High Dimensional Learning with Noisy Labels

Aymane El Firdoussi*
Technology Innovation Institute
Abu Dhabi, UAE

Mohamed El Amine Seddik*
Technology Innovation Institute
Abu Dhabi, UAE

Abstract

This paper provides theoretical insights into high-dimensional binary classification with class-conditional noisy labels. Specifically, we study the behavior of a linear classifier with a label noisiness aware loss function, when both the dimension of data p and the sample size n are large and comparable. Relying on random matrix theory by supposing a Gaussian mixture data model, the performance of the linear classifier when $p, n \to \infty$ is shown to converge towards a limit, involving scalar statistics of the data. Importantly, our findings show that the low-dimensional intuitions to handle label noise do not hold in high-dimension, in the sense that the optimal classifier in low-dimension dramatically fails in high-dimension. Based on our derivations, we design an optimized method that is shown to be provably more efficient in handling noisy labels in high dimensions. Our theoretical conclusions are further confirmed by experiments on real datasets, where we show that our optimized approach outperforms the considered baselines.

1 INTRODUCTION

Machine learning methods are usually built upon low-dimensional intuitions, which do not necessarily hold when processing high-dimensional data. Numerous studies have demonstrated the effects of the curse of dimensionality, by showing that high dimensions can alter the internal functioning of various ML methods designed with low-dimensional intuitions. Classical examples include spectral methods (Couillet and Benaych-Georges, 2016), empirical risk minimization frameworks (El Karoui et al., 2013; Mai and Liao, 2019), transfer & multi-task learning (Tiomoko et al., 2021), deep learning theory with the double descent phenomena (Nakkiran et al., 2021; Mei and Montanari, 2022) and many other works. In all this literature, random matrix theory (RMT) played a central role in deciphering the high-dimensional effects by supposing the so-called RMT regime where both the dimension of data and the sample size are supposed to be large and comparable. We refer the reader to (Bai and Silverstein, 2010) for a general overview on the spectral analysis of large random matrices, and to (Couillet and Liao, 2022) for specific applications of RMT in the realm of machine learning.

In this paper, we aim at exploring the high-dimensional effects on learning with noisy labels. Based on the framework of Natarajan et al. (2018), who derived an unbiased classifier when faced with a binary classification problem with class-conditional noisy labels, we introduce a **Labels-Perturbed Classifier (LPC)** that is essentially a Ridge classifier with parameterized labels. The introduced classifier encapsulates different variants depending on the choice of the label parameters including the unbiased method of Natarajan et al. (2018). Considering a Gaussian mixture data model and supposing a high-dimensional regime, we conduct an RMT analysis of LPC by characterizing the distribution of its decision function and deriving its theoretical test performance in terms of both accuracy and risk. Our analysis allows us to gain insight when learning with noisy labels, and more importantly design an optimized classifier that surprisingly outperforms the unbiased classifier of Natarajan et al. (2018) in high dimensions, even approaching the performance of an oracle classifier that is trained with the correct labels. Through this analysis, we demonstrate again that methods designed with low-dimensional intuitions can dramatically fail in high-dimensions, and careful refinements are needed to de-

*Equal contribution. Work done by first author while at TII.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

sign more robust and interpretable methods. Our theoretical findings are also validated on real data where we show consistent improvements under high label noise.

Summary of contributions: Our contributions can be summarized as follows:

- We show that the previously introduced *Unbiased classifier* in Natarajan et al. (2018) is not optimal when the dimension p of features is comparable to the number of training sample n (i.e high-dimension), and therefore shedding light on the weakness of analyzing ML models when assuming an infinite sample size regime ($n \gg p$).
- We introduce our **Labels-Perturbed Classifier** (LPC) that provably outperforms any other classifier in the same setting for specific parameter values (that we find theoretically), and provide detailed theoretical proofs of our results by leveraging tools from Random Matrix Theory.
- Using our theoretical results, we design a method to estimate the label noise present in the data, a quantity that is unknown a priori.

The remainder of the paper is organized as follows. Section 2 presents related work in the realm of learning with noisy labels. Our setting and main assumptions along with essential RMT notions are presented in Section 3. The main results brought by this paper are deferred to Section 4. In Section 5 we conduct experiments to validate our findings on real data. Finally, Section 6 concludes the paper and discusses future extensions. All our proofs are deferred to the Appendix.

2 RELATED WORK

Numerous studies have been conducted to investigate supervised learning under noisy labels, spanning both theoretical and empirical approaches. These studies range from learning theory and statistical perspectives to practical implementations using neural networks and deep learning techniques.

Key contributions in this field include: *Bayesian Approaches:* Graepel and Herbrich (2000) conducted a Bayesian study on learning with noisy labels. Lawrence and Schölkopf (2001) estimated noise levels in kernel-based learning, a work that was later extended by Li et al. (2007) who incorporated a probabilistic noise model into the Kernel Fisher discriminant and relaxed distribution assumptions. *Robust Optimization Approaches:* Freund (2009) proposed a robust boosting algorithm using a non-convex potential, which demonstrated empirical resilience against random label noise. Jiang (2001) provided a survey of theoretical results on boosting with noisy labels. *Model-Specific Robustness:* Biggio et al. (2011) explored the robustness of SVMs under adversarial label noise and proposed a kernel matrix correction method to enhance robustness. *Algorithmic Innovations:* Several noise-tolerant versions of the perceptron algorithm have been developed, including Passive-aggressive algorithms (Crammer et al., 2006), Confidence-weighted learning (Dredze et al., 2008), AROW (Crammer et al., 2009), and NHERD algorithm (Crammer and Lee, 2010). *Deep Learning Approaches:* Recent works have utilized deep learning techniques to address noisy labels. For example, Li et al. (2020) introduced Dividemix, a semi-supervised learning algorithm for learning with noisy labels. Ma et al. (2018) studied the generalization behavior of deep neural networks (DNNs) for noisy labels in terms of intrinsic dimensionality, proposing a Dimensionality-Driven Learning (D2L) strategy to avoid overfitting. Tanaka et al. (2018) addressed noisy labels in computer vision contexts, while Karimi et al. (2020) applied these techniques to medical imaging.

Our work is closely related to the studies in (Natarajan et al., 2013, 2018), which consider adaptive loss functions and assume the prior knowledge of the noise rates. Scott et al. (2013) do not make this assumption and model the true distribution as satisfying a mutual irreducibility property, then estimating mixture proportions by maximal denoising of noisy distributions. Manwani and Sastry (2013) investigated the impact of the loss function on noise tolerance, showing that empirical risk minimization under the 0-1 loss has robust properties, while the squared loss is noise-tolerant only under uniform noise. For a comprehensive overview of the field, readers can refer to the surveys (Song et al., 2022a,b) on learning with noisy labels.

