot of attention from the machine learning community, and the Lasso has been extended to network structures (Hallac et al., 2015; Jung et al., 2018).

Sparse linear classifiers have been studied from the statistical learning perspective, but based on VC dimension bounds, previously in (Sabato et al., 2015) and, for sparse logistic regression, in (Abramovich & Grinshtein, 2018). A mathematical framework for features selection from real-world data with non-linear observations, also from a non-asymptotic viewpoint, is provided in (Genzel & Kutyniok, 2016).

Lasso penalized regression (Tibshirani, 1996) has been successful in ignoring irrelevant predictors in a regression problem. Some extensions for classification have been proposed in (Diamond & Boyd, 2016; Lee et al., 2006; El Ghaoui et al., 2010; Musa, 2014; Koh et al., 2007; Meier et al., 2008; Van de Geer, 2008). Lasso penalized estimation raised the question of the optimal choice of the hyperparameter which promotes sparsity of the model in practical situations. The standard solution of cross-validation is computational expensive, therefore asking for the design of a proper and reliable cross validation scheme.

Yet, a large and rapidly growing literature (Candès et al., 2006; Beck & Teboulle, 2009) is devoted to developing fast algorithms for solving the Lasso optimization problem. However, despite countless theoretical efforts, the understanding of the Lasso in the context of high dimensional statistics remains rather lacking. The authors in (Candès et al., 2006) have derived upper bound for the reconstruction of a sparse signal providing then guarantees for the Lasso or similar convex optimization methods. Work by Candès & Tao (2007) on the analogous Dantzig selector proposed an upper bound on the reconstruction error to within one constant. With the rise of deep learning, unfolding algorithms like ISTA as neural networks has become a popular research area (Gregor & LeCun, 2010). Recently, this approach has been investigated from a statistical learning perspective as well (Behboodi et al., 2021; Schnoor et al., 2021), but focussing on reconstruction (*i.e.,* a regression problem), and using different tools (generalization error bounds based on the Rademacher complexity), in a non-asymptotic setting.

Based on techniques from Approximate Message Passing, (Bayati & Montanari, 2011; Huang, 2020; Gerbelot et al., 2020; Celentano et al., 2020) derive exact asymptotic expressions for the reconstruction error. These works have been complemented by an analysis using the Convex Gaussian Min-max Theorem (Thrampoulidis et al., 2015; Alrashdi et al., 2020). The present paper is part of this line of work employing an asymptotic analysis of the Lasso. However, unlike previous works, we are interested in a different setting, that of classification, and we propose to derive the analytical expression of the classification error using a sparsity *a priori* on the separating hyperplane. Furthermore, we use different tools, namely the powerful *leave-one-out approach* (Chen et al., 2019; El Karoui et al., 2013; Ding & Chen, 2018).

From a technical point of view, our work is similar to the analysis of machine learning algorithms, among which we find the high dimensional analysis of logistic regression (Mai et al., 2019; El Karoui et al., 2013), support vector machine (Mai & Liao, 2019; Mai & Couillet, 2018) and more recently Softmax classifier (Seddik et al., 2021). However, unlike previous studies, the difficulty of the Lasso lies in the non-differentiability of the cost function and the complex iterative procedure used to solve the minimization problem. From this point of view, the technical difficulty inherent in the study of the Lasso is intrinsically more challenging.

Reproducibility. Python codes for reproducing the results of the paper are available in the supplementary materials.

Notation. Matrices will be represented by bold capital letters (*i.e.,* matrix $\mathbf{A}$). Vectors will be represented in bold minuscule letters (*i.e.,* vector $\mathbf{v}$) and scalars will be represented without bold letters (*i.e.,* variable a). The index pair i, ℓ refers to any data sample i in the class ℓ. Furthermore, the jth ISTA iteration will be denoted by a superscript j. (See details below.) The notation $\mathcal{D}(\mathbf{A})$ for a matrix $\mathbf{A}$ is a vector containing the diagonal elements of $\mathbf{A}$. The notation $\bar{\mathbf{A}}$ for random matrices (and random vectors) stands for $\bar{\mathbf{A}} = \mathbb{E}[\mathbf{A}]$. $\mathbb{1}_n \in \mathbb{R}^n$ is the vector of all ones. For a random vector $\mathbf{v} \in \mathbb{R}^n$, the matrix $\boldsymbol{\Sigma}_\mathbf{v} \in \mathbb{R}^{n\times n}$ denotes its covariance matrix. Specifically, $\boldsymbol{\Sigma}_\mathbf{v} = \mathrm{Cov}(\mathbf{v}) = \mathbb{E}\left[(\mathbf{v}-\bar{\mathbf{v}})(\mathbf{v}-\bar{\mathbf{v}})^\mathsf{T}\right]$, and we will use the shortcut notation $\boldsymbol{\sigma}_\mathbf{v} = \mathcal{D}(\boldsymbol{\Sigma}_\mathbf{v})$ to represent the diagonal elements of the covariance matrix $\boldsymbol{\Sigma}_\mathbf{v}$. For $p \geq 1$, $\| \,.\, \|_p$ denotes the ℓ_p-norm for vectors. The zero vector in $\mathbb{R}^p$ is denoted by $\mathbf{0}_p$, and the identity matrix by $\mathbf{I}_p$.

2. Model and Assumptions

2.1. Lasso Classification Formalism

Optimization problem. Suppose we have n data samples $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ gathered as columns in the data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$. The data matrix consists of two (nonempty) data classes $\mathcal{C}_1$ and $\mathcal{C}_2$ corresponding to the labels ± 1, *i.e.,* $\mathbf{X} = [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}]$ where $\mathbf{X}^{(\ell)} = [\mathbf{x}_1^{(\ell)}, \ldots, \mathbf{x}_{n_\ell}^{(\ell)}]$, $\ell = 1, 2$, and $n_1 + n_2 = n$. Each data point $\mathbf{x}_i^{(\ell)}$ is associated with its label $\mathbf{y}_i^{(\ell)}$, and we denote by $\mathbf{y} = [y_1^{(1)}, \ldots, y_{n_1}^{(1)}, y_1^{(2)}, \ldots, y_{n_2}^{(2)}]^\mathsf{T} \in \{-1, 1\}^n$ the vector containing all labels. Given a new test datum $\mathbf{x}$, the goal is to predict its associated label $\mathbf{y}$ using Lasso regression as follows. We aim to find the best separating hyperplane parametrized by $\boldsymbol{\omega}^\star \in \mathbb{R}^p$ for which the training classification error $\|\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}\|_2^2$ is minimized, but Lasso adds an ℓ_1-constraint (or regularization) on $\boldsymbol{\omega}^\star$. Formally, it means to solve the following minimization problem given by

$$\boldsymbol{\omega}^\star = \underset{\boldsymbol{\omega}\in\mathbb{R}^p}{\arg\min} \, \frac{1}{2}\|\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}\|_2^2 \quad \text{s.t.} \quad \|\boldsymbol{\omega}\|_1 \leq \lambda', \tag{1}$$

for some $\lambda' > 0$. Or, equivalently, for an appropriate choice of the regularization parameter $\lambda > 0$, the Lasso problem boils down to the mere ℓ_1-regularized least-squares problem

$$\boldsymbol{\omega}^\star = \underset{\boldsymbol{\omega}\in\mathbb{R}^p}{\arg\min} \, \frac{1}{2}\|\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}\|_2^2 + \lambda\|\boldsymbol{\omega}\|_1. \tag{2}$$

We should interestingly mention that we use on purpose a least square loss for the label fidelity term ($\frac{1}{2}\|\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}\|_2^2$). One could have used other loss functions (*e.g.,* hinge or logistic loss, again with an ℓ_1-regularization) at the cost of more complicated analysis. Indeed, the asymptotic performances to be set for improved classification have simple expressions under a least square framework. This ensures a full control on the performances and a better understanding of the proposed approach. The analysis of a loss hinge or logistics is possible using the same framework by combining the approach proposed in this article and previous studies that propose an asymptotic analysis of the hinge and logistic loss (Mai et al., 2019; Mai & Liao, 2019; Kammoun & Alouini, 2021) in a framework of a ℓ_2-regularization. This study, however, will present greater complexity and formulas that will be less easy to interpret what justifies the choice of the restriction to a least square loss. Moreover, by a now well-established universality argument in large dimensional statistics, it has been shown in closely related works (Mai & Liao, 2019) that quadratic cost functions are asymptotically optimal (as the data dimension and number increase) and uniformly outperform alternative costs (such as SVM or logistic approaches), even in a classification setting (the proof was in fact obtained in the precise large dimensional setting which we consider here). We have confirmed in Section C.5 of the supplementary materials this observation on several datasets.

Given the optimal separating hyperplane $\boldsymbol{\omega}^\star$, the classification is traditionally performed by considering the sign of $g(\mathbf{x}) = {\boldsymbol{\omega}^\star}^\mathsf{T}\mathbf{x}$. Intuitively, $\boldsymbol{\omega}^\star$ will encode the important features for the classification trained from the training set $\mathbf{X}$ and will infer, through the projection ${\boldsymbol{\omega}^\star}^\mathsf{T}\mathbf{x}$, the class of a new test datum $\mathbf{x}$ relying solely on the selected features. The difficulties inherent to the statistical analysis of the Lasso problem lie in the lack of an explicit expression for $\boldsymbol{\omega}^\star$, and more importantly, in the infeasibility of gradient-based methods since the ℓ_1-norm is not differentiable. To bypass this problem, several algorithms have been designed among which the iterative soft-thresholding algorithm (ISTA) plays an important role and will be employed by us.

Iterative soft-thresholding algorithm. For a sparse minimization of the differentiable function $h(\boldsymbol{\omega}) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}\|_2^2$, ISTA is the iterative algorithm, starting with an initialization of typically $\boldsymbol{\omega}^0 = \mathbf{0}_p$, and for with $j \geq 1$,

$$\begin{aligned} &\textbf{Gradient step: } \mathbf{z}^j = \boldsymbol{\omega}^{j-1} - \tau \nabla h\left(\boldsymbol{\omega}^{j-1}\right), \\ &\textbf{Sparsity step: } \boldsymbol{\omega}^j = S_{\tau\lambda}\left(\mathbf{z}^j\right), \end{aligned} \tag{3}$$

with τ the step size and $S_{\tau\lambda}$ the soft threshold function defined below. This leads to the following iterative procedure

$$\boldsymbol{\omega}^{j+1} = S_{\tau\lambda}\left[\boldsymbol{\omega}^j + \tau\mathbf{X}\left(\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}^j\right)\right],$$

with the initialization $\boldsymbol{\omega}^0 = \mathbf{0}_p$. In the classical ISTA, the shrinkage operator S_λ (applied entrywise) is defined as

$$S_\lambda(x) = \operatorname{sign}(x) \cdot \max(0, |x| - \lambda). \tag{4}$$

Note that when we refer to the function itself, we write S_λ for some threshold $\lambda > 0$; instead, if it is applied in the context of ISTA, the stepsize $\tau > 0$ is included so that we write $S_{\tau\lambda}$ then. We set $\tau = 1/\|\mathbf{X}\|_{2\to 2}^2$, as ISTA is known to converge for $\tau\|\mathbf{X}\|_{2\to 2}^2 < 2$ (Daubechies et al., 2004).

2.2. Large-Dimensional Framework

We aim to find the distribution of $g(\mathbf{x}) = {\boldsymbol{\omega}^\star}^\mathsf{T}\mathbf{x}$ for a new test datum $\mathbf{x}$ under the following large dimensional regime.

Assumption 1 (Growth Rate)**.** *As $n \to \infty$, $p \to \infty$, we assume $p/n \to c_0 > 0$ and $n_\ell/n \to c_\ell \in (0,1)$, $\ell = 1, 2$.*

This assumption of the commensurable relationship between the number of samples and their dimension corresponds to a realistic regime and differs from classical asymptotics where the number of samples is often assumed to be exponentially larger than the feature size, which is very unlikely in real-life applications. Moreover, although the theoretical study was carried out in an asymptotic framework (as $p \to \infty$), the empirical classification error converges towards the asymptotic classification at the rate of $\mathcal{O}(1/\sqrt{p})$, which allows an application to real datasets as stipulated in the experiments of Sections 3 and 4. Furthermore, we will assume the following concentration property on the data.

Assumption 2 (Distribution of $\mathbf{X}$ and $\mathbf{x}$)**.** *There exist two constants $C, c > 0$ (independent of n, p) such that, for any 1-Lipschitz function $f : \mathbb{R}^{p\times n} \to \mathbb{R}$,*

$$\mathbb{P}(|f(\mathbf{X}) - m_{f(\mathbf{X})}| \geq t) \leq Ce^{-(t/c)^2} \qquad \forall t > 0,$$

where m_Z is a median of the random variable Z. We require that the columns of $\mathbf{X}$ are independent and that for $\ell \in \{1, 2\}$, $\mathbf{x}_1^{(\ell)}, \dots, \mathbf{x}_{n_\ell}^{(\ell)}$ are i.i.d. such that $\operatorname{Cov}(\mathbf{x}_i^{(\ell)}) = \mathbf{I}_p$. We further denote the mean and covariance for the columns of $\mathbf{X}$ respectively as $\boldsymbol{\mu}_\ell \equiv \mathbb{E}[\mathbf{x}_1^{(\ell)}]$ and $\mathbf{C}_\ell = \mathbf{I}_p + \boldsymbol{\mu}_\ell\boldsymbol{\mu}_\ell^\mathsf{T}$.

Assumption 2 notably encompasses the following scenarios: the columns of $\mathbf{X}$ are (i) independent Gaussian random vectors with identity covariance, (ii) independent random vectors uniformly distributed on the $\mathbb{R}^p$ sphere of radius $\sqrt{p}$, and, most importantly, (iii) any Lipschitz continuous transformation thereof. Scenario (iii) is of particular relevance for practical data settings as it was recently shown (Seddik et al., 2020). Indeed, random data generated by GANs (for example, images) can be modeled as in case (iii).

The assumption of the covariance matrix being the identity matrix will be used throughout the main paper to prevent the presentation from becoming even more technical. In the supplementary material, the simple isotropic model will be relaxed to the more realistic setting of generic covariances.

3. Main Result and Proof Sketch

3.1. Outline of the Main Technical Steps

Preliminaries. Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ be some random data matrix as per Assumption 2 and let $\mathbf{y} \in \mathbb{R}^n$ be the associated labels. At any iteration $j \geq 0$ and given an initial value of the separating hyperplane $\boldsymbol{\omega}^0 \in \mathbb{R}^p$, the ISTA update scheme writes as $\boldsymbol{\omega}^{j+1} = S_{\tau\lambda}\left(\boldsymbol{\omega}^j - \tau\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}^j + \tau\mathbf{X}\mathbf{y}\right)$. Since the data matrix $\mathbf{X}$ is random, so is the associated optimal separating hyperplane $\boldsymbol{\omega}^\star$. We want to track its random behavior as a function of the statistical properties of $\mathbf{X}$ and the parameters of the model. As the decision score $g(\mathbf{x}) = {\boldsymbol{\omega}^\star}^\mathsf{T}\mathbf{x}$ for classifying each test datum $\mathbf{x}$ turns out to be asymptotically Gaussian (Lemma 1 in the appendix) and depends on $\boldsymbol{\omega}^\star$, and by independence of the training and test dataset, our main focus will be on computing the mean and the covariance of $\boldsymbol{\omega}^\star$ to derive the classification error.

Computing the statistics of $\boldsymbol{\omega}^\star$. The main challenge of computing $\mathbb{E}[\boldsymbol{\omega}^\star]$ and $\operatorname{Cov}(\boldsymbol{\omega}^\star)$ arises from the intricate dependency introduced through ISTA (3). At iteration j, let us denote the random vector $\mathbf{z}^j = \boldsymbol{\omega}^j - \tau\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}^j + \tau\mathbf{X}\mathbf{y}$. Our approach is to construct an iterative scheme such that

$$\mathbb{E}\left[\boldsymbol{\omega}^{j+1}\right] = \mathbb{E}\left[S_{\tau\lambda}(\mathbf{z}^j)\right].$$

Under Assumptions 1 and 2, in the large-dimensional limit of p and n, we prove in Appendix B.2.1 that $\mathbf{z}^j$ is equivalent to a Gaussian random vector. For random vectors $\mathbf{v} \sim \mathcal{N}(\bar{\mathbf{v}}, \boldsymbol{\Sigma}_\mathbf{v})$, we recall the definition of the diagonal of its covariance matrix by $\boldsymbol{\sigma}_{\mathbf{v}} = \mathcal{D}(\boldsymbol{\Sigma}_{\mathbf{v}})$ and define the function

$$\begin{aligned}\varphi : \mathbb{R}_{>0} \times \mathbb{R}^p \times \mathbb{R}^p &\to \mathbb{R}^p, \\ (\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_{\mathbf{v}}) &\mapsto \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}},\boldsymbol{\Sigma}_{\mathbf{v}})}[S_\lambda(\mathbf{v})].\end{aligned}$$

Note that S_λ is applied entrywise, and the expectation is taken entrywise as well, so it is convenient just to pass $\boldsymbol{\sigma}_{\mathbf{v}}$ (*i.e.,* containing the variances of all the single components of $\mathbf{v}$) as an argument to φ. Therefore, for every iteration j,

$$\mathbb{E}\left[\boldsymbol{\omega}^{j+1}\right] = \varphi\left(\tau\lambda, \bar{\mathbf{z}}^j, \boldsymbol{\sigma}_{\mathbf{z}^j}\right).$$