3 STATISTICAL MODEL

3.1 Binary classification with noisy labels

We consider that we are given a sequence of n i.i.d p-dimensional training data $\boldsymbol{x}_1, ..., \boldsymbol{x}_n \in \mathbb{R}^p$ with corresponding correct labels $y_1, ..., y_n = \pm 1$. We consider a noisy label setting where the true labels y_i's are flipped randomly, yielding a noisy dataset $(\boldsymbol{x}_i, \tilde{y}_i)_{i\in[n]}$ such that:

$$\begin{cases} \mathbb{P}\{\tilde{y}_i = -1 \mid y_i = +1\} = \varepsilon_+, \\ \mathbb{P}\{\tilde{y}_i = +1 \mid y_i = -1\} = \varepsilon_-. \end{cases} \qquad \varepsilon_+ + \varepsilon_- < 1.$$

We suppose that $\boldsymbol{x}_i$ is sampled from a Gaussian mixture of two distinct isotropic clusters $\mathcal{C}_1$ and $\mathcal{C}_2$, i.e.,

for $a \in \{1, 2\}$:

$$\boldsymbol{x}_i \in \mathcal{C}_a \quad \Leftrightarrow \quad \begin{cases} \boldsymbol{x}_i = \boldsymbol{\mu}_a + \boldsymbol{z}_i, & \boldsymbol{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p), \\ y_i = (-1)^a. \end{cases} \tag{1}$$

For convenience and without loss of generality, we further assume that $\boldsymbol{\mu}_a = (-1)^a \boldsymbol{\mu}$ for some vector $\boldsymbol{\mu} \in \mathbb{R}^p$. This setting can be recovered by subtracting $\frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}$ from each data point, as such $\boldsymbol{\mu} = \frac{\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1}{2}$ and therefore the SNR $\|\boldsymbol{\mu}\|$ controls the difficulty of the classification problem, in the sense that large values of $\|\boldsymbol{\mu}\|$ yield a simple classification problem whereas when $\|\boldsymbol{\mu}\| \to 0$, the classification becomes impossible.

Remark 3.1 (On the data model)**.** *Note that the above data assumption can be relaxed to considering $\boldsymbol{x}_i = \boldsymbol{\mu}_a + \mathbf{C}_a^{\frac{1}{2}} \boldsymbol{z}_i$ where $\mathbf{C}_a$ is some semi-definite covariance matrix and $\boldsymbol{z}_i$ are random vectors with i.i.d entries of mean 0, variance 1 and bounded fourth order moment. In fact, in the high-dimensional regime when $n, p \to \infty$, the asymptotic performance of the classifier considered subsequently is universal in the sense that it depends only on the statistical means and covariances of the data (Louart and Couillet, 2018; Seddik et al., 2020; Dandi et al., 2024). However, such a general setting comes at the expense of more complex formulas, making the above isotropic assumption more convenient for readability and better interpretation of our findings. We provide a more general result of our main result (Theorem 4.2) by considering arbitrary covariance matrices in the Appendix (Theorem 2.2).*

Naive approach. Given the noisy dataset $(\boldsymbol{x}_i, \tilde{y}_i)_{i \in [n]}$ as per (1), a naive learning approach would consist in ignoring the noisiness of the labels and training a given classifier by minimizing a regularized loss function as follows:

$$\mathcal{L}_0(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \ell(\boldsymbol{w}^\top \boldsymbol{x}_i, \tilde{y}_i) + \gamma \|\boldsymbol{w}\|^2,$$

where ℓ is some convex loss function, and $\gamma \geq 0$ is a regularization parameter.

Improved approach. Natarajan et al. (2018) proposed an unbiased approach which takes into account the noisiness of the labels. Specifically, given any bounded loss function $\ell(s, y)$, their approach consists in considering:

$$\tilde{\ell}(s, y) \equiv \frac{(1 - \varepsilon_{-y})\ell(s, y) - \varepsilon_y \ell(s, -y)}{1 - \varepsilon_+ - \varepsilon_-}. \tag{2}$$

where $\varepsilon_y = \varepsilon_+$ if $y = 1$ and $\varepsilon_y = \varepsilon_-$ otherwise.
The main intuition behind this proposition is that this loss has the nice property of being an unbiased estimator of the loss $\ell(s, y)$ on the correct dataset $(\boldsymbol{x}_i, y_i)_{i \in [n]}$, since it satisfies for any s, y:

$$\mathbb{E}_{\tilde{y}}[\tilde{\ell}(s, \tilde{y})] = \ell(s, y).$$

Our approach We consider a parametrization of the previous loss (2) by introducing scalar parameters $\rho_\pm$ to be optimized, rather than $\varepsilon_\pm$:

$$\tilde{\ell}(s, y, \rho) \equiv \frac{(1 - \rho_{-y})\ell(s, y) - \rho_y \ell(s, -y)}{1 - \rho_+ - \rho_-}. \tag{3}$$

where $\rho_+, \rho_- \in \mathbb{R}$ and $\rho_+ + \rho_- \neq 1$. Hence, our global optimization problem becomes:

$$\mathcal{L}_\rho(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \tilde{\ell}(\boldsymbol{w}^\top \boldsymbol{x}_i, \tilde{y}_i, \rho) + \gamma \|\boldsymbol{w}\|^2. \tag{4}$$

Taking a generic loss function ℓ will lead to an intractable solution $\boldsymbol{w}_\rho$ to (4). However, Mai and Liao (2024) show that taking the squared loss (L^2) is compatible with a Gaussian Mixture Model in a classification setting, and more importantly, it gives a **closed-form** solution to our problem that we can study its performance using Random Matrix Theory. Therefore, taking $\ell(\boldsymbol{w}^\top \boldsymbol{x}_i, \tilde{y}_i) = (\boldsymbol{w}^\top \boldsymbol{x}_i - \tilde{y}_i)^2$, the solution of (4) defines our **Labels-Perturbed Classifier (LPC)**, which is given by:

$$\boldsymbol{w}_\rho = \frac{1}{n} \mathbf{Q}(\gamma) \mathbf{X} \mathbf{D}_\rho \tilde{\boldsymbol{y}}, \quad \mathbf{Q}(z) = \left(\frac{1}{n} \mathbf{X}\mathbf{X}^\top + z \mathbf{I}_p \right)^{-1} \tag{5}$$

where $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$, $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \ldots, \tilde{y}_n)^\top \in \mathbb{R}^n$ and $\mathbf{D}_\rho$ is a diagonal matrix defined as $\mathbf{D}_\rho = \mathrm{Diag}\left(\frac{1 - \rho_{-\tilde{y}_i} + \rho_{\tilde{y}_i}}{1 - \rho_+ - \rho_-} \mid i \in [n] \right) \in \mathcal{D}_n$.

Remark 3.2. *(Extension of our results to other loss functions) Although we use the squared loss in our analysis, our results still hold for any other loss function ℓ, such as the Binary Cross-Entropy loss, as we have shown in Figures 3 and 4 in the Appendix.*

In the remainder, we will study the performance of $\boldsymbol{w}_\rho$ which encapsulates the following cases:

- **Naive Classifier:** which corresponds to $\rho_\pm = 0$.
- **Unbiased Classifier:** by taking $\rho_\pm = \varepsilon_\pm$ as introduced by Natarajan et al. (2018).
- **Optimized Classifier:** by optimizing $\rho_\pm$ to maximize the theoretical test accuracy.
- **Oracle Classifier:** which corresponds to training on the correct labels, i.e., $\rho_\pm = \varepsilon_\pm = 0$.

We aim to characterize the asymptotic performance (i.e., test accuracy and risk) of LPC in the high-dimensional regime where both the sample size n and the data dimension p grow large at a comparable rate, which corresponds to the classical random matrix theory (RMT) regime. Specifically, our analysis confirms that the *unbiased* classifier outperforms the *naive* classifier in a low-dimensional regime, i.e., when $n \gg p$. In contrast, when considering the high-dimensional regime, we show that the *unbiased* classifier becomes sub-optimal and we provide an *optimized* classifier that consists of maximizing the derived test accuracy w.r.t the scalars $\rho_{\pm}$ yielding a closed-form solution. This sheds light on the fact that low-dimensional intuitions do not necessarily hold for high dimensions and careful refinements should be considered to enhance the performance of simple algorithms in these settings.

3.2 RMT Background

In mathematical terms, the understanding of the asymptotic performance of the classifier $\boldsymbol{w}_\rho$ boils down to the characterization of the statistical behavior of the *resolvent matrix* $\mathbf{Q}(z)$ introduced in (5). In the following, we will recall some important notions and results from random matrix theory which will be at the heart of our analysis. We start by defining the main object which is the resolvent matrix.

Definition 3.3 (Resolvent)**.** *For a symmetric matrix $\mathbf{M} \in \mathbb{R}^{p\times p}$, the resolvent $\mathbf{Q}_M(z)$ of $\mathbf{M}$ is defined for $z \in \mathbb{C}\backslash\mathcal{S}(\mathbf{M})$ as:*

$$\mathbf{Q}_M(z) = (\mathbf{M} - z\mathbf{I}_p)^{-1},$$

where $\mathcal{S}(\mathbf{M})$ is the set of eigenvalues or spectrum of $\mathbf{M}$.

The matrix $\mathbf{Q}_M(z)$ will often be denoted $\mathbf{Q}(z)$ or $\mathbf{Q}$ when there is no ambiguity. In fact, the study of the asymptotic performance of $\boldsymbol{w}_\rho$ involves the estimation of linear forms of the resolvent $\mathbf{Q}$ in (5), such as $\frac{1}{n}\operatorname{Tr}\mathbf{Q}$ and $\boldsymbol{a}^\top\mathbf{Q}\boldsymbol{b}$ with $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^p$ of bounded Euclidean norms. Therefore, the notion of a *deterministic equivalent* (Hachem et al., 2007) is crucial as it allows the design of a deterministic matrix, having (in probability or almost surely) asymptotically the same *scalar observations* as the random ones in the sense of *linear forms*. A rigorous definition is provided below.

Definition 3.4 (Deterministic equivalent)**.** *We say that $\bar{\mathbf{Q}} \in \mathbb{R}^{p\times p}$ is a deterministic equivalent for the random resolvent matrix $\mathbf{Q} \in \mathbb{R}^{p\times p}$ if, for any bounded linear form $u : \mathbb{R}^{p\times p} \to \mathbb{R}$, we have that, as $p \to \infty$:*

$$u(\mathbf{Q}) \xrightarrow{a.s.} u(\bar{\mathbf{Q}}),$$

where the convergence is in the almost sure sense.

In particular, a deterministic equivalent for the resolvent $\mathbf{Q}(z)$ defined in (5) is given by the following Lemma, a result that is brought from (Louart and Couillet, 2018).

Lemma 3.5 (Deterministic equivalent of the resolvent)**.** *Under the high-dimensional regime, when $p, n \to \infty$ with $\frac{p}{n} \to \eta \in [0, \infty)$ and assuming $\|\boldsymbol{\mu}\| = \mathcal{O}(1)$. A deterministic equivalent for $\mathbf{Q} \equiv \mathbf{Q}(\gamma)$ as defined in (5) is given by:*

$$\bar{\mathbf{Q}} = \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p}{1+\delta} + \gamma\mathbf{I}_p\right)^{-1},$$

where:

$$\delta = \frac{1}{n}\operatorname{Tr}\bar{\mathbf{Q}} = \frac{\eta - \gamma - 1 + \sqrt{(\eta-\gamma-1)^2 + 4\eta\gamma}}{2\gamma}.$$

In a low-dimensional setting, i.e. when p being fixed while $n \to \infty$, the resolvent $\mathbf{Q}$ converges almost surely to $\left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + (1+\gamma)\mathbf{I}_p\right)^{-1}$ which is also covered by Lemma 3.5 as $\delta \to 0$ in this setting. However, when both p and n are large and comparable, the data dimension induces a bias which is captured by the quantity δ as it becomes $\mathcal{O}(1)$ in the RMT regime. We will highlight in the following that this bias alters the behavior of the classifier $\boldsymbol{w}_\rho$ in high dimensions, in particular, making the *unbiased* classifier $\boldsymbol{w}_\varepsilon$ introduced by Natarajan et al. (2018) unexpectedly sub-optimal when learning with noisy labels in high-dimensions.

4 MAIN RESULTS

4.1 Asymptotic Behavior of the Labels-Perturbed Classifier (LPC)

We are now in place to present our main technical result which describes the asymptotic behavior of LPC as defined in (5). Specifically, we provide our results under the following growth rate assumptions.

Assumption 4.1 (Growth Rates)**.** *Suppose that as $p, n \to \infty$:*

$$1)\ \frac{p}{n} \to \eta \in (0, \infty), \qquad 2)\ \frac{n_a}{n} \to \pi_a \in (0, 1),$$
$$3)\ \|\boldsymbol{\mu}\| = \mathcal{O}(1),$$

where n_a denotes the cardinality of the class $\mathcal{C}_a$ for $a \in [2]$.

The third condition about the magnitude of the mean vector $\boldsymbol{\mu}$ reflects the fact that the classification problem should neither be trivial ($\|\boldsymbol{\mu}\| \gg 1$) nor impossible ($\|\boldsymbol{\mu}\| \to 0$) as the dimension of data grows large. For instance, assuming $\|\boldsymbol{\mu}\|$ of order $O(\sqrt{p})$ would not be relevant as $p \to \infty$ since the classification problem becomes trivial in this regime. We refer the reader to