Interestingly from a computational point of view, the function φ has a closed form expression which is provided in Lemma 3 in Appendix B.3. To evaluate $\mathbb{E}[\boldsymbol{\omega}^{j+1}]$, the main task is to estimate the quantities $\bar{\mathbf{z}}^j$ and $\boldsymbol{\sigma}_{\mathbf{z}^j}$. For the sake of simplicity, we just expose the proof strategy for estimating $\bar{\mathbf{z}}^j$ and defer the derivation for $\boldsymbol{\sigma}_{\mathbf{z}^j}$ to Section B.2.3 in the supplementary material. By linearity of the expectation,

$$\begin{aligned}\bar{\mathbf{z}}^j &= \mathbb{E}\left[\boldsymbol{\omega}^j - \tau\mathbf{X}\mathbf{X}^\top\boldsymbol{\omega}^j + \tau\mathbf{X}\mathbf{y}\right] \\ &= \bar{\boldsymbol{\omega}}^j - \tau\sum_{i=1}^n \mathbb{E}\left[({\boldsymbol{\omega}^j}^\top\mathbf{x}_i)\mathbf{x}_i\right] + \tau\mathbb{E}[\mathbf{X}\mathbf{y}],\end{aligned}$$

where the expectation is taken with respect to the data distribution (Assumption 2). The dependency between the vector $\boldsymbol{\omega}^j$ at iteration j and the (columns of the) data matrix $\mathbf{X}$ prevents a straightforward calculation of $\mathbb{E}[{\boldsymbol{\omega}^j}^\top\mathbf{x}_i]$ required to evaluate $\mathbb{E}[({\boldsymbol{\omega}^j}^\top\mathbf{x}_i)\mathbf{x}_i]$. As such, to handle these statistical dependencies, we rely on a *leave-one-out* procedure. More precisely, our approach is to approximate $\mathbb{E}[{\boldsymbol{\omega}^j}^\top\mathbf{x}_i]$ for both classes using the functions

$$\zeta_{\mathcal{C}_{\pi(i)}}\left(\mathbb{E}\left[\mathbf{x}_i^\top\boldsymbol{\omega}_{-i}^j\right]\right), \qquad \pi(i) \in \{1,2\}, \tag{5}$$

where $\pi(i) \in \{1,2\}$ denotes the index of the class of $\mathbf{x}_i$. $\zeta_{\mathcal{C}_{\pi(i)}} : \mathbb{R} \to \mathbb{R}$ is deterministic and $\boldsymbol{\omega}_{-i}^j \in \mathbb{R}^p$ is calculated similarly to $\boldsymbol{\omega}^j$, but deprived of the contribution of $\mathbf{x}_i$. Formally, $\boldsymbol{\omega}_{-i}^j$ is defined through the fixed point system,

$$\boldsymbol{\omega}_{-i}^j = S_{\tau\lambda}\left(\boldsymbol{\omega}_{-i}^j - \tau\mathbf{X}_{-i}\mathbf{X}_{-i}^\top\boldsymbol{\omega}_{-i}^j + \tau\mathbf{X}_{-i}\mathbf{y}_{-i}\right),$$

where $\mathbf{X}_{-i} = [\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{0}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$ and $\mathbf{y}_{-i} = [y_1, \ldots, y_{i-1}, 0, y_{i+1}, \ldots, y_n] \in \mathbb{R}^n$. Therefore, the estimation of $\mathbb{E}[{\boldsymbol{\omega}^j}^\top\mathbf{x}_i]$ consists in finding the deterministic functions $\zeta_{\mathcal{C}_{\pi(i)}}$, since $\zeta_{\mathcal{C}_{\pi(i)}}(\mathbb{E}[\mathbf{x}_i^\top\boldsymbol{\omega}_{-i}^j])$ is estimated given the independence between $\mathbf{x}_i$ and $\boldsymbol{\omega}_{-i}^j$ and using Stein's lemma (Bridle, 1990). Following similar arguments as in (Seddik et al., 2021), the functions $\zeta_{\mathcal{C}_{\pi(i)}}$ are established through determining the difference between $\boldsymbol{\omega}_{-i}^j$ and $\boldsymbol{\omega}^j$. We introduce the following parameterized fixed point system

$$\begin{aligned}\boldsymbol{\omega}_{-i}^j(t) = S_{\tau\lambda}\Big(\boldsymbol{\omega}_{-i}^j(t) &+ \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^\top\boldsymbol{\omega}_{-i}^j(t)\right) \\ &+ \tau t\mathbf{x}_i(\mathbf{y}_i - \boldsymbol{\omega}_{-i}^j(t)^\top\mathbf{x}_i)\Big), \quad t \in [0,1],\end{aligned}$$

which yields a function that interpolates between the regular solution of ISTA, with $\boldsymbol{\omega}^j = \boldsymbol{\omega}_{-i}^j(1)$, and the one obtained through the leave-one-out approach, $\boldsymbol{\omega}_{-i}^j = \boldsymbol{\omega}_{-i}^j(0)$, for varying parameter t, each after j iterations. Specifically, their difference expresses as

$$\boldsymbol{\omega}^j - \boldsymbol{\omega}_{-i}^j = \boldsymbol{\omega}_{-i}^j(1) - \boldsymbol{\omega}_{-i}^j(0) = \int_0^1 \frac{\partial\boldsymbol{\omega}_{-i}^j(t)}{\partial t}\,\mathrm{d}t \in \mathbb{R}^n,$$

where $\frac{\partial\boldsymbol{\omega}_{-i}^j(t)}{\partial t}$ is the derivative of $\boldsymbol{\omega}_{-i}^j(t)$ with respect to t. It has a closed form solution (which is provided in Appendix B.2.2, together will all other ommited technical details) that allows us to find

$$\zeta_{\mathcal{C}_\ell}(r) = \frac{r + (-1)^\ell\kappa_\ell}{1+\kappa_\ell}, \tag{6}$$

where $\kappa_\ell = \tau\,\mathrm{tr}(\mathbf{C}_\ell\bar{\mathbf{D}}\bar{\mathbf{Q}})$ with $\mathbf{C}_\ell$, $\ell = 1,2$, being defined in Assumption 2, and $\bar{\mathbf{D}}$ and $\bar{\mathbf{Q}}$ defined in Theorem 1 below.

Let us remark that, more rigorously, the functions $\zeta_{\mathcal{C}_\ell}$, $\ell \in \{1,2\}$, are actually updated iteratively, as the involved terms are updated in the iteration described below in Theorem 1. Note that the iteratively updated functions $\zeta_{\mathcal{C}_\ell}$ appear in (7) in Theorem 1 below.

These findings are obtained by considering leave-one-out perturbation arguments commonly used in the literature (see (El Karoui et al., 2013; Seddik et al., 2021; Mai et al., 2019)). For the computation of $\boldsymbol{\sigma}_{\mathbf{z}^j}$, we further need to introduce the mappings $\Gamma(\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_{\mathbf{v}}) = \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}},\boldsymbol{\Sigma}_{\mathbf{v}})}[S_\lambda(\mathbf{v})^2]$ and $\psi(\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_{\mathbf{v}}) = \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}},\boldsymbol{\Sigma}_{\mathbf{v}})}[S_\lambda^{'}(\mathbf{v})]$ appearing in the update of $\bar{\mathbf{D}}$ and $\bar{\mathbf{Q}}$. (Note that S_λ is not differentiable, but differentiable almost everywhere so that the derivative $S_\lambda^{'}$ can be defined in a meaningful way as in (35). Alternatively, to avoid technical problems, one may use the fact S_λ can be approximated with arbitrary precision by a smooth function). The closed form expressions of the functions φ, ψ and Γ are provided in Appendix B.3.

3.2. Main Result - Theoretical Classification Error

Theorem 1. *The theoretical classification error (expected test error with respect to the* $0/1$ *loss)* ε *is given as*

$$\varepsilon = Q\left(\frac{\mathfrak{m}_2 - \mathfrak{m}_1}{\boldsymbol{\sigma}_{\boldsymbol{\omega}}^\top\mathbb{1}_p}\right), \qquad Q(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^t e^{-\frac{x^2}{2}}\,\mathrm{d}x.$$

Here, Q *is the Gaussian Q-function and* $\mathfrak{m}_\ell, \ell \in \{1,2\}$*, and* $\boldsymbol{\sigma}_{\boldsymbol{\omega}}$ *are given as the limits of the iteration described below. We initialize* $\bar{\boldsymbol{\omega}}^{(0)}, \boldsymbol{\sigma}_{\mathbf{z}}^{(0)}, \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(0)}, \mathbf{k}_\ell^{(0)}$ *and the scalar quantities* $\kappa_\ell^{(0)}, \mathfrak{m}_\ell^{(0)}$, $\ell \in \{1,2\}$*, and then proceed as follows. We do the following iteration that consists in iteratively updating*

$$\mathfrak{m}_\ell^{(j)} \to \mathfrak{m}_\ell^{(j+1)}, \qquad \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)} \to \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j+1)}.$$

(For an index ℓ, the calculation is always performed for both classes $\ell \in \{1,2\}$.) Firstly, the updates for $\mathbb{m}_\ell^{(j)} \to \mathbb{m}_\ell^{(j+1)}$:

$$\mathbf{a}_\ell^{(j)} = \frac{\mathbb{m}_\ell^{(j)} + (-1)^\ell \kappa_\ell^{(j)}}{1+\kappa_\ell^{(j)}} \boldsymbol{\mu}_\ell + \frac{1}{1+\kappa_\ell^{(j)}} \bar{\boldsymbol{\omega}}^{(j)}, \tag{7}$$

$$\bar{\mathbf{z}}^{(j)} = \bar{\boldsymbol{\omega}}^{(j)} - \tau \sum_{\ell=1}^{2} n_\ell \left(\mathbf{a}_\ell^{(j)} + (-1)^\ell \boldsymbol{\mu}_\ell \right), \tag{8}$$

$$\bar{\boldsymbol{\omega}}^{(j)} = \varphi\left(\lambda\tau, \bar{\mathbf{z}}^{(j)}, \boldsymbol{\sigma}_{\mathbf{z}}^{(j)}\right), \qquad \mathbb{m}_\ell^{(j+1)} = {\bar{\boldsymbol{\omega}}^{(j)}}^{\mathsf{T}} \boldsymbol{\mu}_\ell. \tag{9}$$

Secondly, the updates for $\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)} \to \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j+1)}$ are calculated via

$$\mathcal{E}_\ell^{(j)} = \kappa_\ell^{(j)} - {\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)}}^{\mathsf{T}} \mathbb{1}_p + (-1)^\ell \left(1+\kappa_\ell^{(j)}\right) \mathbb{m}_\ell^{(j+1)}, \tag{10}$$

$$\mathcal{B}_\ell^{(j)} = {\kappa_\ell^{(j)}}^2 + {\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)}}^{\mathsf{T}} \mathbb{1}_p, \tag{11}$$

$$\boldsymbol{\sigma}_1^{(j)} = -\sum_{\ell=1}^{2} \frac{2\tau n_\ell}{1+\kappa_\ell^{(j)}} \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)} + \frac{\mathcal{E}_\ell^{(j)}}{\left(1+\kappa_\ell^{(j)}\right)^2} \mathbf{k}_\ell^{(j)}, \tag{12}$$

$$\boldsymbol{\sigma}_2^{(j)} = \tau^2 n \mathbb{1}_p + \sum_{\ell=1}^{2} \frac{1-(-1)^\ell \mathbb{m}_\ell^{(j+1)}}{1+\kappa_\ell^{(j)}} \mathbf{k}_\ell^{(j)}, \tag{13}$$

$$\boldsymbol{\sigma}_3^{(j)} = \frac{\tau^2 n^2}{\left(1+\kappa_1^{(j)}\right)\left(1+\kappa_2^{(j)}\right)} \boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j)} + \sum_{\ell=1}^{2} \tau^2 n_\ell \mathcal{B}_\ell^{(j)} \mathcal{D}(\mathbf{C}_\ell), \tag{14}$$

$$\boldsymbol{\sigma}_{\mathbf{z}}^{(j+1)} = \sum_{i=1}^{3} \boldsymbol{\sigma}_i^{(j)}, \tag{15}$$

$$\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{(j+1)} = \Gamma\left(\lambda\tau, \bar{\mathbf{z}}^{(j)}, \boldsymbol{\sigma}_{\mathbf{z}}^{(j+1)}\right). \tag{16}$$

In the final steps, the update for high dimensional biases as

$$\bar{\mathbf{D}}^{(j)} = \psi\left(\lambda\tau, \bar{\mathbf{z}}^{(j)}, \boldsymbol{\sigma}_{\mathbf{z}}^{(j+1)}\right), \tag{17}$$

$$\bar{\mathbf{Q}}^{(j)} = \left(\mathbf{I}_p - \bar{\mathbf{D}}^{(j)} + \sum_{\ell=1}^{2} \tau n_\ell \mathbf{C}_\ell \bar{\mathbf{D}}^{(j)} \right)^{-1}, \tag{18}$$

$$\mathbf{k}_\ell^{(j+1)} = \tau \mathcal{D}\left(\mathbf{C}_\ell \bar{\mathbf{D}}^{(j)} \bar{\mathbf{Q}}^{(j)}\right), \; \kappa_\ell^{(j+1)} = {\mathbf{k}_\ell^{(j)}}^{\mathsf{T}} \mathbb{1}_p. \tag{19}$$

Regarding the termination of the algorithm being described in Theorem 1, we use a simple criterion to stop the iteration as soon as all of the involved parameters are not changed anymore by another update (*i.e.,* iteration step), up to some certain prescribed tolerated deviation (measured by the ℓ_2-norm for vectors, and by the absolute value for scalar quantities).

Algorithm 1 Theoretical classification of ISTA [1]

```
Input: Parameters $\lambda, \tau$; estimated means $\hat{\boldsymbol{\mu}}_\ell$ of classes of size $n_\ell$, $\ell = 1, 2$.
For $\ell \in \{1,2\}$, initialize $\mathbf{k}_\ell, \bar{\boldsymbol{\omega}}, \boldsymbol{\sigma}_{\boldsymbol{\omega}}, \boldsymbol{\sigma}_{\mathbf{z}} = \mathbf{0}_p$ and $\mathbb{m}_\ell, \kappa_\ell = 0$; calculate $\mathbf{a}_\ell$ as per equation (10).
repeat
    Compute $\bar{\mathbf{z}} = \bar{\boldsymbol{\omega}} - \sum_{\ell=1}^{2} \tau n_\ell \left(\mathbf{a}_\ell + (-1)^\ell \hat{\boldsymbol{\mu}}_\ell\right)$.
    Compute $\mathcal{B}_\ell$, $\mathcal{E}_\ell$ as per equations (10) and (11).
    Compute variance $\boldsymbol{\sigma}_1$, $\boldsymbol{\sigma}_2$, $\boldsymbol{\sigma}_3$ and as per equations (12), (13), (14).
    Compute the ridge-less variance $\boldsymbol{\sigma}_{\mathbf{z}} = \boldsymbol{\sigma}_1 + \boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_3$.
    Compute $\bar{\mathbf{D}}$ and $\bar{\mathbf{Q}}$ as per equations (18) and update $\kappa_\ell = \tau \operatorname{tr}\left(\mathbf{C}_\ell \bar{\mathbf{D}} \bar{\mathbf{Q}}\right)$.
    Update $\bar{\boldsymbol{\omega}} = \varphi(\lambda, \bar{\mathbf{z}}, \boldsymbol{\sigma}_{\mathbf{z}})$ and $\boldsymbol{\sigma}_{\boldsymbol{\omega}} = \Gamma(\lambda, \bar{\mathbf{z}}, \boldsymbol{\sigma}_{\mathbf{z}})$.
    Update $\mathbb{m}_\ell = \bar{\boldsymbol{\omega}}^{\mathsf{T}} \boldsymbol{\mu}_\ell$ and $\mathbf{a}_\ell = \zeta_{\mathcal{C}_\ell}(\mathbb{m}_\ell) \boldsymbol{\mu}_\ell + \frac{1}{1+\kappa_\ell} \bar{\boldsymbol{\omega}}$.
until Convergence (criterion met).
Output: Classification error $\varepsilon = Q\left(\frac{\mathbb{m}_2 - \mathbb{m}_1}{\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{\mathsf{T}} \mathbb{1}_p}\right)$.
```

The iterative procedure described in the rather technical theorem above is summarized in Algorithm 1.

On the iterative process of Theorem 1. Although the theorem is highly technical and difficult to grasp on first glance, its structure and its understanding resemble that of the underlying iterative soft-thresholding algorithm, with one gradient-descent-like and one sparsity-promoting step (3). The process is also iterative, starting from a random initialization of the involved parameters. Let us note that the quantity κ_ℓ is generally used in Random Matrix Theory to correct for large biases in high dimensions (for more details, refer to Section B.2 of the appendix); the vectors $\bar{\boldsymbol{\omega}}$ and $\boldsymbol{\sigma}_{\boldsymbol{\omega}}$ are the statistics of the hyperplane obtained through ISTA.

On the convergence of Algorithm 1. Let us remark that proving the convergence of the iterative algorithm described in Theorem 1 is beyond the scope of this paper and provides an interesting avenue for future investigations, possibly by adapting the original proof for ISTA (Daubechies et al., 2004). However, Algorithm 1 shows a very favourable behavior empirically, and remarkably even converges faster than the empirical ISTA algorithm as shown in Section C of the appendix (with lower running time and smaller number of iterations required for convergence). While this may seem surprising first, we argue this observation is actually quite natural, supporting the usefulness of the suggested approach. Indeed, the iterative process of Theorem 1 is

[1] For simplicity, we discard the iteration index j. Similarly, we use the simplified notation for $\zeta_{\mathcal{C}_\ell}$ from (6) rather than the more rigorous formulation in (7).

fully deterministic and avoids large fluctuations between the steps of the iteration that are empirically observed for ISTA. Therefore, the iterative process of Theorem 1 exhibits a more stable character being completely deterministic which explains the much faster convergences observed in practice.

The iterations steps in more detail. Similarly to the classical ISTA (3), Algorithm 1 can be broken into two phases.

1. A phase where the Lasso statistics are computed while ignoring the soft-thresholding operator temporarily. This step is similar to computing the statistics of the separating hyperplane obtained through the ridge-less regression problem (*i.e.,* the mere least-squares problem without ℓ_1-regularization). For the first order statistics, $\bar{\mathbf{z}}$ represents the key quantity. This quantity defined in equation (8) is mainly dependent on the vector $\mathbf{a}_\ell$ from (7) which is a high dimensional correction of the mean $\boldsymbol{\mu}_\ell$. Similarly, second order statistics are materialized by the variance vector $\boldsymbol{\sigma}_\mathbf{z}$ defined in equation (16) which depends on three quantities $\boldsymbol{\sigma}_1$, $\boldsymbol{\sigma}_2$ and $\boldsymbol{\sigma}_3$.
2. A sparsity phase realized in Theorem 1 by the functions φ and Γ in equations (9) and (16) to take the soft-thresholding operator into account. However, here this step is taken to impose sparsity on statistics $\bar{\mathbf{z}}$ and $\boldsymbol{\sigma}_z$ calculated previously. We discuss these functions in more detail in Section 3.3 and provide closed-form solutions in Section B.3 of the supplementary material.

Practical evaluation of Theorem 1. Theorem 1 derives the classification error as a function of the statistics of the data and the model parameters. The model parameters, in particular the regularization parameter λ, are provided as input (or can be estimated using the procedure described in Section 4). The only objects requiring estimation are the data statistics, in particular the means per class $\boldsymbol{\mu}_\ell$, $\ell \in \{1, 2\}$. A simple and natural approach and the one used in the article consists in estimating the mean using the empirical mean denoted by $\hat{\boldsymbol{\mu}}_\ell$. This approach has been shown empirically to be very robust, especially for the evaluation of the performance in the Figure 3 or for the choice of optimal hyperparameters in Section 4. These aspects of the estimation deserve however a more thorough analysis which we leave as an interesting avenue for future research.

3.3. Main Theoretical Insights

The regularization parameter λ. The regularization parameter λ mainly appears in the functions φ and Γ which are crucial to take the sparsity constraint into account. Therefore, the behavior of these functions is sufficient in itself to understand the functioning of the Lasso. Besides on the regularization parameter, φ and Γ also depend on the arguments $\bar{\mathbf{z}}$ and $\boldsymbol{\sigma}_\mathbf{z}$. From a statistical perspective, $\bar{\mathbf{z}}$ and $\boldsymbol{\sigma}_\mathbf{z}$ can viewed as the signal and the noise, respectively. Therefore, their elementwise division $\bar{\mathbf{z}} \oslash \boldsymbol{\sigma}_\mathbf{z}$, may be interpreted as the signal-to-noise ratio, which can be seen as a measure of the difficulty of the problem. Therefore, the signal-to-noise ratio on the y axis of Figure 1 will also be referred to as the *task difficulty*. The x axis represents the regularization parameter. The colour map represents the ratio between the theoretical classification error of the Lasso and the classification error of the ridge-less classification (*i.e.,* the case $\lambda = 0$). Thus, it shows the relative gain in terms of classification of Lasso for different values of λ and varying task difficulties for two different sparsity levels α, *i.e.,* the (average) fraction of zero values in the mean of the data. More specifically, to model a sparsity setting, we consider that $\boldsymbol{\mu}_\ell = (-1)^\ell \mathbf{b} \odot \mathbf{m}$ where $\mathbf{m} \sim \mathcal{N}(0, \frac{1}{p}\mathbf{I}_p)$ and $\mathbf{b}$ is a Bernoulli random vector that puts the single entries to zero with probability α/p. Given the aforementioned setting, Figure 1 can essentially be divided into three main regions.

- A region where the relative theoretical accuracy of the classification is lower than one, which implies a decrease of performance compared to ridge-less classification (which does not impose the sparsity constraint).
- A region where the relative theoretical accuracy is greater than one, which means an increase in performance compared to the ridge-less classification. Sparsity constraint has important pertinence in this region.
- A region where the relative theoretical accuracy of Lasso is one, which is essentially the case for $\lambda = 0$.

Figure 2 compares the empirical with the theoretical classification error of the Lasso-based classification as function of the regularization parameter λ, for three different values of the level sparsity α. It is worth remarking that Theorem 1 gives a very close fit between the theoretical and empirical prediction of ISTA, even in a low-sparsity regime. Yet, as expected, for larger value of α (thus more zero entries in the mean of the data; blue curve) and the same *task difficulty*, for an appropriately chosen value of λ, a non-trivial gain is obtained with respect to the ridge-less case, *i.e.,* for $\lambda = 0$.