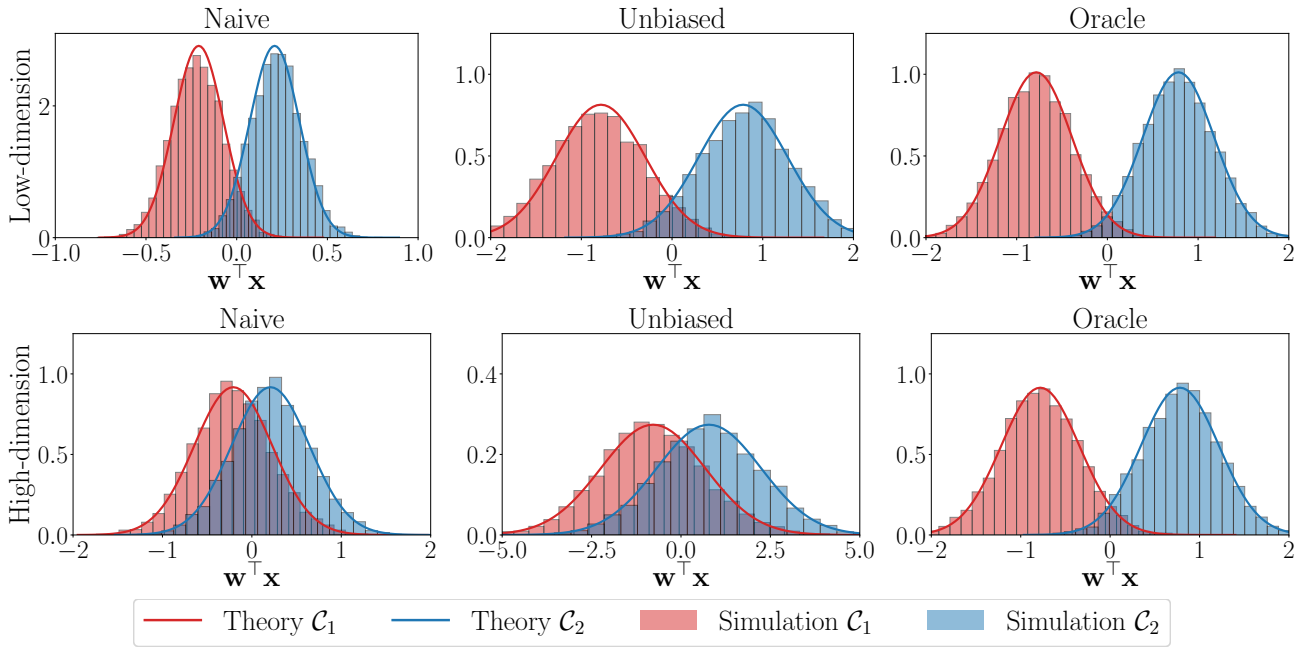

Figure 1: Distribution of the decision function $\boldsymbol{w}_\rho^\top \boldsymbol{x}$ of different variants of LPC for $n = 5000$, $\pi_1 = \frac{1}{3}$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$, $\|\boldsymbol{\mu}\| = 2$, $\gamma = 0.1$, $p = 50$ (first row) and $p = 1000$ (second row). The theoretical Gaussian distributions are predicted as per Theorem 4.2. Note that the variance of the decision function for the *unbiased* classifier increases with the dimension yielding poor accuracy.

(Couillet and Benaych-Georges, 2016) for a more general formulation and justifications of this assumption under an extended k-class Gaussian mixture model.

Further, define the following quantities which will be used subsequently:

$$\lambda_- = \frac{1-\rho_+ + \rho_-}{1-\rho_+ - \rho_-},\ \lambda_+ = \frac{1-\rho_- + \rho_+}{1-\rho_+ - \rho_-}, \tag{6}$$

$$\beta = \frac{1}{1-\rho_+ - \rho_-},\ h = 1 - \frac{\eta}{(1+\gamma(1+\delta))^2} \tag{7}$$

$$\alpha = (\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)). \tag{8}$$

Our main result is therefore given by the following theorem.

Theorem 4.2 (Performance of LPC)**.** *Let $\boldsymbol{w}_\rho$ be the LPC as defined in (5) and suppose that Assumption 4.1 holds. The decision function $\boldsymbol{w}_\rho^\top \boldsymbol{x}$, on some test sample $\boldsymbol{x} \in \mathcal{C}_a$ independent from $\mathbf{X}$, satisfies:*

$$\boldsymbol{w}_\rho^\top \boldsymbol{x} \xrightarrow{\mathcal{D}} \mathcal{N}\left((-1)^a m_\rho,\ \nu_\rho - m_\rho^2\right),$$

where:

$$m_\rho = \frac{\pi_1(\lambda_- - 2\beta\varepsilon_-) + \pi_2(\lambda_+ - 2\beta\varepsilon_+)}{\alpha}\|\boldsymbol{\mu}\|^2,$$

$$\begin{aligned}\nu_\rho = {} & \frac{(\pi_1(2\beta\varepsilon_- - \lambda_-) + \pi_2(2\beta\varepsilon_+ - \lambda_+))^2}{h\alpha} \times \\ & \left(\frac{\|\boldsymbol{\mu}\|^2 + 1}{\alpha} - 2(1-h)\right)\|\boldsymbol{\mu}\|^2 \\ & + \frac{(1-h)}{h}\pi_1(4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2) \\ & + \frac{(1-h)}{h}\pi_2(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2).\end{aligned}$$

Moreover, the asymptotic test accuracy of the classifier is given by $\Phi\left((\nu_\rho - m_\rho^2)^{-\frac{1}{2}} m_\rho\right)$ where $\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt$.

In a nutshell, Theorem 4.2 states that LPC is asymptotically equivalent to the thresholding of two monovariate Gaussian random variables with respective means $-m_\rho$ and m_ρ and second moment ν_ρ, where these statistics express in terms of the different parameters in our setting. Essentially, Theorem 4.2 allows us to draw interpretations on the behavior of the different classifiers described earlier. First, let us start by defining the statistics for the *oracle* classifier which corresponds to setting $\rho_\pm = \varepsilon_\pm = 0$, yielding:

$$m_{\text{oracle}} = \frac{\|\boldsymbol{\mu}\|^2}{\alpha},\quad \nu_{\text{oracle}} = \kappa + \frac{1-h}{h}, \tag{9}$$

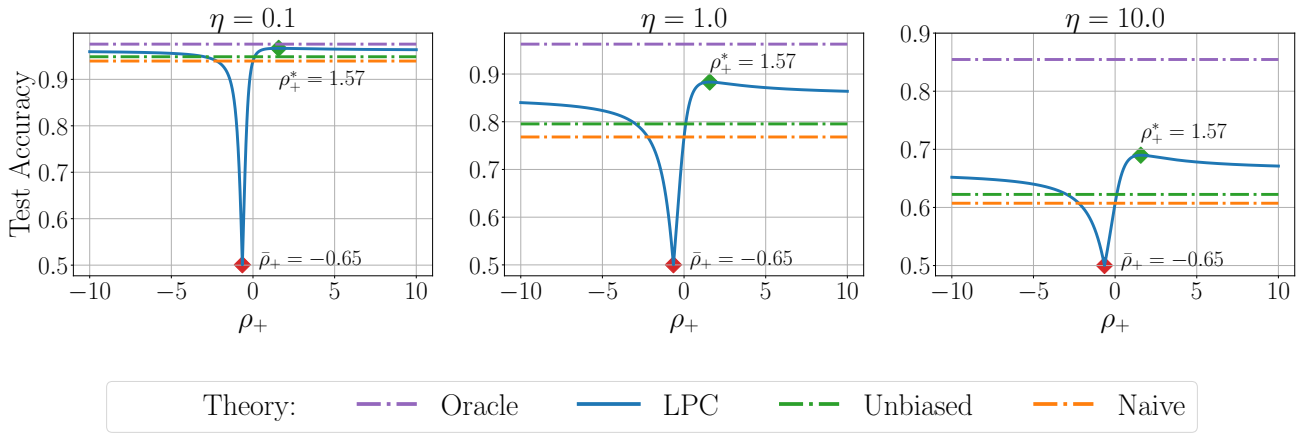

Figure 2: Test accuracy of LPC by fixing $\rho_- = 0$ and varying ρ_+. We considered $n = 1000$, $\pi_1 = 0.3$, $\|\boldsymbol{\mu}\| = 2$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$ and optimal γ. We notice that the test accuracy is maximized at ρ_+^* yielding better accuracy compared with the *unbiased* approach. Note that for small values of η, i.e., for low dimensions, the test accuracy becomes flat in terms of ρ_+ and in the limit $\eta \to 0$ the maximizer ρ_+^* is not identifiable as discussed in Remark 4.3.