Robustness to real data. As mentioned previously, the theoretical analysis is carried out under a mixture of concentrated random vectors. This family of vectors was shown to be robust to the real data as it was demonstrated in (Seddik et al., 2020) that (realistic) images generated by GANs are concentrated random vectors by definition. We confirm in Figure 3 this conclusion empirically by comparing the empirical distribution of the decision score $g(\mathbf{x})$ for two databases versus the distribution as predicted by Theorem 1. The real-world datasets consist in the textual *Amazon-review* dataset (Blitzer et al., 2007) and *MNIST* images dataset (Deng, 2012) described in detail in Section 4.

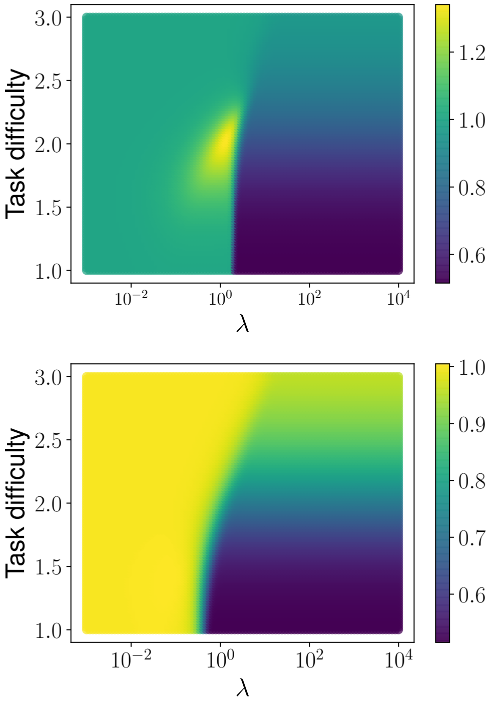

Figure 1. Relative gain in accuracy of the Lasso compared to the ridge-less classifier as a function of the regularization parameter λ and the difficulty of the problem (signal-to-noise ratio). The means are $\boldsymbol{\mu}_\ell = (-1)^\ell \mathbf{b} \odot \mathbf{m}$, $\ell = 1, 2$ where $\mathbf{m} \sim \mathcal{N}(\mathbf{0}_p, \frac{1}{p}\mathbf{I}_p)$, $p =$ 100 and $\mathbf{b}$ is a Bernoulli random vector that puts every single entry to zero with probability α/p for the different settings. (**Top**) High sparsity level of the mean of the data $\alpha = 0.9$. (**Bottom**) Low sparsity with $\alpha = 0.5$. A gain larger than one implies a pertinence of the Lasso constraint, while a gain smaller than one corresponds to the region where the sparsity constraint harms the classification.

4. Application to Hyperparameter Selection

In this section, we illustrate one practical application of Theorem 1 in the context of hyperparameter selection. We use the computational efficient and precise estimate of the theoretical classification error as in Algorithm 1 to perform a grid search to select the best hyperparameter for Lasso-based classification. More precisely, we compare the classification error obtained using the hyperparameter λ found by a grid search on the theoretical classification error and the more widespread one using cross validation for different folds (*i.e.,* different subset size of the data used as validation set).

Figure 4 represents the best classification error using different cross validation folds versus the grid search performed by the theory for different datasets. For each dataset, different tasks are considered and the classification error is averaged over 30 trials with the minimum and maximum value of the classification error represented with an error bar. Although some folds achieve similar performance as obtained by the theory, in other settings, they can lead to unpredictable results with a high variability, whereas the theory provides a more reliable means of assessing performance and can become a reliable alternative to cross-validation.

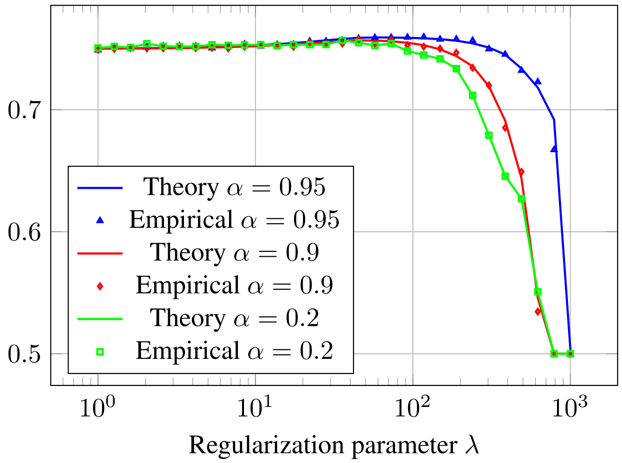

Figure 2. Close fit between the theoretical and empirical (averaged over 1 000 test samples) classification accuracy (as a function of λ), for three different values of α (sparsity level). Gaussian mixture model with class sizes $n_1, n_2 = 500$ and $\mathbf{x}_i^{(\ell)} \sim \mathcal{N}(\boldsymbol{\mu}_\ell, \mathbf{I}_p)$, for $\ell = 1, 2$, with mean $\boldsymbol{\mu}_\ell = (-1)^\ell \mathbf{b} \odot \mathbf{m}$, where $\mathbf{m} \sim \mathcal{N}(\mathbf{0}_p, \frac{1}{p}\mathbf{I}_p)$, and where $\mathbf{b}$ is a Bernoulli random vector that puts each single entry to zero with probability α/p, with the feature size $p = 100$.

Furthermore, cross validation requires an optimal choice of the number of folds which can be painful for the practitioner whereas the theory does not require any parameter to be tuned. From a computational point of view, the grid search performed on the theoretical classification error is comparable to that obtained by the cross validation. This is shown empirically in Section C of the appendix by a comparison of the running time between the choice of the model by the cross validation and the choice of the model by the theory.

The real-world dataset considered in this article are the Amazon review (textual) dataset encoded in $p = 400$-dimensional tf*idf feature vectors of bag-of-words unigrams and bigrams, the MNIST dataset classification (Deng, 2012) where we consider all the differents pairs of classification of digits and the ciphar10 dataset(Krizhevsky et al., 2009). For Amazon review, the positive vs. negative reviews of "`books` (B)", "`dvd` (D)", '`kitchen` (K)' "`electronics` (E)" products are used respectively for the classification. For MNIST and ciphar10 dataset, different images pairs are considered as different classification tasks.

5. Concluding Remarks

In this paper, based on a novel approach from Random Matrix Theory, we propose a theoretical analysis of a Lasso-

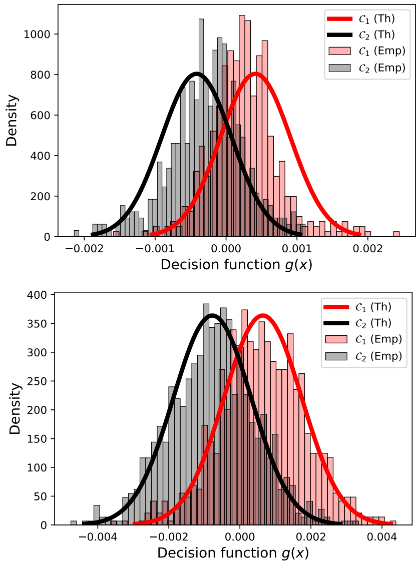

Figure 3. Empirical versus theoretical density of decision score $g(\mathbf{x})$ for (**Top**) Amazon review classification ($p = 400, n_1 = n_2 = 100$) (**Bottom**) MNIST dataset. A PCA processing is applied on the MNIST dataset to extract the $p = 100$ first principal components and $n_1 = n_2 = 100$. For both datasets, 400 samples are used for the test dataset to compute the empirical histogram.

based classification through the analysis of an iterative algorithm (ISTA). The theoretical analysis not only provides interesting insights into its applicability in a classification context, but also offers a reliable alternative to cross-validation.

This theoretical study opens up theoretical perspectives on the analysis of iterative processes that induce very strong dependencies between data. Similar applications are numerous in machine learning and the approach can potentially be used to analyse advanced algorithms such as stochastic gradient descent and tensor-based classification algorithms.

From a more practical point of view, this study opens the exploration of an efficient use of the Lasso in real applications by appropriately choosing the regularization parameter. Natural extensions are the imposition of low-complexity data models like a sparse or low-rank structure (Cai et al., 2010) of the data in classification algorithms and a judicious study of those algorithms. The integration of sparsity constraints in transfer learning algorithms is another avenue of research.

Acknowledgement

We would like to thank Holger Rauhut for interesting discussions. The second author acknowledges funding from the Deutsche Forschungsgemeinschaft (DFG) through the project *Structured Compressive Sensing via Neural Network Learning* (SCoSNeL, MA 1184/36-1) within the SPP 1798 *Compressed Sensing in Information Processing* (CoSIP).

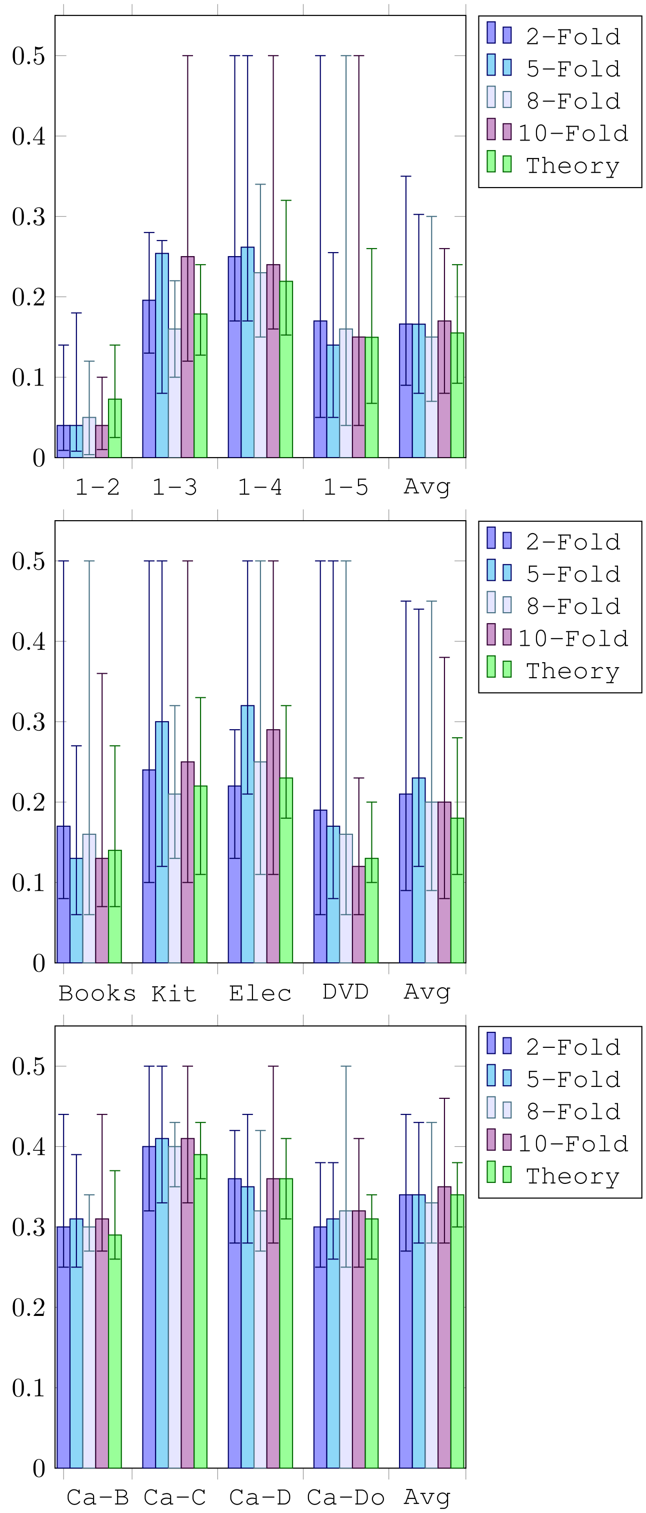

Figure 4. Empirical classification error for different tasks; **(Top)** MNIST: $p = 100$-PCA preprocessing, $n_1 = n_2 = 20$, 500 test samples. **(Center)** Amazon Review dataset: Positive vs. negative review for different classes (Books, Kitchen, Electronics, DVD) with $n_1 = n_2 = 20$, 2 000 test samples; **(Bottom)** Ciphar10 dataset: $p = 100$ PCA preprocessing, $n_1 = n_2 = 20$, 400 test samples for Ca(rs), B(irds), D(eer), Do(gs), and in average (Avg).

References

Abramovich, F. and Grinshtein, V. High-dimensional classification by sparse logistic regression. *IEEE Transactions on Information Theory*, 65(5):3068–3079, 2018.

Alrashdi, A. M., Sifaou, H., Kammoun, A., Alouini, M.-S., and Al-Naffouri, T. Y. Precise error analysis of the lasso under correlated designs. *arXiv preprint arXiv:2008.13033*, 2020.

Baker, A., Aubin, B., Krzakala, F., and Zdeborová, L. Tramp: Compositional inference with tree approximate message passing. *arXiv preprint arXiv:2004.01571*, 2020.

Baraniuk, R., Davenport, M. A., Duarte, M. F., Hegde, C., et al. An introduction to compressive sensing. *Connexions e-textbook*, pp. 24–76, 2011.

Bayati, M. and Montanari, A. The lasso risk for gaussian matrices. *IEEE Transactions on Information Theory*, 58 (4):1997–2017, 2011.

Beck, A. and Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Behboodi, A., Rauhut, H., and Schnoor, E. Compressive sensing and neural networks from a statistical learning perspective. *arXiv preprint arXiv:2010.15658*, 2021.

Blitzer, J., Dredze, M., and Pereira, F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pp. 440–447, 2007.

Bridle, J. S. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing*, pp. 227–236. Springer, 1990.

Cai, J.-F., Candès, E. J., and Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal on optimization*, 20(4):1956–1982, 2010.

Candès, E. and Tao, T. The dantzig selector: Statistical estimation when p is much larger than n. *The annals of Statistics*, 35(6):2313–2351, 2007.

Candès, E. J. and Tao, T. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

Candès, E. J., Romberg, J. K., and Tao, T. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223, 2006.

Candès, E. J. et al. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pp. 1433–1452. Madrid, Spain, 2006.

Celentano, M., Montanari, A., and Wei, Y. The lasso with general gaussian designs with applications to hypothesis testing. *arXiv preprint arXiv:2007.13716*, 2020.

Chen, Y., Fan, J., Ma, C., and Wang, K. Spectral method and regularized mle are both optimal for top-k ranking. *Annals of statistics*, 47(4):2204, 2019.

Couillet, R. and Debbah, M. *Random matrix methods for wireless communications*. Cambridge University Press, New York, NY, USA, first edition, 2011.

Couillet, R. and Louart, C. Concentration of solutions to random equations with concentration of measure hypotheses. 2020.

Daubechies, I., Defrise, M., and De Mol, C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 57(11):1413–1457, 2004.

Deng, L. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Diamond, S. and Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Ding, L. and Chen, Y. Leave-one-out approach for matrix completion: Primal and dual analysis. *arXiv preprint arXiv:1803.07554*, 2018.

Donoho, D. L. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

El Ghaoui, L., Viallon, V., and Rabbani, T. Safe feature elimination in sparse supervised learning technical report no. *Technical report, UCB/EECS-2010–126, EECS Department, University of California, Berkeley*, 2010.

El Karoui, N., Bean, D., Bickel, P. J., Lim, C., and Yu, B. On robust regression with high-dimensional predictors. *Proceedings of the National Academy of Sciences*, 110 (36):14557–14562, 2013.

Elad, M., Figueiredo, M. A., and Ma, Y. On the role of sparse and redundant representations in image processing. *Proceedings of the IEEE*, 98(6):972–982, 2010.

Fleury, B., Guédon, O., and Paouris, G. A stability result for mean width of lp-centroid bodies. *Advances in Mathematics*, 214(2):865–877, 2007.

Foucart, S. and Rauhut, H. *A Mathematical Introduction to Compressive Sensing*. Applied and Numerical Harmonic Analysis. Springer New York, New York, NY, 2013.

Genzel, M. and Kutyniok, G. A mathematical framework for feature selection from real-world data with non-linear observations. *arXiv preprint arXiv:1608.08852*, 2016.

Gerbelot, C. and Berthier, R. Graph-based approximate message passing iterations. *arXiv preprint arXiv:2109.11905*, 2021.

Gerbelot, C., Abbara, A., and Krzakala, F. Asymptotic errors for high-dimensional convex penalized linear regression beyond gaussian matrices. In *Conference on Learning Theory*, pp. 1682–1713. PMLR, 2020.

Gregor, K. and LeCun, Y. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 399–406, 2010.

Hallac, D., Leskovec, J., and Boyd, S. Network lasso: Clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 387–396, 2015.

Huang, H. Asymptotic risk and phase transition of $l\_\{1\}$-penalized robust estimator. *The Annals of Statistics*, 48 (5):3090–3111, 2020.

Jung, A., Tran, N., and Mara, A. When is network lasso accurate? *Frontiers in Applied Mathematics and Statistics*, 3:28, 2018.

Kammoun, A. and Alouini, M.-S. On the precise error analysis of support vector machines. *IEEE Open Journal of Signal Processing*, 2:99–118, 2021.

Klartag, B. A central limit theorem for convex sets. *Inventiones mathematicae*, 168(1):91–131, 2007.

Koh, K., Kim, S.-J., and Boyd, S. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, 8(Jul):1519–1555, 2007.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lee, S.-I., Lee, H., Abbeel, P., and Ng, A. Y. Efficient l˜ 1 regularized logistic regression. In *Aaai*, volume 6, pp. 401–408, 2006.

Louart, C. and Couillet, R. Concentration of measure and large random matrices with an application to sample covariance matrices. *arXiv preprint arXiv:1805.08295*, 2018.

Lu, X. and Li, X. Group sparse reconstruction for image segmentation. *Neurocomputing*, 136:41–48, 2014.

Mai, X. and Couillet, R. Statistical analysis and improvement of large dimensional svm. *private communication*, 2018.

Mai, X. and Liao, Z. High dimensional classification via regularized and unregularized empirical risk minimization: Precise error and optimal loss. *arXiv preprint arXiv:1905.13742*, 2019.

Mai, X., Liao, Z., and Couillet, R. A large scale analysis of logistic regression: Asymptotic performance and new insights. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3357–3361. IEEE, 2019.

Mallat, S. *A wavelet tour of signal processing*. Elsevier, 1999.

Meier, L., Van De Geer, S., and Bühlmann, P. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1): 53–71, 2008.

Mezard, M. and Montanari, A. *Information, physics, and computation*. Oxford University Press, 2009.

Musa, A. B. A comparison of l1-regularizion, pca, kpca and ica for dimensionality reduction in logistic regression. *International Journal of Machine Learning and Cybernetics*, 5(6):861–873, 2014.

Sabato, S., Shalev-Shwartz, S., Srebro, N., Hsu, D. J., and Zhang, T. Learning sparse low-threshold linear classifiers. *J. Mach. Learn. Res.*, 16:1275–1304, 2015.

Schnoor, E., Behboodi, A., and Rauhut, H. Compressive sensing and neural networks from a statistical learning perspective. *arXiv preprint arXiv:2010.15658*, 2021.

Seddik, M. E. A., Louart, C., Tamaazousti, M., and Couillet, R. Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures. In *International Conference on Machine Learning*, pp. 8573–8582. PMLR, 2020.

Seddik, M. E. A., Louart, C., Couillet, R., and Tamaazousti, M. The unexpected deterministic and universal behavior of large softmax classifiers. In *International Conference on Artificial Intelligence and Statistics*, pp. 1045–1053. PMLR, 2021.

Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Thrampoulidis, C., Oymak, S., and Hassibi, B. Regularized linear regression: A precise analysis of the estimation error. In *Conference on Learning Theory*, pp. 1683–1709. PMLR, 2015.

Tibshirani, R. Regression selection and shrinkage via the lasso. *Journal of the Royal Statistical Society Series B*, 58(1):267–288, 1996.

Tiomoko, M., Ali, H. T., and Couillet, R. Deciphering and optimizing multi-task learning: a random matrix approach. In *International Conference on Learning Representations*, 2020.

Van de Geer, S. A. High-dimensional generalized linear models and the lasso. *The Annals of Statistics*, 36(2): 614–645, 2008.

Abstract

The appendix contains the main technical arguments omitted in the core of the article due to space limitation, and is organized as follows. Section A recalls the optimization problem of Lasso as well as the main goal of the theoretical analysis and the assumptions on the data. Section B derives the asymptotic classification error of the Lasso-based classification. To this end, Section B.1 proves the Gaussian distribution of the classification score under concentrated random vector assumptions. Section B.2 details the strategy of the derivation of the first and second order moment of the classification score as well various lemmas related to Gaussian statistics of the soft threshold functions needed for Theorem 1 of the main paper. Sections B.2.2 and B.2.3 then provide the overall derivation of the mean and variance, respectively, of the score of decision for the case of generic covariance matrix. The identity covariance matrix is retrieved as a special case. Section C complements the experimental part of the main article by providing additional experiments and additional insights to the experiments derived in the main paper. Section D explains how to use the code provided as supplementary files to reproduce the results of the paper.

A. Lasso-based classification algorithm

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$ of n data points of feature size p and its associated label vector $\mathbf{y} = [y_1, \ldots, y_n] \in \mathbb{R}^n$ denoted as the training dataset, the goal is to predict the label y for a new test datum $\mathbf{x} \in \mathbb{R}^p$. Since in this article we tackle a binary classification problem, $y_i \in \{-1, 1\}$ for $i = 1, \ldots, n$. The multi-class setting can be retrieved by an application of the one-versus-all approach which uses binary classifiers. For this reason we focus on binary classification. Specifically, $\mathbf{X}$ is a classification problem from the training samples $\mathbf{X} = [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}] \in \mathbb{R}^{p \times n}$ with $\mathbf{X}^{(\ell)} = [\mathbf{x}_1^{(\ell)}, \ldots, \mathbf{x}_{n_\ell}^{(\ell)}] \in \mathbb{R}^{p \times n_\ell}$ the n_ℓ vectors of class $\mathcal{C}_\ell$, $\ell \in \{1, 2\}$. In particular, $n = n_1 + n_2$. The Lasso-based classification algorithm consists in deriving an optimal separating hyperplane $\boldsymbol{\omega}^\star$ (or an approximation thereof) as the solution of the ℓ_1-regularized least squares problem

$$\boldsymbol{\omega}^\star = \operatorname*{arg\,min}_{\boldsymbol{\omega} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}^\mathsf{T} \boldsymbol{\omega}\|_2^2 + \lambda \|\boldsymbol{\omega}\|_1, \tag{20}$$

where $\| \,.\, \|_1$ denotes the ℓ_1-norm, and $\lambda > 0$ is the regularization parameter. Although being convex, the optimization problem (20) is not differentiable due to the non-differentiability of the ℓ_1-norm. An efficient algorithm to solve it is called the iterative soft-thresholding algorithm (ISTA, (Daubechies et al., 2004)), which means to solve the fixed point equation

$$\boldsymbol{\omega} = S_{\tau\lambda} \left(\boldsymbol{\omega} - \tau \mathbf{X} \left(\mathbf{X}^\mathsf{T} \boldsymbol{\omega} - \mathbf{y} \right) \right), \tag{21}$$

where $\tau > 0$ is the step size. The fixed point equation has been proved to converge under appropriate choice of the step size τ in (Daubechies et al., 2004). Specifically $\tau = 1/\|\mathbf{X}\|_{2\to 2}^2$. The classification decision score then unfolds based on the sign of the decision score $g(\mathbf{x}) = {\boldsymbol{\omega}^\star}^\mathsf{T} \mathbf{x}$ on the test datum $\mathbf{x}$. Note that no bias is considered, since the data matrix $\mathbf{X}$ can be centered therefore removing the bias of the decision score. The centering is generally performed through

$$\mathring{\mathbf{X}} = \mathbf{X} \left(\mathbf{I}_n - \frac{1}{n} \mathbb{1}_n \mathbb{1}_n^\mathsf{T} \right). \tag{22}$$

The goal of this work is to predict the theoretical classification error. To that end, one needs to understand the statistical behavior of the decision score $g(\mathbf{x})$ and in particular its distribution (Section B.1) and its first and second order moments (Section B.2.2 and B.2.3), as explained in Section 3. Let us formally recall the Assumptions 1 and 2 from the main paper.

Assumption 1 (Growth Rate). *As $n \to \infty$ and $p \to \infty$, we assume $p/n \to c_0 > 0$ and $n_\ell/n \to c_\ell > 0$ for $\ell = 1, 2$.*

Assumption 2 (Distribution of $\mathbf{X}$ and $\mathbf{x}$). *There exist two constants $C, c > 0$ (independent of n, p) such that, for any 1-Lipschitz function $f : \mathbb{R}^{p \times n} \to \mathbb{R}$,*

$$\mathbb{P}(|f(\mathbf{X}) - m_{f(\mathbf{X})}| \geq t) \leq C e^{-(t/c)^2} \qquad \forall t > 0,$$

where m_Z is a median of the random variable Z. We further impose that the columns of $\mathbf{X}$ are independent. These conditions guarantee the existence of a mean and covariance for the columns of $\mathbf{X}$ and we denote, for all $i \in \{1, \ldots, n_\ell\}$,

$$\boldsymbol{\mu}_\ell \equiv \mathbb{E}[\mathbf{x}_i^{(\ell)}],$$
$$\boldsymbol{\Sigma}_\ell \equiv \operatorname{Cov}(\mathbf{x}_i^{(\ell)}).$$

Furthermore, we write $\mathbf{C}_\ell = \mathbf{\Sigma}_\ell + \boldsymbol{\mu}_\ell \boldsymbol{\mu}_\ell^\mathsf{T}$ *and use the dummy variable* $\mathbf{x} \in \mathbb{R}^p$ *for testing, which is independent of* $\mathbf{X}$.

B. Large dimensional analysis of ISTA

Under the aforementioned setting and assumptions, the goal of this section is precisely to understand the statistical behavior of $g(\mathbf{x}) = \boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x}$ for a new test datum $\mathbf{x}$, which is at the heart of the derivation of the classification error. Specifically, we will look at the distribution of $g(\mathbf{x})$ first in Section B.1, explain the high level strategy of the proof as well as provide the necessary lemmas in Section B.2 and finally and compute its first (Section B.2.2) and second order moment (Section B.2.3).

The standard decision for $\mathbf{x}$ to be allocated to class $\mathcal{C}_1$ ($\mathbf{x} \to \mathcal{C}_\mathbf{1}$) or to class $\mathcal{C}_2$ ($\mathbf{x} \to \mathcal{C}_\mathbf{2}$) is obtained by the following test

$$g(\mathbf{x}) = \boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x} \underset{\mathcal{C}_1}{\overset{\mathcal{C}_2}{\gtrless}} \eta,$$

with $\eta \in \mathbb{R}$ a chosen threshold, the classification error rate ε of which (assuming equal prior class probability) is given by $\varepsilon = \frac{1}{2}\mathbb{P}\left(\mathbf{x} \to \mathcal{C}_1 | \mathbf{x} \in \mathcal{C}_2\right) + \frac{1}{2}\mathbb{P}\left(\mathbf{x} \to \mathcal{C}_2 | \mathbf{x} \in \mathcal{C}_1\right)$. Assuming that $g(\mathbf{x}) = \boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x}$ has a normal distribution $\mathcal{N}(\mathfrak{m}_1, \sigma_1^2)$ for $\mathbf{x} \in \mathcal{C}_1$ and a normal distribution $\mathcal{N}(\mathfrak{m}_2, \sigma_2^2)$ for $\mathbf{x} \in \mathcal{C}_2$, we obtain

$$\begin{aligned}
&\frac{1}{2}\mathbb{P}\left(\mathbf{x} \to \mathcal{C}_1 \,|\, \mathbf{x} \in \mathcal{C}_2\right) + \frac{1}{2}\mathbb{P}\left(\mathbf{x} \to \mathcal{C}_2 \,|\, \mathbf{x} \in \mathcal{C}_1\right) \\
=&\frac{1}{2}\mathbb{P}\left(\boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x} > \eta \,|\, \mathbf{x} \sim \mathcal{N}\left(\mathfrak{m}_1, \sigma_1^2\right)\right) + \frac{1}{2}\mathbb{P}\left(\boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x} < \eta \,|\, \mathbf{x} \sim \mathcal{N}\left(\mathfrak{m}_2, \sigma_2^2\right)\right) \\
=&\frac{1}{2}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{0} \exp\left(-\frac{(x-\eta+\mathfrak{m}_1)^2}{2\sigma_1^2}\right)\mathrm{d}x + \frac{1}{2}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{0} \exp\left(-\frac{(x-\mathfrak{m}_2+\eta)^2}{2\sigma_2^2}\right)\mathrm{d}x \\
=&\frac{1}{2}Q\left(\frac{\mathfrak{m}_1-\eta}{\sigma_1}\right) + \frac{1}{2}Q\left(-\frac{\mathfrak{m}_2-\eta}{\sigma_2}\right) \\
=&\frac{1}{2}Q\left(\frac{\mathfrak{m}_1-\eta}{\sigma_1} - \frac{\mathfrak{m}_2-\eta}{\sigma_2}\right).
\end{aligned}$$

with $Q(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t} e^{-\frac{x^2}{2}}\,\mathrm{d}x$ the Gaussian Q-function. For equal covariance matrix per class ($\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$), the variance of the decision score is the same for class 1 and 2, *i.e.,* $\sigma_1 = \sigma_2 \equiv \sigma$ (see more details in Section (B.2.3)) and the classification error is given by $\varepsilon = \frac{1}{2}Q\left(\frac{\mathfrak{m}_1-\mathfrak{m}_2}{\sigma}\right)$, similar to the main theorem. Furthermore, when additionally the data (assuming equal prior class probability) is centered (*i.e.,* $\mathbb{E}[\mathbf{x} \,|\, \mathbf{x} \in \mathcal{C}_1] + \mathbb{E}[\mathbf{x} \,|\, \mathbf{x} \in \mathcal{C}_2] = 0$) as per equation (22), then also $\mathbb{E}[g(\mathbf{x}) \,|\, \mathbf{x} \in \mathcal{C}_1] = -\mathbb{E}[g(\mathbf{x}) \,|\, \mathbf{x} \in \mathcal{C}_2]$ so that the optimal threshold is $\eta = 0$ and the decision is given by $\eta = 0$, *i.e.,*

$$g(\mathbf{x}) \underset{\mathcal{C}_1}{\overset{\mathcal{C}_2}{\gtrless}} 0.$$

B.1. Distribution of the classification score

The following Lemma ensures we can use the Gaussian Q function to predict the classification error in Theorem 1.

Lemma 1. *Under concentrated random vector assumption on the random vector* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ *as* $p, n \to \infty$, $g(\mathbf{x}) = \boldsymbol{\omega}^\mathsf{T}\mathbf{x}$ *converges in probability to a Gaussian random variable.*

Proof. Under a Gaussian mixture assumption for the input data $\mathbf{X}$, the convergence in distribution of the statistics of the classification score $g_i(\mathbf{x})$ is immediate as the projection of the deterministic vector $\boldsymbol{\omega}$ on the Gaussian random vector $\mathbf{x}$, it follows that $\boldsymbol{\omega}^\mathsf{T}\mathbf{x}$ is asymptotically Gaussian. Since conditionally on the training data $\mathbf{X}$, the classification score $g(\mathbf{x})$ is expressed as the projection of the deterministic vector $\boldsymbol{\omega}$ on the concentrated random vector $\mathbf{x}$, the following central limit theorem (CLT) version for concentrated vector unfolds by proving that projections of deterministic vector on concentrated random vector is asymptotically gaussian. This is ensured by the following result, Theorem 2 (see more details also in (Seddik et al., 2021; Tiomoko et al., 2020)). □

Theorem 2 (CLT for concentrated random vectors (Klartag, 2007; Fleury et al., 2007))**.** *If* $\mathbf{x} \in \mathbb{R}^p$ *is a concentrated random vector with* $\mathbb{E}[\mathbf{x}] = \mathbf{0}$, $\mathbb{E}[\mathbf{x}\mathbf{x}^\mathsf{T}] = \mathbf{I}_p$ *with an observable diameter of order* $\mathcal{O}(1)$ *and* σ *be the uniform measure on* *the sphere $\mathcal{S}^{p-1} \subset \mathbb{R}^p$ of radius 1, then for any integer k, small compared to p, there exist two constants C, c and a set $\Theta \subset (\mathcal{S}^{p-1})^k \subset \mathbb{R}^{p\times k}$ such that $\underbrace{\sigma \otimes \ldots \otimes \sigma}_{k}(\Theta) \geq 1 - \sqrt{p}Ce^{-c\sqrt{p}}$ and for all $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k) \in \Theta$,*

$$\sup_{t\in\mathbb{R}} \left|\mathbb{P}(\mathbf{a}^{\mathsf{T}}\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \geq t) - F_{0,1}(t)\right| \leq Cp^{-\frac{1}{4}} \qquad \forall \mathbf{a} \in \mathbb{R}^k,$$

where $F_{0,1}$ is the cumulative distribution function of the standard normal distribution $\mathcal{N}(0,1)$; see equation (46).

Then the result unfolds naturally. Since $g(\mathbf{x})$ is asymptotically Gaussian by Lemma 1, we will focus on computing its first and second order moments. Before we lay out the technical details, we explain the overall proof strategy in the next section.

B.2. Proof strategy

Given an initial value of the separating hyperplane $\boldsymbol{\omega}^0 \in \mathbb{R}^p$, and a random data matrix $\mathbf{X} \in \mathbb{R}^{p\times n}$ according to Assumption 2, with labels $\mathbf{y} \in \mathbb{R}^n$, we recall that the optimal hyperplane through the ISTA is obtained by the fixed point equation (21). Due to the randomness of the data matrix $\mathbf{X}$, the separating hyperplane $\boldsymbol{\omega}^\star$ obtained from the iterative scheme will have a random behavior with the statistics we want to track as function of the statistics of $\mathbf{X}$ and the parameters of the model. Therefore, our focus will be on computing the first and second order statistics of the separating hyperplane $\boldsymbol{\omega}^\star$. Due to the independence between the training set (used to obtain the feature selector $\boldsymbol{\omega}^\star$ through ISTA) and the test set, the mean and variance of $g(\mathbf{x})$, for any test point $\mathbf{x}$ belonging to either class $\mathcal{C}_\ell$, $\ell = 1, 2$, can be easily computed to be

$$\mathbb{E}[g(\mathbf{x})] = \mathbb{E}[\boldsymbol{\omega}^{\star\mathsf{T}}\mathbf{x}] = \bar{\boldsymbol{\omega}}^{\star^{\mathsf{T}}}\mathbb{E}[\mathbf{x}] = \bar{\boldsymbol{\omega}}^{\star^{\mathsf{T}}}\boldsymbol{\mu}_\ell, \tag{23}$$

$$\mathrm{Var}(g(\mathbf{x})) = \mathbb{E}[g(\mathbf{x})^2] - \mathbb{E}[g(\mathbf{x})]^2 = \mathrm{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_\ell\right) + \mathcal{O}(n^{-1/2}), \tag{24}$$

where we use the shortcut notation $\bar{\boldsymbol{\omega}}_\star = \mathbb{E}[\boldsymbol{\omega}^\star]$. Furthermore, we recall that $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\Sigma}_\ell$ denote the mean and the covariance of the data of the class $\mathcal{C}_\ell$, respectively. The main challenge of computing the mean and the variance of $g(\mathbf{x})$ arises from the computation of $\bar{\boldsymbol{\omega}}^\star = \mathbb{E}[\boldsymbol{\omega}^\star]$ and $\boldsymbol{\Sigma}_{\boldsymbol{\omega}} = \mathrm{Cov}(\boldsymbol{\omega}^\star)$. The main challenge of computing $\mathbb{E}[\boldsymbol{\omega}^\star]$ and $\mathrm{Cov}(\boldsymbol{\omega}^\star)$ arises from the intricate dependency introduced by the iterative scheme. At the iteration j of the fixed point equation (but before applying the soft-thresholding function), let us denote the random vector

$$\mathbf{z}^j = \boldsymbol{\omega}^j - \tau\mathbf{X}\mathbf{X}^{\mathsf{T}}\boldsymbol{\omega}^j + \tau\mathbf{X}\mathbf{y}. \tag{25}$$

In order to find (23) and (24), our strategy is to construct an iterative scheme such that

$$\mathbb{E}[\boldsymbol{\omega}^{j+1}] = \mathbb{E}[S_{\tau\lambda}(\mathbf{z}^j)], \tag{26}$$

$$\mathrm{Cov}(\boldsymbol{\omega}^{j+1}) = \mathbb{E}[S_{\tau\lambda}(\mathbf{z}^j)S_{\tau\lambda}(\mathbf{z}^j)^{\mathsf{T}}]. \tag{27}$$

Since the soft-thresholding function is applied entrywise, to proceed and compute the quantities provided in equations (26) and (27), the following steps are carried out:

- Prove that each element of the vector $\mathbf{z}^j$ is Gaussian.
- Provide closed form solutions for Gaussian integrals over soft threshold functions.
- Compute the statistics of $\mathbf{z}^j$ using the leave-one out approach and derive $\mathbb{E}[\boldsymbol{\omega}^{j+1}]$ and $\mathrm{Cov}(\boldsymbol{\omega}^{j+1})$ from the two previous steps.

B.2.1. Gaussian distribution for entries of $\mathbf{z}^j$

Lemma 2. *Under concentrated random vector assumption on the random vector $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ as $p, n \to \infty$, $\mathbf{z}_k^j$ converges in probability to a Gaussian random variable for each component $1 \leq k \leq p$.*

Proof. We need to find the limiting distribution as $p, n \to \infty$ for every iteration j of the random scalar quantity z_k^j, the ith entry of $\mathbf{z}^j$. Denoting the kth standard basis vector by $\mathbf{e}_k$, it holds that

$$z_i^j = \boldsymbol{\omega}^{j^{\mathsf{T}}}\mathbf{e}_k - \tau\sum_{i=1}^{n}\left(\boldsymbol{\omega}^{j^{\mathsf{T}}}\mathbf{x}_i - y_i\right)x_{ik}.$$

Using the results on concentration of solution to random equations from (Seddik et al., 2021; Couillet & Louart, 2020), we can deduce that $\boldsymbol{\omega}^j$ is a concentrated random vector with an observable diameter of $\mathcal{O}(1/\sqrt{p})$, which means that for every deterministic vector $\mathbf{v}$ of unit norm, $\mathrm{Var}(\boldsymbol{\omega}^\mathsf{T}\mathbf{v}) = \mathcal{O}(1/\sqrt{p})$. This proves that asymptotically in the limit of $p, n \to \infty$, the random vectors ${\boldsymbol{\omega}^j}^\mathsf{T}\mathbf{x}_i$ and ${\boldsymbol{\omega}^j}^\mathsf{T}\mathbf{e}_k$ are deterministic with respect to the random variable x_{ik} for which the variance is of order 1. The result then unfolds from a mere application of the central limit theorem to the sum of independent random variables. □

Since $\mathbf{z}^j$ is asymptotically (as $p, n \to \infty$ with $p = \mathcal{O}(n)$) a Gaussian random vector, computing the mean and the variance in equations (26) and (27) then rely on computing statistics of a functional of gaussian random variables. For $\mathbf{v} \sim \mathcal{N}(\bar{\mathbf{v}}, \boldsymbol{\Sigma}_\mathbf{v})$, we denote the diagonal of its covariance matrix by $\boldsymbol{\sigma}_\mathbf{v} = \mathcal{D}(\boldsymbol{\Sigma}_\mathbf{v})$ and

$$\varphi(\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_\mathbf{v}) = \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}},\boldsymbol{\Sigma}_\mathbf{v})}[S_\lambda(\mathbf{v})], \tag{28}$$

$$\Gamma(\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_\mathbf{v}) = \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}},\boldsymbol{\Sigma}_\mathbf{v})}[S_\lambda(\mathbf{v})S_\lambda(\mathbf{v})^\mathsf{T}]. \tag{29}$$

Note that S_λ on the right-hand side is applied entrywise, and the expectation is taken entrywise as well. Therefore, it is convenient just to pass $\boldsymbol{\sigma}_\mathbf{v}$ (i.e. containing the variances of all single components of $\mathbf{v}$, rather than the entire covariance matrix $\boldsymbol{\Sigma}_v$) as arguments to the two functions φ and Γ. Therefore,

$$\mathbb{E}[\boldsymbol{\omega}^{j+1}] = \varphi(\lambda\tau, \bar{\mathbf{z}}^j, \boldsymbol{\sigma}_{\mathbf{z}^j}),$$
$$\mathrm{Cov}(\boldsymbol{\omega}^{j+1}) = \Gamma(\lambda\tau, \bar{\mathbf{z}}^j, \boldsymbol{\sigma}_{\mathbf{z}^j}).$$

Finally, one needs first to compute the statistics of $\mathbf{z}^j$ and second to compute the functions φ and Γ. The closed form solution for φ and Γ from (28) and (29) will be given below in Section B.3. We focus for now on computing the statistics of $\mathbf{z}^j$.