where

$$\kappa = \frac{1}{h\alpha}\left(\frac{\|\boldsymbol{\mu}\|^2+1}{\alpha} - 2(1-h)\right)\|\boldsymbol{\mu}\|^2.$$

Therefore, the statistics of the decision functions for the *naive* ($\rho_\pm = 0$) and *unbiased* ($\rho_\pm = \varepsilon_\pm$) classifiers are expressed respectively as follows:

$$\textit{Naive}\begin{cases} m_{\text{naive}} &= (1-2(\pi_1\varepsilon_- + \pi_2\varepsilon_+))\cdot m_{\text{oracle}}, \\ \nu_{\text{naive}} &= (1-2(\pi_1\varepsilon_- + \pi_2\varepsilon_+))^2\cdot\kappa + \frac{1-h}{h}. \end{cases}$$

$$\textit{Unbiased}\begin{cases} m_{\text{unbiased}} &= m_{\text{oracle}}, \\ \nu_{\text{unbiased}} &= \kappa + \frac{1-h}{h}\theta. \end{cases}$$

with:

$$\theta = \pi_1(4\beta^2\varepsilon_-(\varepsilon_+-\varepsilon_-)+\lambda_-^2)+\pi_2(4\beta^2\varepsilon_+(\varepsilon_--\varepsilon_+)+\lambda_+^2)$$

From these quantities, we can explain the behavior of the different classifiers in the low-dimensional versus high-dimensional regimes. In fact, when $n \gg p$ the dimensions ratio $\eta \to 0$ implies that $h \to 1$ as per (6). Therefore, in the low-dimensional setting, the *unbiased* classifier statistics match those of the *oracle* as expected. However, in the high-dimensional regime, i.e., when $h \neq 1$, while the *unbiased* classifier remains unbiased, the second moment gets amplified due to label noise, resulting in a larger variance compared with the *oracle* classifier. Indeed, we have:

$$m_{\text{unbiased}} - m_{\text{oracle}} = 0,$$
$$\nu_{\text{unbiased}} - \nu_{\text{oracle}} = \frac{1-h}{h}(\pi_1(4\beta^2\varepsilon_-(\varepsilon_+-\varepsilon_-)+\lambda_-^2)+$$
$$\pi_2(4\beta^2\varepsilon_+(\varepsilon_--\varepsilon_+)+\lambda_+^2)-1) \neq 0.$$

This behavior is highlighted in Figure 1 which depicts the histogram of the decision function for the different classifiers along with the theoretical Gaussian distributions as per Theorem 4.2, in both the low-dimensional and high-dimensional settings.

Interestingly, when observing the asymptotic test accuracy in terms of ρ_+ and ρ_- as depicted in Figure 2, we remarkably find that the accuracy is maximized for any fixed ρ_- at some value $\rho_+^*(\rho_-)$, and the maximum accuracy is higher than the *unbiased* accuracy in high-dimension. Moreover, since $\varphi(\cdot)$ is monotonous, such maximizer can be obtained analytically by maximizing the ratio $(\nu_\rho - m_\rho^2)^{-\frac{1}{2}}m_\rho$ as derived in Appendix section 4, which yields the following closed-form expression:

$$\boxed{\rho_+^*(\rho_-) = \frac{\pi_1^2\varepsilon_-(\varepsilon_- - 1) + \pi_2^2\varepsilon_+(1-\varepsilon_+)}{\pi_1\pi_2(1-\varepsilon_+-\varepsilon_-)} + \rho_-.} \tag{10}$$

Therefore, our *optimized classifier* is defined by taking $\rho_- = 0$ and $\rho_+ = \rho_+^*(0)$ in the expression of $\boldsymbol{w}_\rho$ as per (5). We notably notice that ρ_+^* depends solely on the noise probabilities $\varepsilon_\pm$ and the class proportions π_1 and π_2, especially, it does not involve the SNR $\|\boldsymbol{\mu}\|$, the regularization γ and the dimension ratio η which is quite unexpected. We also notice that the worst performance of LPC with parameters $\bar{\rho}_+, \bar{\rho}_-$ (again $\bar{\rho}_-$ can be fixed to 0) corresponds to the one of a random guess and can be derived by solving $m_\rho = 0$ which

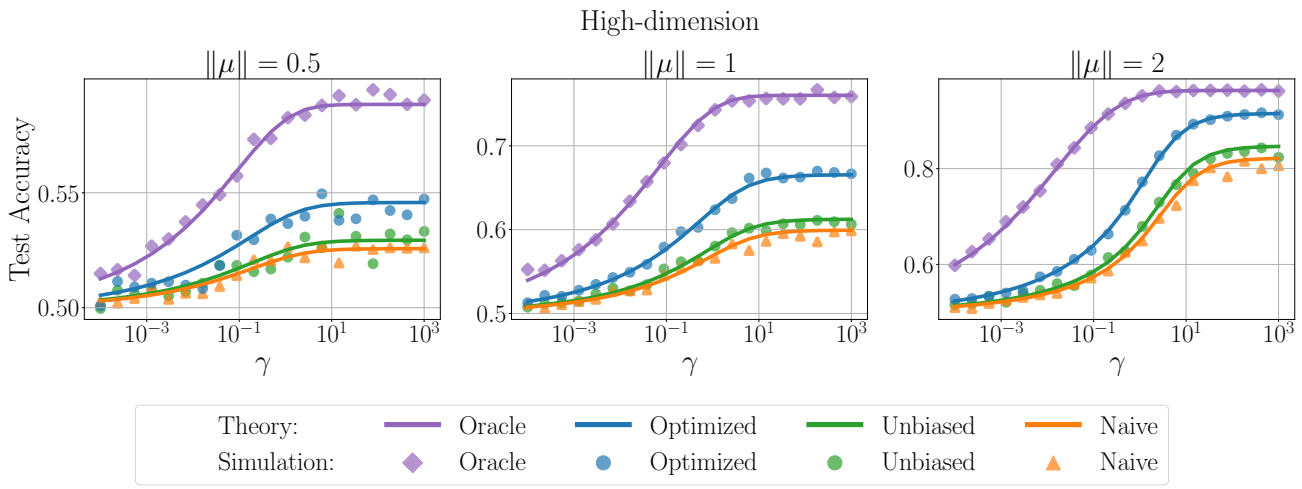

Figure 3: Empirical versus theoretical test accuracy for different variants of LPC. We used $(n, p = 200, 200)$, $\pi_1 = 0.3$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$ and varied γ.

yields (for $\pi_1 \neq \frac{1}{2}$):

$$\bar{\rho}_+(\rho_-) = \frac{1 - 2\pi_1\varepsilon_- - 2\pi_2\varepsilon_+}{2\pi_1 - 1} + \rho_-. \tag{11}$$

Figure 3 depicts both the empirical and theoretical test performance of LPC and its different variants, where we essentially notice a very accurate matching between simulation and the theoretical predictions even for a finite sample size, plus a dominance of our *optimized* approach compared to the other ones. See also Figure 1 in the Appendix for more plots varying other parameters. In fact, even though we work under an asymptotic regime, our estimation of the test performance is consistent, as it can be shown that their fluctuations are of order $n^{-\frac{1}{2}}$ under Assumption 4.1, this is a consequence of the concentration results of the resolvent $\mathbf{Q}$ as shown in (Louart and Couillet, 2018).