B.2.2. Computing $\mathbb{E}[\mathbf{z}^j]$

The goal of this section is to find at each iteration the mean of $\mathbf{z}^j = \boldsymbol{\omega}^j + \tau\mathbf{X}\left(\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}^j\right)$, which can be rewritten by discarding the index j for simplicity of exposition as

$$\bar{\mathbf{z}} = \bar{\boldsymbol{\omega}} + \tau\sum_{i=1}^n \boldsymbol{\mu}_{\pi(i)}y_i - \tau\sum_{i=1}^n \mathbb{E}\left[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}\right], \tag{30}$$

where we recall that $\pi(i) \in \{1, 2\}$ denotes the class ($\mathcal{C}_1$ or $\mathcal{C}_2$, respectively) of the sample i; compare also (5). The intrinsic difficulty inherent to the calculus of $\bar{\mathbf{z}}$ arises from the computation of $\sum\limits_{i=1}^n \mathbb{E}\left[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}\right]$ due to the non-trivial dependency between $\boldsymbol{\omega}$ and $\mathbf{x}_i$. Specifically we will first tackle $\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}$, and then the whole expression $\sum\limits_{i=1}^n \mathbb{E}\left[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}\right]$ will be obtained after an application of the Stein identities. To handle this dependency, we propose to write

$$\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega} = \boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i = \boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{x}_i + \boldsymbol{\omega}_\triangle^\mathsf{T}\mathbf{x}_i, \tag{31}$$

where $\boldsymbol{\omega}_{-i}$ is the vector of regression of the Lasso optimization deprived from the contribution of the ith sample, rigorously defined by the solution of the following iterative scheme

$$\boldsymbol{\omega}_{-i} = S_{\tau\lambda}\left(\boldsymbol{\omega}_{-i} + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^\mathsf{T}\boldsymbol{\omega}_{-i}\right)\right) \quad \in \mathbb{R}^p,$$

with $\mathbf{X}_{-i} = [\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{0}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$ and $\mathbf{y}_{-i} = [y_1, \ldots, y_{i-1}, 0, y_{i+1}, \ldots, y_n]^\mathsf{T} \in \mathbb{R}^n$ the data matrix and label vector to each the sample i has been removed. Therefore $\boldsymbol{\omega}_{-i}$ is deprived of the sample $\mathbf{x}_i$ is independent of $\mathbf{x}_i$ which allows to handle easily the term $\boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{x}_i$. We focus now on the second term $\boldsymbol{\omega}_\triangle^\mathsf{T}\mathbf{x}_i$. To that end let us define the parameterized fixed point system (with a parameter $t \in [0, 1]$)

$$\boldsymbol{\omega}_{-i}(t) = S_{\tau\lambda}\left(\boldsymbol{\omega}_{-i}(t) + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^\mathsf{T}\boldsymbol{\omega}_{-i}(t)\right) + \tau t\mathbf{x}_i(y_i - \mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}_{-i}(t))\right) \tag{32}$$

$$= S_{\tau\lambda}\left(\boldsymbol{\omega}_{-i}(t) + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^\mathsf{T}\boldsymbol{\omega}_{-i}(t)\right) + \rho_i(t)\right), \tag{33}$$

with $\rho_i(t) = t\tau\left(y_i - \boldsymbol{\omega}_{-i}(t)^\mathsf{T}\mathbf{x}_i\right)$, such that $\boldsymbol{\omega}_{-i}(0) = \boldsymbol{\omega}_{-i}$ and $\boldsymbol{\omega}_{-i}(1) = \boldsymbol{\omega}$. This mapping can be seen as a path between the two weight vectors $\boldsymbol{\omega} = \boldsymbol{\omega}_{-i}(0)$ and $\boldsymbol{\omega}_{-i} = \boldsymbol{\omega}_{-i}(1)$ of the Lasso classifier, with and without applying the leave-one-out approach. To derive (32) in more detail, we use the original fixed point equation and split up the argument of the soft-thresholding function in two parts, one deprived of the ith component (leave-one-out approach) and a second one that makes up for the difference. Formally,

$$\begin{aligned}
\boldsymbol{\omega} &= S_{\tau\lambda}\left(\boldsymbol{\omega} + \tau\mathbf{X}\left(\mathbf{y} - \mathbf{X}^{\mathsf{T}}\boldsymbol{\omega}\right)\right) \\
&= S_{\tau\lambda}\left(\boldsymbol{\omega} + \tau\sum_{i=1}^{n}\mathbf{x}_i\left(y_i - \mathbf{x}_i^{\mathsf{T}}\boldsymbol{\omega}\right)\right) \\
&= S_{\tau\lambda}\left(\boldsymbol{\omega} + \tau\sum_{k\neq i}^{n}\mathbf{x}_k\left(y_k - \mathbf{x}_k^{\mathsf{T}}\boldsymbol{\omega}\right) + \tau\mathbf{x}_i\left(y_i - \mathbf{x}_i^{\mathsf{T}}\boldsymbol{\omega}\right)\right) \\
&= S_{\tau\lambda}\left(\boldsymbol{\omega} + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^{\mathsf{T}}\boldsymbol{\omega}\right) + \tau\mathbf{x}_i\left(y_i - \mathbf{x}_i^{\mathsf{T}}\boldsymbol{\omega}\right)\right),
\end{aligned}$$

and now including a factor t in the third summand leads to (32). By the uniqueness of the fixed points (by the convergence proof for ISTA (Daubechies et al., 2004)) for $t \in [0,1]$, $t \mapsto \boldsymbol{\omega}_{-i}(t)$ given implicitly in (32) defines a function. Furthermore, this function is continuous, and by the fundamental theorem of calculus, the difference between $\boldsymbol{\omega}$ and its leave-one-out approximation $\boldsymbol{\omega}_{-i}$ can be expressed as

$$\boldsymbol{\omega}_{\Delta} = \boldsymbol{\omega} - \boldsymbol{\omega}_{-i} = \boldsymbol{\omega}^{j}_{-i}(1) - \boldsymbol{\omega}^{j}_{-i}(0) = \int_0^1 \frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t}\,\mathrm{d}t \in \mathbb{R}^p, \tag{34}$$

where $\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t}$ is the derivative of $\boldsymbol{\omega}_{-i}(t)$ with respect to t. Using the function ρ_i introduced in equation (33) then leads to

$$\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} = \left[\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} - \tau\mathbf{X}_{-i}\mathbf{X}_{-i}^{\mathsf{T}}\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} + \frac{\partial\rho(t)}{\partial t}\mathbf{x}_i\right] \odot S_{\tau\lambda}^{'}\left(\boldsymbol{\omega}_{-i}(t) + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^{\mathsf{T}}\boldsymbol{\omega}_{-i}(t)\right) + \rho_i(t)\mathbf{x}_i\right),$$

where $\odot$ is the Hadamard product, i.e. multiplication entrywise. Note that the soft-thresholding function S_λ is differentiable almost everywhere, except for the points $\pm\lambda$. Its derivative is defined piecewise with the points $\pm\lambda$ being assigned to either of the neighboring intervals on which S_λ is being linear (or, more formally, applying the notion of subgradients), for instance

$$S_\lambda' : \mathbb{R} \to \mathbb{R}, \quad x \mapsto \begin{cases} 1 & \text{if } x \ \leq -\lambda, \\ 0 & \text{if } |x| \ < \lambda, \\ 1 & \text{if } x \ \geq \lambda. \end{cases} \tag{35}$$

To make the proof more precise, one could also work with the notion of subgradients or use a smooth approximations of S_λ. However, to avoid the presentation becoming even more technical, we simply use the definition given above. In order to pass from this notation to a matrix times vector multiplication, let us define for convenience

$$\mathbf{D}_i(t) = \operatorname{diag}\left(S_{\tau\lambda}^{'}\left(\boldsymbol{\omega}_{-i}(t) + \tau\mathbf{X}_{-i}\left(\mathbf{y}_{-i} - \mathbf{X}_{-i}^{\mathsf{T}}\boldsymbol{\omega}_{-i}(t)\right) + \rho_i(t)\mathbf{x}_i\right)\right).$$

Then we can rewrite the above equation as

$$\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} = \left[\mathbf{D}_i(t)\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} - \tau\mathbf{D}_i(t)\mathbf{X}_{-i}\mathbf{X}_{-i}^{\mathsf{T}}\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} + \mathbf{D}_i(t)\frac{\partial\rho_i(t)}{\partial t}\mathbf{x}_i\right],$$

and by summarizing terms we finally obtain

$$\left[\mathbf{I}_p - \mathbf{D}_i(t) + \tau\mathbf{D}_i(t)\mathbf{X}_i\mathbf{X}_i^{\mathsf{T}}\right]\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} = \mathbf{D}_i(t)\frac{\partial\rho_i(t)}{\partial t}\mathbf{x}_i \qquad \longleftrightarrow \qquad \mathbf{Q}_i^{-1}(t)\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} = \mathbf{D}_i(t)\frac{\partial\rho_i(t)}{\partial t}\mathbf{x}_i,$$

which leads to a closed form solution for $\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t}$ given by

$$\frac{\partial\boldsymbol{\omega}_{-i}(t)}{\partial t} = \frac{\partial\rho_i(t)}{\partial t}\mathbf{Q}_i(t)\mathbf{D}_i(t)\mathbf{x}_i, \tag{36}$$

where $\mathbf{Q}_i(t) = \left(\mathbf{I}_p - \mathbf{D}_i(t) + \tau \mathbf{D}_i(t)\mathbf{X}_{-i}\mathbf{X}_{-i}^{\mathsf{T}}\right)^{-1}$. Relying on concentration of measure arguments and similarly as proved in (Couillet & Louart, 2020; Seddik et al., 2021), $\mathbf{Q}_i(t)\mathbf{D}_i(t)\mathbf{x}_i$ is almost constant with respect to t, and therefore, the leave-one-out approach will give us a good approximation. Now, plugging (36) into the integral in (34), and by $\rho_i(0) = 0$,

$$\boldsymbol{\omega}_{\Delta} = \int_0^1 \frac{\partial \rho_i(t)}{\partial t} \mathbf{Q}_i(t)\mathbf{D}_i(t)\mathbf{x}_i \, \mathrm{d}t = \rho_i(1)\mathbf{Q}\mathbf{D}\mathbf{x}_i = \tau(y_i - \boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i)\mathbf{Q}\mathbf{D}\mathbf{x}_i,$$

where $\mathbf{Q} := \mathbf{Q}_i(1)$ and $\mathbf{D} := \mathbf{D}_i(1)$. Inserting the obtained expression of $\boldsymbol{\omega}_{\Delta}$ into equation (31), we obtain

$$\begin{aligned} \mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i] &= \mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i + \tau(y_i - \boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i)\mathbf{x}_i^{\mathsf{T}}\mathbf{D}\mathbf{Q}\mathbf{x}_i] \\ &= \mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i] + \mathbb{E}[\tau y_i \mathbf{x}_i^{\mathsf{T}}\mathbf{D}\mathbf{Q}\mathbf{x}_i] - \mathbb{E}[\tau \boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i\mathbf{x}_i^{\mathsf{T}}\mathbf{D}\mathbf{Q}\mathbf{x}_i]. \end{aligned} \tag{37}$$

Let us denote $\bar{\kappa}_i = \tau\mathbb{E}[\mathbf{x}_i^{\mathsf{T}}\mathbf{D}\mathbf{Q}\mathbf{x}_i]$ and the associated vector $\bar{\boldsymbol{\kappa}} = [\bar{\kappa}_1, \ldots, \bar{\kappa}_n]$, we deduce after some simplifications and particularly using Steins Lemma, equation (39) in Proposition 1 below (see also (Seddik et al., 2021, Remark A.9))

$$\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i] = \mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i] + y_i\bar{\kappa}_i - \bar{\kappa}_i\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i] + \mathcal{O}(n^{-1/2}).$$

By collecting terms and rearranging, we are able to express $\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i]$ as a function of $\mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i]$, i.e. using the leave-one-out approach to obtain the closed form solution

$$\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i] = \frac{\mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i] + y_i\bar{\kappa}_i}{1 + \bar{\kappa}_i} + \mathcal{O}(n^{-1/2}).$$

Therefore, we will denote $\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i] = \zeta_{\mathcal{C}_{\pi(i)}}(\mathbb{E}[\boldsymbol{\omega}_{-i}^{\mathsf{T}}\mathbf{x}_i])$, as used in the main paper in equation (6), as follows by

$$\zeta_{\mathcal{C}_{\pi(i)}}(r) = \frac{r + y_i\bar{\kappa}_i}{1 + \bar{\kappa}_i}.$$

We are now ready to compute the expectation of $\mathbf{z}^j$ using the Stein identities (reformulated for our purposes, using the soft-thresholding function) as depicted by Proposition 1 as follows.

Proposition 1 (Stein identities). *Given $\mathbf{x} \in \mathbb{R}^p$, a Gaussian random vector satisfying $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then for any $\mathbf{v}, \boldsymbol{\omega} \in \mathbb{R}^p$ and $\mathbf{A} \in \mathbb{R}^{p \times p}$,*

$$\mathbb{E}[S_{\tau\lambda}(\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x})\mathbf{v}^{\mathsf{T}}\mathbf{x}] = \mathbb{E}[S_{\tau\lambda}(\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x})]\mathbf{v}^{\mathsf{T}}\boldsymbol{\mu} + \mathbb{E}[S_{\tau\lambda}^{'}(\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x})]\mathbf{v}^{\mathsf{T}}\boldsymbol{\Sigma}\boldsymbol{\omega}, \tag{38}$$

$$\mathbb{E}[S_{\tau\lambda}(\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x})\mathbf{x}^{\mathsf{T}}\mathbf{A}\mathbf{x}] = \mathbb{E}[S_{\tau\lambda}(\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x})] \operatorname{tr}(\mathbf{A}\boldsymbol{\Sigma}) + \mathcal{O}(n^{-1/2}). \tag{39}$$

Recalling that $\mathbf{C}_\ell = \boldsymbol{\Sigma}_\ell + \boldsymbol{\mu}_\ell\boldsymbol{\mu}_\ell^{\mathsf{T}}$ for $\ell = 1, 2$ (cf. Assumption 2), from the first Stein identity (38) in Proposition 1, it is immediate that for any $\mathbf{x}_i$ belonging to class $\mathcal{C}_\ell$ and for any $\mathbf{v} \in \mathbb{R}^p$, we have (with κ_i being defined just after (37) above)

$$\mathbb{E}[\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i\mathbf{v}^{\mathsf{T}}\mathbf{x}_i] = \frac{\mathbf{v}^{\mathsf{T}}\mathbf{C}_\ell\bar{\boldsymbol{\omega}} + y_i\bar{\kappa}_i\mathbf{v}^{\mathsf{T}}\boldsymbol{\mu}_\ell}{1 + \bar{\kappa}_i}. \tag{40}$$

Next, let us recall the concept of *deterministic equivalents*, a classical object in random matrix theory (Couillet & Debbah, 2011, Chapter 6), for the matrices $\mathbf{Q}$ and $\mathbf{D}$ which are at the core of the formulation of $\bar{\kappa}_i$. More precisely, a deterministic matrix $\bar{\mathbf{F}} \in \mathbb{R}^{n \times p}$ is said to be a *deterministic equivalent* of a given random matrix $\mathbf{F} \in \mathbb{R}^{n \times p}$, if for any deterministic linear functional $f : \mathbb{R}^{n \times p} \to \mathbb{R}$ of bounded norm (uniformly over p, n), $f(\mathbf{F} - \bar{\mathbf{F}}) \to 0$ almost surely as $n, p \to \infty$. (Analogously, one may define deterministic equivalents for row or column vectors by fixing either $p = 1$, $n \to \infty$ or $n = 1$, $p \to \infty$.) In particular, for $\mathbf{u}, \mathbf{v}$ of unit ℓ_2-norm, $\mathbf{u}^{\mathsf{T}}(\mathbf{F} - \bar{\mathbf{F}})\mathbf{v} \xrightarrow{\text{a.s.}} 0$ and, for $\mathbf{A} \in \mathbb{R}^{p \times n}$ deterministic of bounded operator norm, $\frac{1}{n} \operatorname{tr} \mathbf{A}(\mathbf{F} - \bar{\mathbf{F}}) \xrightarrow{\text{a.s.}} 0$. We will shortly write $\mathbf{F} \leftrightarrow \bar{\mathbf{F}}$ to indicate that $\bar{\mathbf{F}}$ is a deterministic equivalent for $\mathbf{F}$. Deriving deterministic equivalents of the various objects under consideration will be a crucial tool to derive the main result Theorem 1. To begin with, the deterministic equivalent $\bar{\mathbf{z}}$ of $\mathbf{z}$ is obtained using equation (30), further denoting by $\mathbf{a}_\ell$ the deterministic equivalent of $\boldsymbol{\omega}^{\mathsf{T}}\mathbf{x}_i\mathbf{x}_i$ for $\mathbf{x}_i$ in class $\mathcal{C}_\ell$, that is, by (40),

$$\begin{aligned} \bar{\mathbf{z}} &= \bar{\boldsymbol{\omega}} - \tau\sum_{\ell=1}^{2} n_\ell \left(\mathbf{a}_\ell + (-1)^\ell\boldsymbol{\mu}_\ell\right), \\ \mathbf{a}_\ell &= \frac{\mathbf{C}_\ell\bar{\boldsymbol{\omega}} + (-1)^\ell\kappa_\ell\boldsymbol{\mu}_\ell}{1 + \kappa_\ell}, \qquad \ell = 1, 2, \\ \kappa_\ell &= \tau\mathbb{E}\left[\operatorname{tr}(\mathbf{C}_\ell\mathbf{D}\mathbf{Q})\right], \qquad \ell = 1, 2. \end{aligned} \tag{41}$$

Since $\mathbf{x}_1^{(\ell)}, \dots, \mathbf{x}_{n_\ell}^{(\ell)}$, $\ell = 1, 2$, are i.i.d. data vectors, we impose the natural constraint of equal $\bar{\kappa}_1 = \ldots = \bar{\kappa}_{n_\ell}$ (with κ_i being defined just after (37)) within every class $\ell = 1, 2$. As such, we may reduce the complete score vector $\bar{\boldsymbol{\kappa}} \in \mathbb{R}^n$ under the form

$$\bar{\boldsymbol{\kappa}} = [\kappa_1 \mathbb{1}_{n_1}^\mathsf{T}, \kappa_2 \mathbb{1}_{n_2}^\mathsf{T}]^\mathsf{T} = [\underbrace{\kappa_1, \dots, \kappa_1}_{n_1 \text{ times}}, \underbrace{\kappa_2, \dots, \kappa_2}_{n_2 \text{ times}}]^\mathsf{T}. \tag{42}$$