Remark 4.3 (On the relevance of the RMT analysis)**.** *Our RMT analysis relies on the main assumption that both p and n are large and comparable as per Assumption 4.1. This assumption is in fact fundamentally crucial for exhibiting the maximizer ρ_+^* defined above. Indeed, supposing an infinite sample size setting where p is fixed while taking only $n \to \infty$ or alternatively $h \to 1$, would result in $(\nu_\rho - m_\rho^2)^{-\frac{1}{2}} m_\rho \to \|\boldsymbol{\mu}\|$. Therefore, the existence of ρ_+^* is only tractable under the large dimensional setting, which motivates the importance of this assumption.*

4.2 Estimation of label noise probabilities

Another important aspect of our *optimized* classifier is the fact that it supposes the prior knowledge of the noise probabilities $\varepsilon_\pm$ which is also the case for the *unbiased* classifier of (Natarajan et al., 2018). In this section, based on our theoretical derivations, we propose a simple procedure for estimating $\varepsilon_\pm$ by supposing that the SNR $\|\boldsymbol{\mu}\|$ and the class proportions π_1, π_2 are known, in fact the latest can be consistently estimated with very few training samples as described in (Tiomoko et al., 2021).

To estimate $\varepsilon_\pm$, we rely on the expression of the second moment $\nu_\rho = \nu_\rho(\varepsilon_+, \varepsilon_-)$ as per Theorem 4.2, by viewing it as a function of $\varepsilon_\pm$. Specifically, we consider two different arbitrary couples $\rho_1 = (\rho_+^1, \rho_-^1)$ and $\rho_2 = (\rho_+^2, \rho_-^2)$ and solve the system:

$$\begin{cases} \hat{\nu}_{\rho_1} = \nu_{\rho_1}(\varepsilon_+, \varepsilon_-), \\ \hat{\nu}_{\rho_2} = \nu_{\rho_2}(\varepsilon_+, \varepsilon_-). \end{cases} \tag{12}$$

where $\hat{\nu}_\rho = \frac{1}{n}\sum_{i=1}^n (\boldsymbol{x}_i^\top \boldsymbol{w}_\rho^{-i})^2$ is the empirical estimate of ν_ρ and $\boldsymbol{w}_\rho^{-i}$ corresponds to LPC trained on all examples except $\boldsymbol{x}_i$, which discards the statistical dependencies. Figure 2 (Appendix) depicts the estimated versus ground truth value of ε_+ and shows consistent estimation for different values of the SNR $\|\boldsymbol{\mu}\|$.

5 EXPERIMENTS

In this section, we present experiments with real data to validate our approach. We use the Amazon review dataset (Blitzer et al., 2007) which includes several binary classification tasks corresponding to positive versus negative reviews of `books`, `dvd`, `electronics` and `kitchen`. We apply the standard scaler from `sklearn` (Pedregosa et al., 2011) and estimate $\|\boldsymbol{\mu}\|$ with the normalized data.

Note that our *optimized* classifier does not require the knowledge of $\|\boldsymbol{\mu}\|$ since ρ_+^* depends only on the class proportions π_a's and the noise probabilities $\varepsilon_\pm$ as per

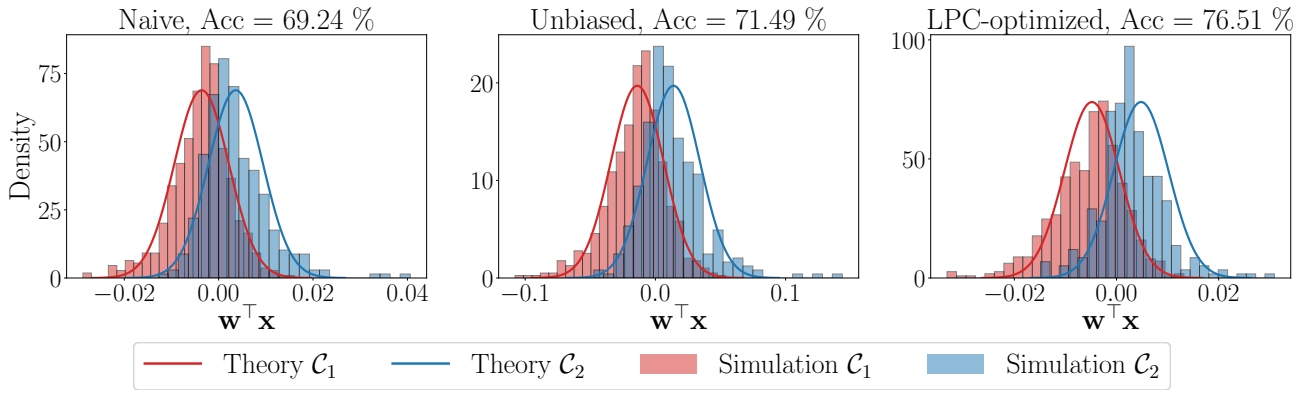

Figure 4: Histogram of the decision function of different LPC variants on the `books` dataset (Blitzer et al., 2007), along with the theoretical distribution as predicted by Theorem 4.2. We considered $n = 1600$, $p = 400$, $\pi_1 = 0.3$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$ and optimal γ.

Table 1: Accuracy comparison over Amazon review datasets (Blitzer et al., 2007) for $n = 1600$, $p = 400$, $\pi_1 = 0.3$, $\varepsilon_- = 0.4$ and optimal γ. As theoretically anticipated, our *optimized* approach yields better classification accuracy even approaching *oracle* trained with the true labels.

ε_+	**Sub-Dataset**	**Naive** (%)	**Unbiased** (%)	**Optimized** (%)	**Oracle** (%)
0.3	`Books`	72.69 ± 0.11	71.66 ± 0.25	76.36 ± 0.21	78.78 ± 0.07
	`Dvd`	73.75 ± 0.42	72.24 ± 0.3	77.43 ± 0.04	80.57 ± 0.12
	`Electronics`	78.22 ± 0.05	77.22 ± 0.09	81.57 ± 0.12	83.22 ± 0.09
	`Kitchen`	79.64 ± 0.07	78.62 ± 0.05	82.17 ± 0.06	84.28 ± 0.1
0.4	`Books`	66.84 ± 0.31	66.68 ± 0.22	75.69 ± 0.22	78.78 ± 0.07
	`Dvd`	67.2 ± 0.37	67.33 ± 0.34	76.86 ± 0.16	80.57 ± 0.12
	`Electronics`	72.13 ± 0.18	72.36 ± 0.06	81.04 ± 0.08	83.22 ± 0.09
	`Kitchen`	73.46 ± 0.29	73.85 ± 0.23	81.65 ± 0.17	84.28 ± 0.1
0.5	`Books`	55.37 ± 0.25	59.5 ± 0.43	75.26 ± 0.19	78.78 ± 0.07
	`Dvd`	55.32 ± 0.41	59.68 ± 0.57	76.42 ± 0.13	80.57 ± 0.12
	`Electronics`	57.96 ± 0.11	63.21 ± 0.36	80.73 ± 0.01	83.22 ± 0.09
	`Kitchen`	58.15 ± 0.61	64.71 ± 0.7	81.32 ± 0.11	84.28 ± 0.1

(10). However, if the latest quantities are unknown, one can estimate them as we discussed in the previous section and therefore the knowledge of $\|\boldsymbol{\mu}\|$ is required, but can also be consistently estimated with few data samples as discussed earlier.