Thus, deterministic equivalents are particularly suitable to handle bilinear forms involving the random matrix $\mathbf{F}$, so in particular the statistics of κ_k, seen as bilinear forms involving the random matrices $\mathbf{Q}$ and $\mathbf{D}$. A deterministic equivalent for $\mathbf{D}$ denoted $\bar{\mathbf{D}}$ is easily obtained by computing the expectation

$$\mathbb{E}\left[\mathbf{D}\right] = \mathbb{E}\left[\operatorname{diag}\left(S_{\tau\lambda}^{'}\left(\mathbf{z}^j\right)\right)\right] = \operatorname{diag}\left(\psi\left(\tau\lambda, \bar{\mathbf{z}}^j, \boldsymbol{\sigma}_{\mathbf{z}^j}\right)\right), \tag{43}$$

with $\mathbf{z}^j \equiv \boldsymbol{\omega}^j + \tau\mathbf{X}\left(\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\omega}^j\right)$ and where $\psi(\lambda, \bar{\mathbf{v}}, \boldsymbol{\sigma}_\mathbf{v}) = \mathbb{E}_{\mathbf{v}\sim\mathcal{N}(\bar{\mathbf{v}}, \boldsymbol{\Sigma}_\mathbf{v})}\left[S_\lambda^{'}(\mathbf{v})\right]$. The deterministic equivalent for $\mathbf{Q}$ is retrieved similarly as provided in (Louart & Couillet, 2018, Section 3.2) and given as

$$\bar{\mathbf{Q}} = \left(\mathbf{I}_p - \bar{\mathbf{D}} + \sum_{\ell=1}^{2} \frac{\tau n_\ell}{1 + \kappa_\ell} \mathbf{C}_\ell \bar{\mathbf{D}}\right)^{-1}.$$

B.2.3. Computing $\mathrm{Cov}(\mathbf{z}^j)$

By discarding the index j in equation (25) for simplicity of the exposition, we will compactly write $\mathbf{z} = \boldsymbol{\omega} - \tau\mathbf{X}(\mathbf{X}^\mathsf{T}\boldsymbol{\omega} - \mathbf{y})$ in the sequel. In order to compute $\mathrm{Cov}(\mathbf{z}) = \mathbb{E}[\mathbf{z}\mathbf{z}^\mathsf{T}] - \bar{\mathbf{z}}\bar{\mathbf{z}}^\mathsf{T}$, we by calculating both $\mathbf{z}\mathbf{z}^\mathsf{T}$ and $\bar{\mathbf{z}}\bar{\mathbf{z}}^\mathsf{T}$ as follows. Firstly, for $\mathbf{z}\mathbf{z}^\mathsf{T}$,

$$\begin{aligned}\mathbf{z}\mathbf{z}^\mathsf{T} &= \left[\boldsymbol{\omega} - \tau\mathbf{X}(\mathbf{X}^\mathsf{T}\boldsymbol{\omega} - \mathbf{y})\right]\left[\boldsymbol{\omega} - \tau\mathbf{X}(\mathbf{X}^\mathsf{T}\boldsymbol{\omega} - \mathbf{y})\right]^\mathsf{T} = \boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T} - \tau\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T} + \tau\boldsymbol{\omega}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T} - \tau\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T} + \tau\mathbf{X}\mathbf{y}\boldsymbol{\omega}^\mathsf{T} \\ &\quad + \tau^2\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T} - \tau^2\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T} - \tau^2\mathbf{X}\mathbf{y}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T} + \tau^2\mathbf{X}\mathbf{y}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T},\end{aligned} \tag{44}$$

by standard linear algebra. Similarly, including the expectation, we obtain for $\mathbf{z}\bar{\mathbf{z}}^\mathsf{T}$ by the linearity of the expected value

$$\begin{aligned}\bar{\mathbf{z}}\bar{\mathbf{z}}^\mathsf{T} &= \mathbb{E}\left[\boldsymbol{\omega} - \tau\mathbf{X}(\mathbf{X}^\mathsf{T}\boldsymbol{\omega} - \mathbf{y})\right]\mathbb{E}\left[\boldsymbol{\omega} - \tau\mathbf{X}(\mathbf{X}^\mathsf{T}\boldsymbol{\omega} - \mathbf{y})\right]^\mathsf{T} = \bar{\boldsymbol{\omega}}\bar{\boldsymbol{\omega}}^\mathsf{T} - \tau\bar{\boldsymbol{\omega}}\mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}] + \tau\bar{\boldsymbol{\omega}}\mathbb{E}[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}] - \tau\mathbb{E}[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\bar{\boldsymbol{\omega}}^\mathsf{T} \\ &\quad + \tau\mathbb{E}[\mathbf{X}\mathbf{y}]\bar{\boldsymbol{\omega}}^\mathsf{T} + \tau^2\mathbb{E}[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}] - \tau^2\mathbb{E}[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\mathbb{E}[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}] - \tau^2\mathbb{E}[\mathbf{X}\mathbf{y}]\mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}] + \tau^2\mathbb{E}[\mathbf{X}\mathbf{y}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}].\end{aligned} \tag{45}$$

Next, we pass to the expectation in (44) and combine it with (45) to obtain an expression for $\mathrm{Cov}(\mathbf{z})$. Let us recall, however, that we are essentially interested in $\operatorname{tr}\left(\mathrm{Cov}(\mathbf{z})\boldsymbol{\Sigma}_\ell\right)$ from (24). Therefore we will compute $\mathbb{E}\left[\operatorname{tr}(\mathbf{P}\mathrm{Cov}(\mathbf{z}))\right]$ for any matrix $\mathbf{P}$ of bounded norm (asymptotically, in the sense of finding an deterministic equivalent for $\mathrm{Cov}(\mathbf{z})$). By taking the trace, while combining (44) and (45), note that we are able to summarize some terms. Namely whenever both some matrix and its transpose appear in the calculation above. For instance, $\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}$ and its transpose $\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}$ both appearing in (44) have the same trace, and similarly $\bar{\boldsymbol{\omega}}\mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}]$ and $\mathbb{E}[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\bar{\boldsymbol{\omega}}^\mathsf{T}$ in (45). Therefore, those terms can be summarized in a_2 below; similar for the other terms.

$$\begin{aligned}&\mathbb{E}[\operatorname{tr}(\mathbf{P}\mathrm{Cov}(\mathbf{z}))] \\ =&\mathbb{E}[\operatorname{tr}(\mathbf{P}\mathbf{z}\mathbf{z}^\mathsf{T})] - \operatorname{tr}(\mathbf{P}\bar{\mathbf{z}}\bar{\mathbf{z}}^\mathsf{T}) \\ =&\underbrace{\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T} - \mathbf{P}\bar{\boldsymbol{\omega}}\bar{\boldsymbol{\omega}}^\mathsf{T}\right)\right]}_{a_1} \underbrace{- 2\tau\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T} - \mathbb{E}\left[\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\bar{\boldsymbol{\omega}}^\mathsf{T}\right)\right]}_{a_2} + \underbrace{2\tau\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{y}\boldsymbol{\omega}^\mathsf{T} - \mathbb{E}\left[\mathbf{P}\mathbf{X}\mathbf{y}\right]\bar{\boldsymbol{\omega}}^\mathsf{T}\right)\right]}_{a_3} \\ &+ \underbrace{\tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T} - \mathbb{E}\left[\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right]\right)\right]}_{a_4} \\ &+ \underbrace{-2\tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T} + \mathbb{E}\left[\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right]\right)\right]}_{a_5} + \underbrace{\tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{y}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T} - \mathbb{E}[\mathbf{P}\mathbf{X}\mathbf{y}]\mathbb{E}[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}]\right)\right]}_{a_6}.\end{aligned}$$

The calculation of the terms $a_1, \dots, a_6$ will enable us to find the deterministic equivalent $\bar{\mathbf{A}}_i$ satisfying $a_i = \operatorname{tr}(\mathbf{P}\bar{\mathbf{A}}_i)$ for any $\mathbf{P} \in \mathbb{R}^{p \times p}$ of asymptotically bounded norm. To begin with, the very first term $\bar{\mathbf{A}}_1 = \boldsymbol{\Sigma}_{\boldsymbol{\omega}}$ is easily obtained since it corresponds to the definition of the covariance matrix. Throughout the calculation, we need to find a random equivalent for $\boldsymbol{\omega}^\mathsf{T}\mathbf{P}\mathbf{x}_i$ that we handle using the decomposition of $\boldsymbol{\omega} = \boldsymbol{\omega}_{-i} + \boldsymbol{\omega}_\triangle$ previously performed (compare (31) and (34)),

$$\boldsymbol{\omega}^\mathsf{T}\mathbf{P}\mathbf{x}_i = \boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{P}\mathbf{x}_i + \boldsymbol{\omega}_\triangle^\mathsf{T}\mathbf{P}\mathbf{x}_i = \boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{P}\mathbf{x}_i + \tau\frac{(y_i - \boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{x}_i)\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{D}\mathbf{Q}\mathbf{x}_i}{1+\bar{\kappa}_i}.$$

Furthermore, we define $K_\ell = \tau\mathbb{E}\left[\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{D}\mathbf{Q}\mathbf{x}_i | \mathbf{x}_i \in \mathcal{C}_\ell\right] = \tau \operatorname{tr}\left(\mathbf{C}_\ell\mathbf{P}\bar{\mathbf{D}}\bar{\mathbf{Q}}\right)$ and the associated matrix $\mathbf{K}_\ell = \tau\mathbf{C}_\ell\mathbf{P}\bar{\mathbf{D}}\bar{\mathbf{Q}}$ and its diagonal elements $\mathbf{k}_\ell = \mathcal{D}(\mathbf{K}_\ell)$. Before we continue with the laborious derivations, let us also recall κ_l from (41) and $\bar{\boldsymbol{\kappa}}$ from (42). Using straightforward algebraic calculations, we obtain successively

$$\begin{aligned}
a_2 &= -2\tau \operatorname{tr}\left(\mathbb{E}[\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}]\right) + 2\tau \operatorname{tr}\left(\mathbb{E}[\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\,\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\right]\right) \\
&= -2\tau\left(\mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}]\right) + 2\tau\mathbb{E}\left[\bar{\boldsymbol{\omega}}^\mathsf{T}\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right] \\
&= -2\tau\sum_{i=1}^n \mathbb{E}[\boldsymbol{\omega}^\mathsf{T}\mathbf{P}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}] + 2\tau\sum_{\ell=1}^2 n_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mathbf{P}\mathbf{a}_\ell \\
&= -2\tau\sum_{i=1}^n \mathbb{E}\left[\left(\boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{P}\mathbf{x}_i + \frac{y_iK_i - K_i\boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{x}_i}{1+\bar{\kappa}_i}\right)\left(\frac{\mathbf{x}_i^\mathsf{T}\boldsymbol{\omega}_{-i} + y_i\bar{\kappa}_i}{1+\bar{\kappa}_i}\right)\right] + 2\tau\sum_{\ell=1}^2 n_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mathbf{P}\mathbf{a}_\ell \\
&= -2\tau\sum_{\ell=1}^2 n_\ell\frac{\operatorname{tr}(\mathbf{P}\mathbf{C}_\ell\mathbf{C}_{\boldsymbol{\omega}})}{1+\kappa_\ell} - 2\tau\sum_{\ell=1}^2\frac{y_i\bar{\kappa}_i \operatorname{tr}\left(\mathbf{P}\boldsymbol{\mu}_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\right)}{1+\kappa_\ell} - 2\tau\sum_{\ell=1}^2\frac{(-1)^\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} - 2\tau\sum_{\ell=1}^2\frac{\kappa_\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} \\
&\quad + 2\tau\sum_{\ell=1}^2\frac{K_i\operatorname{tr}\left(\mathbf{C}_{\boldsymbol{\omega}}\mathbf{C}_\ell\right)}{(1+\kappa_\ell)^2} + 2\tau\sum_{\ell=1}^2\frac{(-1)^\ell\kappa_\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} + 2\tau\sum_{\ell=1}^2\mathbf{P}\bar{\boldsymbol{\omega}}\left[\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell\boldsymbol{\mu}_\ell^\mathsf{T}}{1+\kappa_\ell} + \frac{(-1)^\ell\kappa_\ell\boldsymbol{\mu}_\ell^\mathsf{T}}{1+\kappa_\ell} + \frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\Sigma}_\ell}{1+\kappa_\ell}\right] \\
&= -2\tau\sum_{\ell=1}^2 n_\ell\frac{\operatorname{tr}(\mathbf{P}\boldsymbol{\Sigma}_\ell\boldsymbol{\Sigma}_{\boldsymbol{\omega}})}{1+\kappa_\ell} - 2\tau\sum_{\ell=1}^2\frac{n_\ell(-1)^\ell\kappa_\ell \operatorname{tr}\left(\mathbf{P}\boldsymbol{\mu}_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\right)}{1+\kappa_\ell} - 2\tau\sum_{\ell=1}^2\frac{n_\ell(-1)^\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} - 2\tau\sum_{\ell=1}^2\frac{n_\ell\kappa_\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} \\
&\quad + 2\tau\sum_{\ell=1}^2\frac{n_\ell K_\ell\operatorname{tr}\left(\mathbf{C}_{\boldsymbol{\omega}}\mathbf{C}_\ell\right)}{(1+\kappa_\ell)^2} + 2\tau\sum_{\ell=1}^2\frac{n_\ell(-1)^\ell\kappa_\ell K_\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\mu_\ell}{(1+\kappa_\ell)^2} \\
&\quad + 2\tau\sum_{\ell=1}^2 n_\ell \operatorname{tr}\left(\mathbf{P}\bar{\boldsymbol{\omega}}\left[\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell\boldsymbol{\mu}_\ell^\mathsf{T}}{1+\kappa_\ell} + \frac{(-1)^\ell\kappa_\ell\boldsymbol{\mu}_\ell^\mathsf{T}}{1+\kappa_\ell} + \frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\Sigma}_\ell}{1+\kappa_\ell}\right]\right).
\end{aligned}$$

where $\mathbf{C}_{\boldsymbol{\omega}} = \boldsymbol{\Sigma}_{\boldsymbol{\omega}} + \bar{\boldsymbol{\omega}}\bar{\boldsymbol{\omega}}^\mathsf{T}$. Thus, we obtain the deterministic equivalent $\bar{\mathbf{A}}_2$

$$\begin{aligned}
\bar{\mathbf{A}}_2 = &\sum_{\ell=1}^2 -\frac{2\tau n_\ell\boldsymbol{\Sigma}_\ell\boldsymbol{\Sigma}_{\boldsymbol{\omega}}}{1+\kappa_\ell} - 2\frac{\tau n_\ell(-1)^\ell\kappa_\ell\boldsymbol{\mu}_\ell\bar{\omega}^\mathsf{T}}{1+\kappa_\ell} - 2\frac{n_\ell\tau(-1)^\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell\mathbf{K}_\ell}{(1+\kappa_\ell)^2} - 2\frac{\tau n_\ell\kappa_\ell\mathbf{K}_\ell}{(1+\kappa_\ell)^2} + 2\frac{\tau n_\ell \operatorname{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_\ell\right)\mathbf{K}_\ell}{(1+\kappa_\ell)^2} \\
&\quad + 2\frac{\tau n_\ell(-1)^\ell(\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell)\mathbf{K}_\ell}{(1+\kappa_\ell)^2} + 2\frac{\tau n_\ell\bar{\boldsymbol{\omega}}\mathbf{a}_\ell^\mathsf{T}}{(1+\kappa_\ell)}.
\end{aligned}$$

Next, for a_3 we obtain

$$\begin{aligned}
a_3 &= 2\tau\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{y}\boldsymbol{\omega}^\mathsf{T}\right)\right] - 2\tau\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{y}\bar{\boldsymbol{\omega}}^\mathsf{T}\right)\right] \\
&= 2\tau\sum_{i=1}^n y_i\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] - 2\tau y_i\mathbb{E}\left[\bar{\boldsymbol{\omega}}^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] \\
&= 2\tau\sum_{i=1}^n y_i\mathbb{E}\left[\boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{P}\mathbf{x}_i + \frac{y_iK_i - K_i\boldsymbol{\omega}_{-i}^\mathsf{T}\mathbf{x}_i}{1+\bar{\kappa}_i}\right] - 2\tau y_i\mathbb{E}\left[\bar{\boldsymbol{\omega}}^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] \\
&= 2\sum_{\ell=1}^2\frac{\tau n_\ell K_\ell\left(1-(-1)^\ell\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell\right)}{1+\kappa_\ell}.
\end{aligned}$$

Therefore, the deterministic equivalent $\bar{\mathbf{A}}_3$ is given by

$$\bar{\mathbf{A}}_3 = 2\sum_{\ell=1}^{2} \frac{\tau n_\ell \mathbf{K}_\ell \left(1-(-1)^\ell \bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell\right)}{1+\kappa_\ell}.$$

For a_4 we get

$$\begin{aligned}
a_4 &= \tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right)\right] - \tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right]\right)\right]\\
&= \tau^2\sum_{i,j=1}^{n}\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_j\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right] - \tau^2\operatorname{tr}\left(\mathbf{P}\mathbb{E}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right]\right)\\
&= \tau^2\sum_{i=1}^{n}\mathbb{E}\left[(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i)^2\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] + \tau^2\sum_{i\neq j}^{n}\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_j\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right] - \tau^2\operatorname{tr}\left(\mathbf{P}\mathbb{E}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right]\right)\\
&= \tau^2\sum_{i=1}^{n}\mathbb{E}\left[(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i)^2\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] + \tau^2\sum_{i\neq j}^{n}\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_j\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right] - \tau^2\operatorname{tr}\left(\mathbf{P}\mathbb{E}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{X}\mathbf{X}^\mathsf{T}\right]\right).
\end{aligned}$$

Using the second Stein identity, 39 in Proposition 1 (see also (Seddik et al., 2021, Remark A.9)), we have

$$\begin{aligned}
\mathbb{E}\left[(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i)^2\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_i\right] &= \mathbb{E}\left[(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i)^2\right]\operatorname{tr}\left(\mathbf{P}\mathbf{C}_i\right) + \mathcal{O}(n^{-1/2})\\
\mathbb{E}\left[\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_j\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right] &= \left(\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_i + y_i\bar{\kappa}_i}{1+\bar{\kappa}_i}\boldsymbol{\mu}_i + \frac{\boldsymbol{\Sigma}_i\bar{\boldsymbol{\omega}}}{1+\bar{\kappa}_i}\right)^\mathsf{T}\mathbf{P}\left(\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_i + y_i\bar{\kappa}_i}{1+\bar{\kappa}_i}\boldsymbol{\mu}_i + \frac{\boldsymbol{\Sigma}_i\bar{\boldsymbol{\omega}}}{1+\bar{\kappa}_i}\right)\\
&\quad + \frac{\operatorname{tr}\left(\boldsymbol{\Sigma}_i\mathbf{P}\boldsymbol{\Sigma}_j\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\right)}{(1+\bar{\kappa}_i)(1+\bar{\kappa}_j)} + \mathcal{O}(n^{-1/2})
\end{aligned}$$

By denoting

$$\mathcal{E}_\ell = \mathbb{E}\left[(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i)^2|\mathbf{x}_i\in\mathcal{C}_\ell\right] = \frac{\operatorname{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_\ell\right) + 2(-1)^\ell\kappa_\ell\boldsymbol{\omega}^\mathsf{T}\boldsymbol{\mu}_\ell + \kappa_\ell^2}{(1+\kappa_\ell)^2},$$

we obtain

$$\bar{\mathbf{A}}_4 = \frac{\tau^2 n_1^2\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_1}{(1+\kappa_1)^2} + 2\frac{\tau^2 n_1 n_2\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_2}{(1+\kappa_1)(1+\kappa_2)} + \frac{\tau^2 n_2^2\boldsymbol{\Sigma}_2\boldsymbol{\Sigma}_{\boldsymbol{\omega}}\boldsymbol{\Sigma}_2}{(1+\kappa_2)^2} + \sum_{\ell=1}^{2}\tau^2 n_\ell\mathcal{E}_\ell\mathbf{C}_\ell.$$