Moreover, as theoretically anticipated, the *optimized classifier* outperforms the *naive* and *unbiased* classifiers in terms of accuracy. Table 1 shows the performance in terms of classification accuracy of the different classifiers, on different datasets and varying the noise probabilities. We clearly observe that the *optimized* approach yields spectacular performances which are almost close to the *oracle* that assumes perfect knowledge of the true labels, even under a high noise regime. The code is provided in the supplementary material for reproducibility of our empirical results.

6 CONCLUSION & FUTURE DIRECTIONS

This paper introduced new insights into learning with noisy labels in high dimensions. Relying on tools from random matrix theory, we provided an asymptotic characterization of the performance of the introduced classifier which accounts for label noise through scalar quantities. Based on this analysis, we identified that the low-dimensional intuitions to handle label noise do not extend to high-dimension and developed a new approach that is proven to be more efficient by design. We also showed through empirical evidence that our approach yields improved performance on real data.

In our current investigation, we restricted our analysis to the cases of squared loss and binary classification.

Our results can be extended beyond these settings by accounting for a general bounded loss function $\ell(s, y)$ and multi-class classification problems. We provide in the Appendix (section 5) some experiments with synthetic and real data using the binary-cross-entropy loss function that show similar behavior to the squared loss (see Figures 3 and 4), namely, the existence of an optimum $\rho_{\pm}^{*}$ that outperforms the *unbiased* approach in high dimensions. The extension of our study to this setting can be performed by leveraging the empirical risk minimization framework (El Karoui et al., 2013; Mai and Liao, 2019) which allows the RMT analysis of general loss functions. Moreover, as we provided in Appendix section 6, our results extend to a k-class classification setting where we empirically show improved performance by optimizing a set of $2k$ scalar parameters (which play the same role as $\rho_{\pm}$ of the binary case). Such extension is straightforward in the case of squared loss given our current results and will be addressed in future work.

References

Zhidong Bai and Jack W Silverstein. *Spectral analysis of large dimensional random matrices*, volume 20. Springer, 2010.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Support vector machines under adversarial label noise. In *Asian conference on machine learning*, pages 97–112. PMLR, 2011.

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

Romain Couillet and Florent Benaych-Georges. Kernel spectral clustering of large dimensional data. 2016.

Romain Couillet and Zhenyu Liao. *Random matrix methods for machine learning*. Cambridge University Press, 2022.

Koby Crammer and Daniel Lee. Learning via gaussian herding. *Advances in neural information processing systems*, 23, 2010.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, Yoram Singer, and Manfred K Warmuth. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7(3), 2006.

Koby Crammer, Alex Kulesza, and Mark Dredze. Adaptive regularization of weight vectors. *Advances in neural information processing systems*, 22, 2009.

Yatin Dandi, Ludovic Stephan, Florent Krzakala, Bruno Loureiro, and Lenka Zdeborová. Universality laws for gaussian mixtures in generalized linear models. *Advances in Neural Information Processing Systems*, 36, 2024.

Mark Dredze, Koby Crammer, and Fernando Pereira. Confidence-weighted linear classification. In *Proceedings of the 25th international conference on Machine learning*, pages 264–271, 2008.

Noureddine El Karoui, Derek Bean, Peter J Bickel, Chinghway Lim, and Bin Yu. On robust regression with high-dimensional predictors. *Proceedings of the National Academy of Sciences*, 110(36):14557–14562, 2013.

Yoav Freund. A more robust boosting algorithm. *arXiv preprint arXiv:0905.2138*, 2009.

Thore Graepel and Ralf Herbrich. The kernel gibbs sampler. *Advances in Neural Information Processing Systems*, 13, 2000.

Walid Hachem, Philippe Loubaton, and Jamal Najim. Deterministic equivalents for certain functionals of large random matrices. 2007.

Wenxin Jiang. Some theoretical aspects of boosting in the presence of noisy data. In *Proceedings of the Eighteenth International Conference on Machine Learning*. Citeseer, 2001.

Davood Karimi, Haoran Dou, Simon K Warfield, and Ali Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical image analysis*, 65:101759, 2020.

Neil Lawrence and Bernhard Schölkopf. Estimating a kernel fisher discriminant in the presence of label noise. In *18th international conference on machine learning (ICML 2001)*, pages 306–306. Morgan Kaufmann, 2001.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

Yunlei Li, Lodewyk FA Wessels, Dick de Ridder, and Marcel JT Reinders. Classification in the presence of class noise using a probabilistic kernel fisher method. *Pattern Recognition*, 40(12):3349–3357, 2007.

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

Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, pages 3355–3364. PMLR, 2018.

Xiaoyi Mai and Zhenyu Liao. High dimensional classification via regularized and unregularized empirical

risk minimization: Precise error and optimal loss. *arXiv preprint arXiv:1905.13742*, 2019.

Xiaoyi Mai and Zhenyu Liao. The breakdown of gaussian universality in classification of high-dimensional linear factor mixtures. *arXiv preprint arXiv:2410.05609*, 2024.

Naresh Manwani and PS Sastry. Noise tolerance under risk minimization. *IEEE transactions on cybernetics*, 43(3):1146–1151, 2013.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep Ravikumar, and Ambuj Tewari. Cost-sensitive learning with noisy labels. *Journal of Machine Learning Research*, 18(155):1–33, 2018.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Clayton Scott, Gilles Blanchard, and Gregory Handy. Classification with asymmetric label noise: Consistency and maximal denoising. In *Conference on learning theory*, pages 489–511. PMLR, 2013.

Mohamed El Amine Seddik, Cosme Louart, Mohamed Tamaazousti, and Romain Couillet. Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures. In *International Conference on Machine Learning*, pages 8573–8582. PMLR, 2020.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 2022a.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022b.

Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5552–5560, 2018.

Malik Tiomoko, Hafiz Tiomoko, and Romain Couillet. Deciphering and optimizing multi-task learning: a random matrix approach. In *ICLR 2021-9th International Conference on Learning Representations*, 2021.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]: see sections 3 and 4.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]: see supplementary material.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]: see Assumption 4.1.
   (b) Complete proofs of all theoretical results. [Yes]: see Appendix (section 2).
   (c) Clear explanations of any assumptions. [Yes]: see sections 3 and 4.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]: see supplementary material (code).
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]: see Figure captions and cupplementary material.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

High-Dimensional Learning with Noisy Labels: Supplementary Material

This appendix is organized as follows: Section 1 lists some useful lemmas that will be at the core of our analysis. In Section 2, we provide a more general result of Theorem 4.2 (presented in the main paper) discussed in Remark 3.1 along with the main proof derivations using RMT. Section 3 provides additional plots to support our theoretical results. Section 4 provides derivations for finding the optimal parameter ρ_+^* which defines our optimized classifier. In Section 5 we provide some experiments with synthetic and real data to support the extension of our analysis to arbitrary loss functions instead of the squared loss as supposed in the main paper. Finally, Section 6 presents experiments showing that our analysis can be further extended to multi-class classification which is left for a future investigation.