In the next step, we obtain a_5 given by

$$\begin{aligned}
a_5 =& -2\tau^2\mathbb{E}\left[\operatorname{tr}\left(\mathbf{P}\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right)\right] + 2\tau^2\mathbb{E}\operatorname{tr}\left(\mathbf{P}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right]\right)\\
=& -2\tau^2\sum_{i,j=1}^{n}y_j\mathbb{E}\left[\operatorname{tr}\left(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right)\right] + 2\tau^2\mathbb{E}\operatorname{tr}\left(\mathbf{P}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right]\right)\\
=& -2\tau^2\sum_{i=1}^{n}y_i\mathbb{E}\left[\operatorname{tr}\left(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_i\right)\right] - 2\tau^2\sum_{i\neq j}^{n}y_j\mathbb{E}\left[\operatorname{tr}\left(\boldsymbol{\omega}^\mathsf{T}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\mathbf{P}\mathbf{x}_j\right)\right] + 2\tau^2\mathbb{E}\operatorname{tr}\left(\mathbf{P}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right]\right)\\
=& -2\tau^2\sum_{i=1}^{n}y_i\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_i + y_{\bar{\bar{i}}}\kappa_i}{1+\bar{\kappa}_i}\operatorname{tr}\left(\mathbf{P}\mathbf{C}_i\right) - 2\tau^2\sum_{i\neq j}^{n}\left(\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_i + y_i\bar{\kappa}_i}{1+\bar{\kappa}_i}\boldsymbol{\mu}_i + \frac{\boldsymbol{\Sigma}_i\bar{\boldsymbol{\omega}}}{1+\bar{\kappa}_i}\right)^\mathsf{T}\mathbf{P}\boldsymbol{\mu}_j\\
&\quad + 2\tau^2\mathbb{E}\operatorname{tr}\left(\mathbf{P}\left[\mathbf{X}\mathbf{X}^\mathsf{T}\boldsymbol{\omega}\right]\mathbb{E}\left[\mathbf{y}^\mathsf{T}\mathbf{X}^\mathsf{T}\right]\right)\\
=& -2\tau^2\sum_{\ell=1}^{2}n_\ell(-1)^\ell\frac{\bar{\boldsymbol{\omega}}^\mathsf{T}\boldsymbol{\mu}_\ell + (-1)^\ell\kappa_\ell}{1+\kappa_\ell}\operatorname{tr}\left(\mathbf{P}\mathbf{C}_\ell\right),
\end{aligned}$$

so that we obtain the deterministic equivalent $\bar{\mathbf{A}}_5$,

$$\bar{\mathbf{A}}_5 = -2\sum_{\ell=1}^{2}\frac{n_\ell\tau^2(-1)^\ell\left(\boldsymbol{\omega}^\mathsf{T}\boldsymbol{\mu}_\ell + (-1)^\ell\bar{\kappa}_\ell\right)\mathbf{C}_\ell}{(1+\kappa_\ell)}.$$

Finally, the deterministic equivalent $\bar{\mathbf{A}}_6$ is given by

$$\bar{\mathbf{A}}_6 = n_\ell \tau^2 \mathbf{\Sigma}_\ell.$$

Remark 1 (Special case of identity covariance matrix $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{I}_p$)**.** *In this case, only the diagonal element of* $\mathbf{\Sigma}_\ell$ *is important since* $\operatorname{Var}(g(\mathbf{x})) = \operatorname{tr}(\mathbf{\Sigma}) = \boldsymbol{\sigma}_{\boldsymbol{\omega}} \mathbb{1}_p$ *with* $\boldsymbol{\sigma}_{\boldsymbol{\omega}} = \mathcal{D}(\mathbf{\Sigma}_{\boldsymbol{\omega}})$*. Therefore, one only needs the deterministic equivalent of the diagonal elements of* $\bar{\mathbf{A}}_i$ *which we denote by* $\boldsymbol{\sigma}_i$*. Similarly, we define* $\mathbf{k}_\ell = \mathcal{D}(\mathbf{K}_\ell)$*. A direct application of the deterministic equivalent obtained for* $\mathbf{A}_i$ *allows to obtain*

$$\begin{aligned}
\boldsymbol{\sigma}_1 &= \boldsymbol{\sigma}_{\boldsymbol{\omega}}, \\
\boldsymbol{\sigma}_2 &= \sum_{\ell=1}^{2} -2\frac{\tau n_\ell \boldsymbol{\sigma}_{\boldsymbol{\omega}}}{1+\kappa_\ell} - 2\frac{\tau n_\ell (-1)^\ell \boldsymbol{\omega}^\mathsf{T} \boldsymbol{\mu}_\ell \mathbf{k}_\ell}{(1+\kappa_\ell)^2} - 2\frac{\tau n_\ell \kappa_\ell \mathbf{k}_\ell}{(1+\kappa_\ell)^2} + 2\frac{\tau n_\ell \sigma_{\boldsymbol{\omega}} \mathbb{1}_p \mathbf{k}_\ell}{(1+\kappa_\ell)^2} + 2\tau\frac{n_\ell (-1)^\ell \kappa_\ell \boldsymbol{\omega}^\mathsf{T} \boldsymbol{\mu}_\ell \mathbf{k}_\ell}{(1+\kappa_\ell)^2}, \\
\boldsymbol{\sigma}_3 &= 2\sum_{\ell=1}^{2} \frac{\tau n_\ell \left(1 - (-1)^\ell \boldsymbol{\omega}^\mathsf{T} \boldsymbol{\mu}_\ell\right) \mathbf{k}_\ell}{1+\kappa_\ell}, \\
\boldsymbol{\sigma}_4 &= \frac{n^2 \tau^2 \boldsymbol{\sigma}_{\boldsymbol{\omega}}}{(1+\kappa_1)(1+\kappa_2)} + \tau^2 \sum_{\ell=1}^{2} n_\ell \mathcal{E}_\ell \mathcal{D}(\mathbf{C}_\ell), \\
\boldsymbol{\sigma}_5 + \boldsymbol{\sigma}_6 &= n\tau^2 \mathbb{1}_p - \sum_{\ell=1}^{2} 2\frac{\tau^2 n_\ell (-1)^\ell \left(\boldsymbol{\omega}^\mathsf{T} \boldsymbol{\mu}_\ell + (-1)^\ell \kappa_\ell\right) \mathcal{D}(\mathbf{C}_\ell)}{1+\kappa_\ell}.
\end{aligned}$$

Furthermore, the result of Theorem 1 of the main paper unfolds by rearranging the terms.

B.3. Explicit formulas for the functions φ, ψ and Γ

In the proof, we used the three help functions φ, ψ and Γ. The goal of this section is to obtain precise and simplified expressions for those functions that are easy to interprete and enable an efficient calculation. Even though already introduced before in equations (28), (29) and (43), let us recall the functions for completeness and in the interest of better readability.

$$\begin{aligned}
\varphi(\lambda, \mu, \sigma) &= \mathbb{E}_{z \sim \mathcal{N}(\mu, \sigma^2)}[S_\lambda(z)], \\
\psi(\lambda, \mu, \sigma) &= \mathbb{E}_{z \sim \mathcal{N}(\mu, \sigma^2)}[S'_\lambda(z)], \\
\Gamma(\lambda, \mu, \sigma) &= \mathbb{E}_{z \sim \mathcal{N}(\mu, \sigma^2)}[S_\lambda(z)^2].
\end{aligned}$$

Next, let us recall the density function $f_{\mu,\sigma^2}(y)$ and the distribution function $F_{\mu,\sigma^2}(y)$ of the univariate normal distribution $\mathcal{N}(\mu, \sigma^2)$ as well as the error function erf and their various properties that will be needed in the proofs below.

$$\begin{gathered}
f_{\mu,\sigma^2}(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right), \\
\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-\tau^2}\, \mathrm{d}\tau, \qquad \operatorname{erf}'(x) = \frac{2}{\sqrt{\pi}} e^{-x^2}, \qquad \operatorname{erf}(-x) = -\operatorname{erf}(x), \\
\operatorname{erf}(a, b) = \frac{2}{\sqrt{\pi}} \int_a^b e^{-\tau^2}\, \mathrm{d}\tau, \qquad \operatorname{erf}(a, b) = \operatorname{erf}(b) - \operatorname{erf}(a), \\
F_{\mu,\sigma^2}(y) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{y-\mu}{\sqrt{2\sigma^2}}\right)\right), \qquad (46) \\
\operatorname{erf}(0) = 0, \qquad \lim_{x \to \infty} \operatorname{erf}(x) = 1, \qquad \lim_{x \to -\infty} \operatorname{erf}(x) = -1.
\end{gathered}$$

Furthermore, in the sequel we will make use of the anti-derivative H_{μ,σ^2} of the function $y \mapsto y \cdot f_{\mu,\sigma^2}(y)$ as well as the anti-derivative $G_{\mu,\sigma^2}(y)$ of the function $y \mapsto y^2 \cdot f_{\mu,\sigma^2}(y)$. They are given by given by

$$H_{\mu,\sigma^2}(y) = \frac{\sigma}{2}\left(-\frac{\mu}{\sigma} \operatorname{erf}\left(\frac{\mu - y}{\sqrt{2}\sigma}\right) - \sqrt{\frac{2}{\pi}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)\right), \qquad (47)$$

$$G_{\mu,\sigma^2}(y) = -\frac{\mu^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu - y}{\sqrt{2\sigma^2}}\right) - \frac{\sigma\,(\mu + y)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu - y)^2}{2\sigma^2}\right). \qquad (48)$$

The following Lemmas 3, 4 and 5 will provide the desired formulas for the three help functions φ, ψ and Γ.

Lemma 3 (Mean of $S_\lambda(z)$.)**.** *Let $z \sim \mathcal{N}(\mu, \sigma^2)$ and S_λ be the soft-thresholding operator with $\lambda > 0$. Furthermore, denote by f_{μ,σ^2} the density function of $\mathcal{N}(\mu, \sigma^2)$. Then, the $\varphi(\lambda, \mu, \sigma) = \mathbb{E}[S_\lambda(z)]$ is given by*

$$\begin{aligned}\varphi(\lambda, \mu, \sigma) =& \mu + \frac{\sigma}{\sqrt{2\pi}}\left[\exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right) - \exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right)\right] \\ &+ \frac{(\mu-\lambda)}{2}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right) - \frac{(\mu+\lambda)}{2}\operatorname{erf}\left(\frac{(\mu+\lambda)}{\sqrt{2}\sigma}\right).\end{aligned}$$

Note that $\lim_{\lambda\to 0}\mathbb{E}[S_\lambda(z)] = \mu$, and furthermore $\lim_{\lambda\to\infty}\mathbb{E}[S_\lambda(z)] = 0$. Indeed, note that the summands containing the erf function can be rewritten as

$$\begin{aligned}&\frac{(\mu-\lambda)}{2}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right) - \frac{(\mu+\lambda)}{2}\operatorname{erf}\left(\frac{(\mu+\lambda)}{\sqrt{2}\sigma}\right) \\ =&\frac{\mu}{2}\left(\operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2}\sigma}\right)\right) - \frac{\lambda}{2}\left(\operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2}\sigma}\right) + \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2}\sigma}\right)\right)\end{aligned}$$

By passing to the limit for $\lambda \to \infty$, using basic properties of the erf function and using the rule of de L'Hospital for the second summand, we obtain

$$\lim_{\lambda\to\infty}\left[\frac{\mu}{2}\left(\operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2}\sigma}\right)\right) - \frac{\lambda}{2}\left(\operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2}\sigma}\right) + \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2}\sigma}\right)\right)\right] = -\mu,$$

which cancels with the other summand μ, while the exponentials vanish in the limit $\lambda \to \infty$.

Proof. Since S_λ is a piecewise linear (or even constant zero) function on the intervals $(-\infty, -\lambda]$, $[-\lambda, -\lambda]$ and $[\lambda, \infty)$, the mean $\mathbb{E}_{z\sim\mathcal{N}(\mu,\sigma^2)}[S_\lambda(z)]$ can be easily obtained by integration via

$$\begin{aligned}\int_{-\infty}^{\infty} S_\lambda(y) f_{\mu,\sigma^2}(y)\,\mathrm{d}y &= \int_{-\infty}^{-\lambda}(y+\lambda)f_{\mu,\sigma^2}(y)\,\mathrm{d}y + \int_{-\lambda}^{\lambda} 0\cdot f_{\mu,\sigma^2}(y)\,\mathrm{d}y + \int_{\lambda}^{\infty}(y-\lambda)f_{\mu,\sigma^2}(y)\,\mathrm{d}y \\ &= \int_{-\infty}^{-\lambda}(y+\lambda)f_{\mu,\sigma^2}(y)\,\mathrm{d}y + \int_{\lambda}^{\infty}(y-\lambda)f_{\mu,\sigma^2}(y)\,\mathrm{d}y \\ &= \int_{-\infty}^{0} y f_{\mu,\sigma^2}(y-\lambda)\,\mathrm{d}y + \int_{0}^{\infty} y f_{\mu,\sigma^2}(y+\lambda)\,\mathrm{d}y \\ &= \int_{-\infty}^{0} y f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y + \int_{0}^{\infty} y f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y. \end{aligned} \tag{49}$$

Let us first focus on the second summand and use (47) (replacing μ by $\mu - \lambda$, and using basic properties of the involved functions):

$$\begin{aligned}&\int_{0}^{\infty} y f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y = \left[H_{\mu-\lambda,\sigma^2}(y)\right]_0^\infty \\ =&\left[\frac{\sigma}{2}\left(-\frac{(\mu-\lambda)}{\sigma}\operatorname{erf}\left(-\frac{y-(\mu-\lambda)}{\sqrt{2}\sigma}\right) - \sqrt{\frac{2}{\pi}}\exp\left(-\frac{(y-(\mu-\lambda))^2}{2\sigma^2}\right)\right)\right]_0^\infty \\ =&\left[\frac{\sigma}{2}\frac{(\mu-\lambda)}{\sigma}\right] - \left[\frac{\sigma}{2}\left(-\frac{(\mu-\lambda)}{\sigma}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right) - \sqrt{\frac{2}{\pi}}\exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right)\right)\right] \\ =&\frac{\sigma}{2}\left[\frac{(\mu-\lambda)}{\sigma} + \frac{(\mu-\lambda)}{\sigma}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right) + \sqrt{\frac{2}{\pi}}\exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right)\right] \\ =&\frac{(\mu-\lambda)}{2} + \frac{(\mu-\lambda)}{2}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right) + \frac{\sigma}{\sqrt{2\pi}}\exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right).\end{aligned}$$

Next, we deal with the first summand above and again use (47) (this time replacing μ by $\mu+\lambda$); similar to above, we obtain

$$\begin{aligned}
&\int_{-\infty}^{0} y f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y = \left[H_{\mu+\lambda,\sigma^2}(y)\right]_{-\infty}^{0} \\
&= \left[\frac{\sigma}{2}\left(-\frac{(\mu+\lambda)}{\sigma}\operatorname{erf}\left(-\frac{y-(\mu+\lambda)}{\sqrt{2}\sigma}\right)-\sqrt{\frac{2}{\pi}}\exp\left(-\frac{(y-(\mu+\lambda))^2}{2\sigma^2}\right)\right)\right]_{-\infty}^{0} \\
&= \left[\frac{\sigma}{2}\left(-\frac{(\mu+\lambda)}{\sigma}\operatorname{erf}\left(\frac{(\mu+\lambda)}{\sqrt{2}\sigma}\right)-\sqrt{\frac{2}{\pi}}\exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right)\right)\right]+\left[\frac{(\mu+\lambda)}{2}\right] \\
&= -\frac{(\mu+\lambda)}{2}\operatorname{erf}\left(\frac{(\mu+\lambda)}{\sqrt{2}\sigma}\right)-\frac{\sigma}{\sqrt{2\pi}}\exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right)+\frac{(\mu+\lambda)}{2}.
\end{aligned}$$

Altogether, we obtain the closed-form solution of $\varphi(\lambda,\mu,\sigma)$,

$$\begin{aligned}
\int_{-\infty}^{\infty} S_\lambda(y) f_{\mu,\sigma^2}(y)\,\mathrm{d}y =& \mu+\frac{\sigma}{\sqrt{2\pi}}\left[\exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right)-\exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right)\right] \\
&+\frac{(\mu-\lambda)}{2}\operatorname{erf}\left(\frac{(\mu-\lambda)}{\sqrt{2}\sigma}\right)-\frac{(\mu+\lambda)}{2}\operatorname{erf}\left(\frac{(\mu+\lambda)}{\sqrt{2}\sigma}\right).
\end{aligned}$$

finishing the proof. □

Lemma 4. *[Mean of $S'_\lambda(z)$.] Let $z \sim \mathcal{N}(\mu,\sigma^2)$ and S_λ be the soft-thresholding operator with $\lambda > 0$. Furthermore, denote by f_{μ,σ^2} the density function of $\mathcal{N}(\mu,\sigma^2)$. Then, the mean $\mathbb{E}[S'_\lambda(z)]$ is given by*

$$\psi(\lambda,\mu,\sigma)=\mathbb{E}_{z\sim\mathcal{N}(\mu,\sigma^2)}[S'_\lambda(z)]=1+\frac{1}{2}\left(\operatorname{erf}\left(-\frac{\lambda+\mu}{\sqrt{2\sigma^2}}\right)-\operatorname{erf}\left(\frac{\lambda-\mu}{\sqrt{2\sigma^2}}\right)\right).$$

Note that by the properties of the erf function, on immediately obtains the limit $\lim_{\lambda\to\infty}\mathbb{E}[S'_\lambda(z)]=0$.

Proof. Since S'_λ is a piecewise linear constant function on the intervals $(-\infty,-\lambda)$, $(-\lambda,-\lambda)$ and (λ,∞), ignoring the borders as they do not contribute to the integration the mean $\mathbb{E}_{z\sim\mathcal{N}(\mu,\sigma^2)}[S'_\lambda(z)]$ can be easily obtained by integration via

$$\begin{aligned}
\int_{-\infty}^{\infty} S_\lambda(y) f_{\mu,\sigma^2}(y)\,\mathrm{d}y =& \int_{-\infty}^{-\lambda} 1\cdot f_{\mu,\sigma^2}(y)\,\mathrm{d}y+\int_{-\lambda}^{\lambda} 0\cdot f_{\mu,\sigma^2}(y)\,\mathrm{d}y+\int_{\lambda}^{\infty} 1\cdot f_{\mu,\sigma^2}(y)\,\mathrm{d}y \\
=& \int_{-\infty}^{-\lambda} f_{\mu,\sigma^2}(y)\,\mathrm{d}y+\int_{\lambda}^{\infty} f_{\mu,\sigma^2}(y)\,\mathrm{d}y \\
=& \int_{-\infty}^{0} f_{\mu,\sigma^2}(y-\lambda)\,\mathrm{d}y+\int_{0}^{\infty} f_{\mu,\sigma^2}(y+\lambda)\,\mathrm{d}y \\
=& \int_{-\infty}^{0} f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y+\int_{0}^{\infty} f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y \\
=& \int_{-\infty}^{0} f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y+1-\int_{-\infty}^{0} f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y \\
=& \frac{1}{2}\left(1+\operatorname{erf}\left(-\frac{\lambda+\mu}{\sqrt{2\sigma^2}}\right)\right)+1-\frac{1}{2}\left(1+\operatorname{erf}\left(\frac{\lambda-\mu}{\sqrt{2\sigma^2}}\right)\right) \\
=& 1+\frac{1}{2}\left(\operatorname{erf}\left(-\frac{\lambda+\mu}{\sqrt{2\sigma^2}}\right)-\operatorname{erf}\left(\frac{\lambda-\mu}{\sqrt{2\sigma^2}}\right)\right),
\end{aligned}$$

finishing the proof. □

Lemma 5. *[Variance of $S_\lambda(z)$.] Let $z \sim \mathcal{N}(\mu, \sigma^2)$ and S_λ be the soft-thresholding operator with $\lambda > 0$. Furthermore, denote by f_{μ,σ^2} the density function of $\mathcal{N}(\mu, \sigma^2)$. Then, the variance $\Gamma(\lambda, \mu, \sigma) = \operatorname{Var}(S_\lambda(z))$ is given by*

$$\begin{aligned}\Gamma(\lambda, \mu, \sigma) =& \mu^2 + \lambda^2 + \sigma^2 + \frac{(\mu+\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2\sigma^2}}\right) + \frac{\sigma(\mu+\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right) \\ &- \frac{(\mu-\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2\sigma^2}}\right) - \frac{\sigma(\mu-\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right) - \mathbb{E}[S_\lambda(z)]^2,\end{aligned}$$

with $\mathbb{E}[S_\lambda(z)]$ given by Lemma 3.

Proof. The mean $\mathbb{E}_{z\sim\mathcal{N}(\mu,\sigma^2)}[S_\lambda^2(z)]$ can be easily obtained by integration via

$$\begin{aligned}\int_{-\infty}^{\infty} S_\lambda(y)^2 f_{\mu,\sigma^2}(y)\,\mathrm{d}y &= \int_{-\infty}^{-\lambda} (y+\lambda)^2 f_{\mu,\sigma^2}(y)\,\mathrm{d}y + \int_{-\lambda}^{\lambda} 0 \cdot f_{\mu,\sigma^2}(y)\,\mathrm{d}y + \int_{\lambda}^{\infty} (y-\lambda)^2 f_{\mu,\sigma^2}(y)\,\mathrm{d}y \\ &= \int_{-\infty}^{0} y^2 f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y + \int_{0}^{\infty} y^2 f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y. \end{aligned} \tag{50}$$

Using the formula for the anti-derivative (48) allows to retrieve for the first summand in (50)

$$\begin{aligned}&\int_{-\infty}^{0} y^2 f_{\mu+\lambda,\sigma^2}(y)\,\mathrm{d}y = \left[G_{\mu+\lambda,\sigma^2}(y)\right]_{-\infty}^{0} \\ =& \left[-\frac{(\mu+\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu+\lambda-y}{\sqrt{2\sigma^2}}\right) - \frac{\sigma(\mu+\lambda+y)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu+\lambda-y)^2}{2\sigma^2}\right)\right]_{-\infty}^{0} \\ =& -\frac{(\mu+\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2\sigma^2}}\right) - \frac{\sigma(\mu+\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right) + \frac{(\mu+\lambda)^2 + \sigma^2}{2}.\end{aligned}$$

For the second summand in (50), we obtain in a similar way

$$\begin{aligned}&\int_{0}^{\infty} y^2 f_{\mu-\lambda,\sigma^2}(y)\,\mathrm{d}y = \left[G_{\mu-\lambda,\sigma^2}(y)\right]_{0}^{\infty} \\ =& \left[-\frac{(\mu-\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu-\lambda-y}{\sqrt{2\sigma^2}}\right) - \frac{\sigma(\mu-\lambda+y)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu-\lambda-y)^2}{2\sigma^2}\right)\right]_{0}^{\infty} \\ =& \frac{(\mu-\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2\sigma^2}}\right) + \frac{\sigma(\mu-\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right) + \frac{(\mu-\lambda)^2 + \sigma^2}{2}.\end{aligned}$$

Therefore, combining our findings finally yields

$$\begin{aligned}\mathbb{E}_{z\sim\mathcal{N}(\mu,\sigma^2)}[S_\lambda^2(z)] =& \mu^2 + \lambda^2 + \sigma^2 + \frac{(\mu+\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu+\lambda}{\sqrt{2\sigma^2}}\right) + \frac{\sigma(\mu+\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu+\lambda)^2}{2\sigma^2}\right) \\ &- \frac{(\mu-\lambda)^2 + \sigma^2}{2} \operatorname{erf}\left(\frac{\mu-\lambda}{\sqrt{2\sigma^2}}\right) - \frac{\sigma(\mu-\lambda)}{\sqrt{2\pi}} \exp\left(-\frac{(\mu-\lambda)^2}{2\sigma^2}\right).\end{aligned}$$

We can then deduce the result by using $\operatorname{Var}_{z\sim\mathcal{N}(\mu,\sigma^2)}(S_\lambda(z)) = \mathbb{E}\left[S_\lambda^2(z)\right] - \mathbb{E}\left[S_\lambda(z)\right]^2$. □

C. Additional experiments

C.1. Running time comparison of theoretical analysis of ISTA

In this section we illustrate in Figure 5 and 6 the discussion of the complexity of the Theorem 1 by providing the number of iterations and the running time required for convergence of the deterministic iterative process (see Theorem 1) and the iterative process of the classical ISTA algorithm. This experiment experimentally confirms the discussion on the iterative process made in the main article and later in section 4 where the issue of algorithmic complexity was raised.

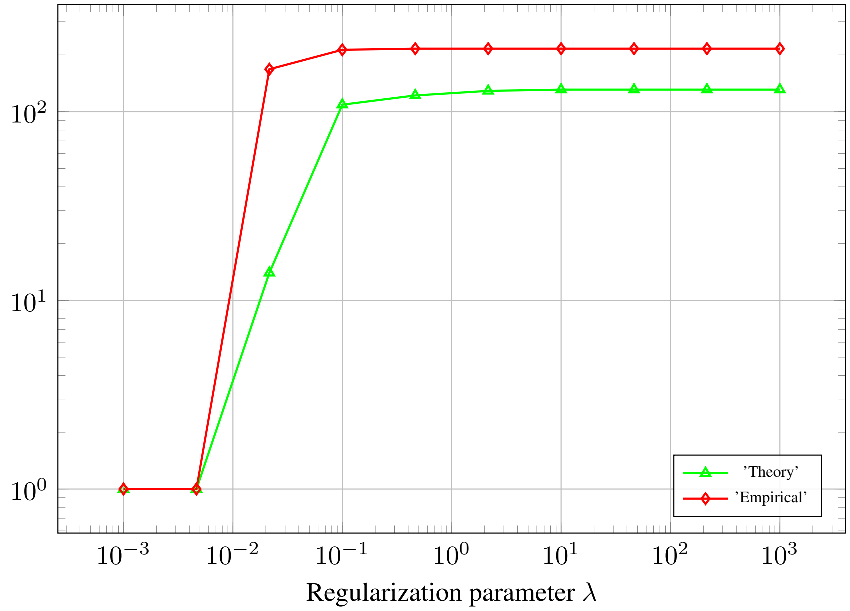

Figure 5. Number of iterations required for empirical versus theoretical ISTA as function of the regularization parameter for a tolerance of e^{-7}. Gaussian mixture model with class sizes $n_1, n_2 = 1\,00$ and $\mathbf{x}_i^{(\ell)} \sim \mathcal{N}(\boldsymbol{\mu}_\ell, \mathbf{I}_p)$, for $\ell = 1, 2$, with mean $\boldsymbol{\mu}_\ell = (-1)^\ell \mathbf{b} \odot \mathbf{m}$, where $\mathbf{m} \sim \mathcal{N}(\mathbf{0}_p, \frac{1}{p}\mathbf{I}_p)$, and where $\mathbf{b}$ is a Bernoulli random vector that puts each single entry to zero with probability α/p, with the feature size $p = 1000$ and $\alpha = 0.9$.

C.2. On the regularization parameter

To complement the experimental part on the influence of the regularization parameter λ, we represent in Figure 7 as function of the regularization parameter for different values of the sparsity level of the mean of the data $\boldsymbol{\mu}_\ell$.

C.3. On the influence of the parameter β.

In this section we propose instead of using the soft threshold function $S_\lambda(x) = (\text{sign}(x) \cdot \max(0, |x| - \lambda))$ as used in the main, we propose a weighted version $S_{\beta,\lambda}(x) = \beta\,(\text{sign}(x) \cdot \max(0, |x| - \lambda))$ in order to infer the influence of β. As shown in Figure 8, the parameter β does not play any role on the theoretical performance. However, small values of β generally leads to a faster convergence of the iterative process.

C.4. On different shrinkage functions

One of the advantages of the theoretical study is to provide a way to compare different functions used in the shrinkage phase of the ISTA algorithm. In this section in Figure 9, we compare two functions: the soft threshold function used in the paper and the piecewise non linear function defined as

$$f(x, \lambda) = (\text{sign}(x) \cdot \max(0, |x| - \lambda))^2 .$$

As such the theoretical analysis can be used to evaluate the pertinent of any shrinkage function for classifying data under a sparse constrain on the the separating hyperplane.

C.5. On the use of different loss functions

D. Code Readme

This section explains how to use the code implementing the “Large Dimensional Analysis of Lasso-based Classification” proposed in the core of the article.

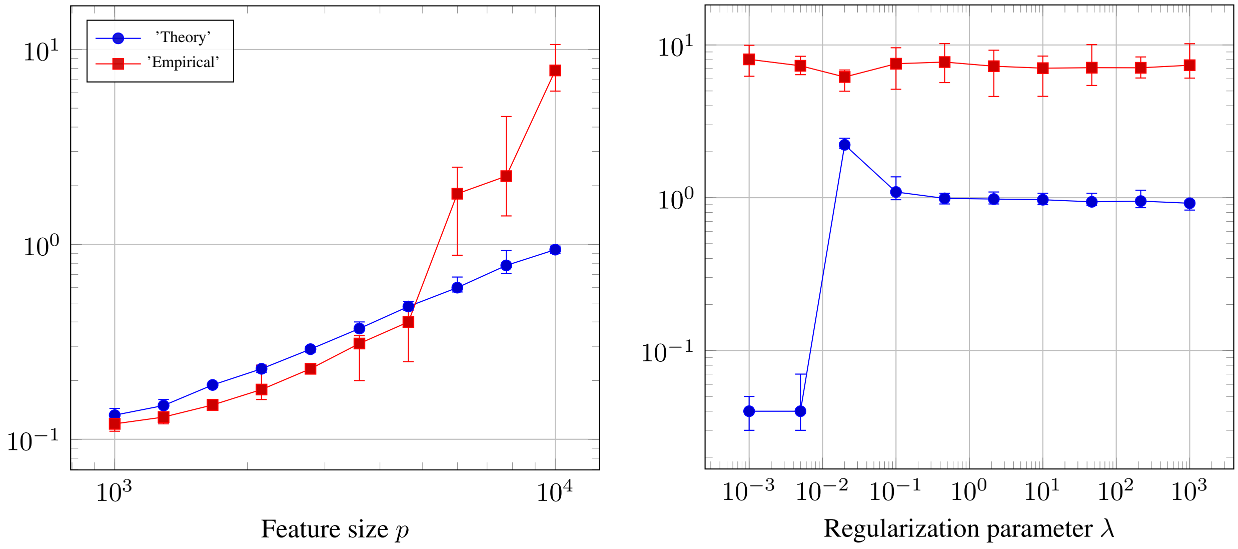

Figure 6. Running time for empirical versus theoretical ISTA for (**left**) as function of the feature size p and (**right**) as a function of the regularization parameter for a tolerance of e^{-7}. Gaussian mixture model with class sizes $n_1, n_2 = 1\ 000$ and $\mathbf{x}_i^{(\ell)} \sim \mathcal{N}(\boldsymbol{\mu}_\ell, \mathbf{I}_p)$, for $\ell = 1, 2$, with mean $\boldsymbol{\mu}_\ell = (-1)^\ell \mathbf{b} \odot \mathbf{m}$, where $\mathbf{m} \sim \mathcal{N}(\mathbf{0}_p, \frac{1}{p}\mathbf{I}_p)$, and where $\mathbf{b}$ is a Bernoulli random vector that puts each single entry to zero with probability α/p with $\alpha = 0.9$.

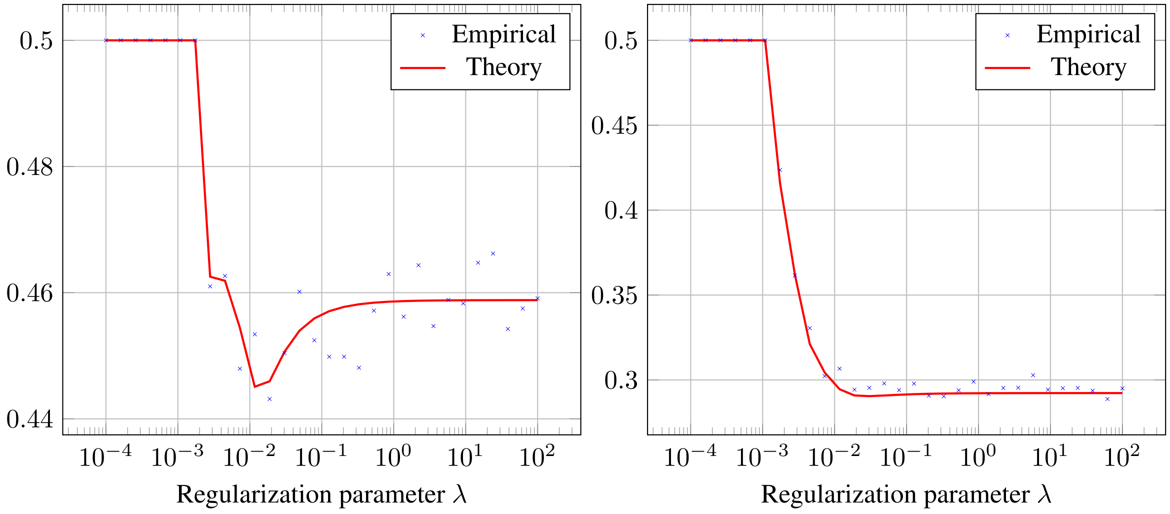

Figure 7. Theoretical versus empirical classification error as function of the regularization parameter, $\boldsymbol{\mu}_\ell \sim \mathcal{N}(\mathbf{0}_p, \mathbf{I}_p)$ and components of $\boldsymbol{\mu}_\ell$ put at zeros with probability $\alpha = 0.95$ (**left**) and $\alpha = 0.5$ (**right**).

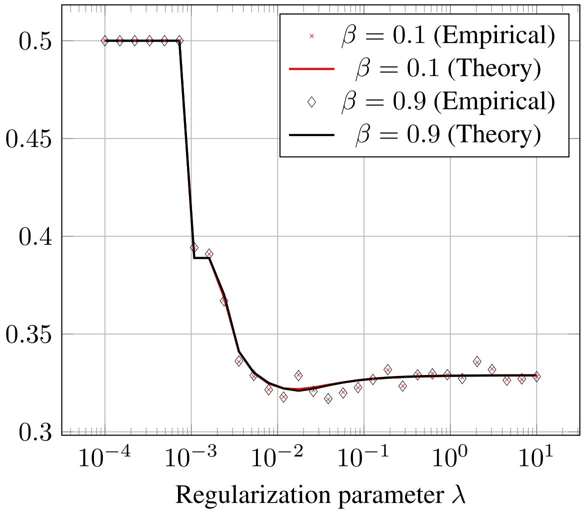

Figure 8. Theoretical versus empirical classification error as function of the regularization parameter $\boldsymbol{\mu}_\ell \sim \mathcal{N}(\mathbf{0}_p, \mathbf{I}_p)$ and components of $\boldsymbol{\mu}_\ell$ put at zeros with probability $\alpha = 0.9$.

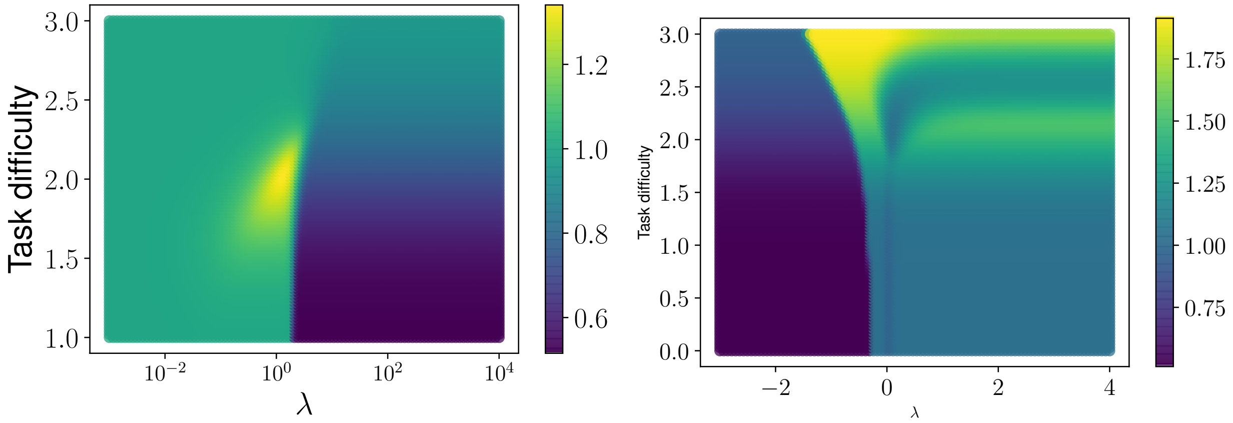

Figure 9. Relative gain in accuracy of the Lasso compared to the ridge-less regression as a function of the regularization parameter and the difficulty of the problem (inverse of the signal-to-noise ratio) for two different shrinkage function of sparsity for $\alpha = 0.9$ and (**left**) soft thresholding-operator and (**right**) Function $f(x, \lambda) = (\text{sign}(x) \cdot \max(0, |x| - \lambda))^2$.

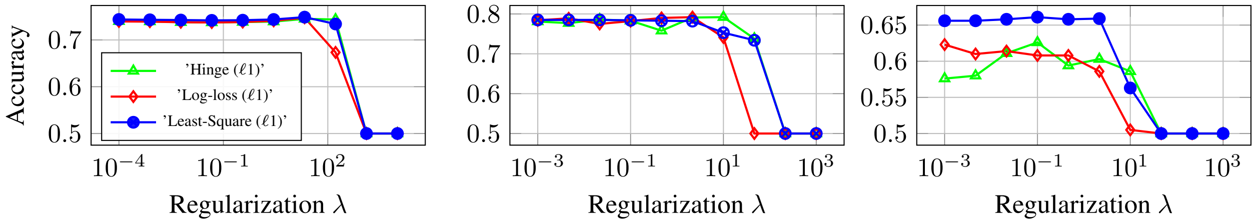

Figure 10. Synthetic (left); MIT-BIH dataset (middle) and Amazon Review (right). (Similar to Figure 2 in the main paper!)

D.1. Archive content

- The function implementing our method is called `ista_theory.py` which computes the theoretical classification error as well as the statistics of the decision score $g(\mathbf{x})$.
- The main script comparing the theoretical versus empirical classification error is `empirical_versus_theory.py`. The code is also used to visualize the theoretical histogram and theoretical Gaussian predictions.

- The main script illustrating the phase diagram illustration the key role of the function φ and Γ is `diagram_phase_lasso.py`.
- The main script illustrating the hyperparameter selection using the theoretical classification error `hyperparameter_selection.py`.
- The script `utils`: containing all the important functions needed to execute the main scripts.
- Folder `data`: where the different datasets can be uploaded.

D.2. Code `empirique_versus_theory.py`

The different options proposed to execute the script `PFA.m` are as follows:

- **"data"** to be chosen between *'Synthetic'*, *'Amazon'*and *MNIST* to test the close fit between theory and empirical ISTA as well as the empirical/theoretical histograms.
- **"domain"** is the domain of the dataset (for MNIST either *'ciphar'* or *'mnist-like'*, for Amazon either *'Books'*, *'Kitchen'*, *'Elec'*, *'DVD'*)

D.3. Reproducing the results of the article

The following sections detail the parameter setting to reproduce the experiments of the main article.

D.3.1. Figure 1

Script → `diagram_phase_lasso.py`
p → 100
α → 0.01/0.5

D.3.2. Figure 2

Script → `empirique_versus_theory.py`
dataset → Synthetic
domain → *'ciphar'*
α → 0.01/0.05/0.5
p → 100

D.3.3. Figure 3

Script → `empirique_versus_theory.py`
dataset → Synthetic/Amazon/MNIST
domain → *'ciphar'/'Books'/'mnist-like'*
α → 0.01
p → 100

D.3.4. Figure 4

Script → `hyperparameter_selection.py`
dataset → Amazon/MNIST
domain → *'Books', 'Kitchen', 'DVD', 'Elec'/'mnist-like', 'ciphar'*
α → 0.01
p → 100