1 USEFUL LEMMAS

The following lemmas will be useful in deriving all the main results of the paper.

1.1 General lemmas

Lemma 1.1 (Resolvent identity)**.** *For invertible matrices* $\mathbf{A}$ *and* $\mathbf{B}$*, we have:*

$$\mathbf{A}^{-1} - \mathbf{B}^{-1} = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A})\mathbf{B}^{-1}.$$

Lemma 1.2 (Sherman-Morisson)**.** *For* $\mathbf{A} \in \mathbb{R}^{p\times p}$ *invertible and* $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^p$*,* $\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top$ *is invertible if and only if:* $1 + \boldsymbol{v}^\top \mathbf{A}^{-1}\boldsymbol{u} \neq 0$*, and:*

$$(\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\boldsymbol{u}\boldsymbol{v}^\top\mathbf{A}^{-1}}{1 + \boldsymbol{v}^\top\mathbf{A}^{-1}\boldsymbol{u}}.$$

Besides,

$$(\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top)^{-1}\boldsymbol{u} = \frac{\mathbf{A}^{-1}\boldsymbol{u}}{1 + \boldsymbol{v}^\top\mathbf{A}^{-1}\boldsymbol{u}}.$$

1.2 Deterministic equivalents

Recall the expression of the resolvent matrix defined in equation (5):

$$\mathbf{Q}(\gamma) = \left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + \gamma\mathbf{I}_p\right)^{-1}$$

We define the matrix $\mathbf{Q}_{-i}$ as the resolvent obtained by removing the contribution of the i^{th} sample, i.e:

$$\mathbf{Q}_{-i} = \left(\frac{1}{n}\sum_{k\neq i}\boldsymbol{x}_k\boldsymbol{x}_k^\top + \gamma\mathbf{I}_p\right)^{-1},$$

then we have that:

$$\mathbf{Q} = \left(\mathbf{Q}_{-i}^{-1} + \frac{1}{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top\right)^{-1},$$

Thus by Sherman-Morisson's lemma 1.2:

$$\mathbf{Q} = \mathbf{Q}_{-i} - \frac{1}{n}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{1+\delta}, \quad \mathbf{Q}\boldsymbol{x}_i = \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i}{1+\delta}, \quad \delta = \frac{\eta - \gamma - 1 + \sqrt{(\eta-\gamma-1)^2 + 4\eta\gamma}}{2\gamma}.$$

Using the above identities, we can easily prove the deterministic equivalent's lemma 3.5 stated in the main paper:

Lemma 1.3 (Deterministic equivalent of the resolvent)**.** *Under the high-dimensional regime, when $p, n \to \infty$ with $\frac{p}{n} \to \eta \in (0, \infty)$ and assuming $\|\boldsymbol{\mu}\| = \mathcal{O}(1)$. A deterministic equivalent for $\mathbf{Q} \equiv \mathbf{Q}(\gamma)$ is given by:*

$$\bar{\mathbf{Q}} = \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p}{1+\delta} + \gamma \mathbf{I}_p \right)^{-1}, \quad \delta = \frac{1}{n} \operatorname{Tr} \bar{\mathbf{Q}} = \frac{\eta - \gamma - 1 + \sqrt{(\eta - \gamma - 1)^2 + 4\eta\gamma}}{2\gamma}.$$

Proof. We will prove the deterministic equivalent of $\mathbf{Q}$. In general, we want to find a deterministic equivalent $\bar{\mathbf{Q}}$ of the same form of $\mathbf{Q}$, i.e we consider $\bar{\mathbf{Q}}(\gamma) = (\mathbf{S} + \gamma \mathbf{I}_p)^{-1}$ and we want to find a deterministic matrix $\mathbf{S} \in \mathbb{R}^{p \times p}$ such that for any linear form u:

$$u(\mathbf{Q}) \xrightarrow{\text{a.s.}} u(\bar{\mathbf{Q}}),$$

Or more simply:

$$u(\mathbb{E}[\mathbf{Q}] - \bar{\mathbf{Q}}) \to 0.$$

We have that:

$$\begin{aligned}
\mathbb{E}[\mathbf{Q}] - \bar{\mathbf{Q}} &= \mathbb{E}[\mathbf{Q} - \bar{\mathbf{Q}}] \\
&= \mathbb{E}[\mathbf{Q} \left(\mathbf{S} - \frac{1}{n} \mathbf{X}\mathbf{X}^\top \right) \bar{\mathbf{Q}}] \\
&= \mathbb{E}\left[\left(\mathbf{Q}\mathbf{S} - \frac{1}{n} \sum_{i=1}^n \mathbf{Q}\boldsymbol{x}_i \boldsymbol{x}_i^\top \right) \bar{\mathbf{Q}} \right]
\end{aligned}$$

And since: $\mathbf{Q}\boldsymbol{x}_i = \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i}{1+\delta_Q}$ and that we want $\mathbb{E}[\mathbf{Q}] = \bar{\mathbf{Q}}$ in linear forms, we get that:

$$\begin{aligned}
\mathbb{E}\left[\left(\mathbf{Q}\mathbf{S} - \frac{1}{n} \sum_{i=1}^n \mathbf{Q}\boldsymbol{x}_i \boldsymbol{x}_i^\top \right) \bar{\mathbf{Q}} \right] &= \bar{\mathbf{Q}}\mathbf{S}\bar{\mathbf{Q}} - \frac{1}{n} \sum_{i=1}^n \frac{1}{1+\delta} \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i \boldsymbol{x}_i^\top] \bar{\mathbf{Q}} \\
&= \bar{\mathbf{Q}}\mathbf{S}\bar{\mathbf{Q}} - \frac{1}{n} \sum_{i=1}^n \frac{1}{1+\delta} \bar{\mathbf{Q}}(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p)\bar{\mathbf{Q}} \quad \text{(By independence of } \boldsymbol{x}_i \text{ and } \mathbf{Q}_{-i}) \\
&= \bar{\mathbf{Q}} \left(\mathbf{S} - \frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p}{1+\delta} \right) \bar{\mathbf{Q}}
\end{aligned}$$

Finally, it suffices to take: $\mathbf{S} = \frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p}{1+\delta}$ to get the desired result. □

Lemma 1.4 (Relevant Identities)**.** *Let $\bar{\mathbf{Q}} \in \mathbb{R}^{p \times p}$ be the deterministic matrix defined in lemma 3.5. If $\mathbf{C}_a = \mathbf{I}_p$, then we have:*

$$\boldsymbol{\mu}^\top \bar{\mathbf{Q}} \boldsymbol{\mu} = \frac{(1+\delta)\|\boldsymbol{\mu}\|^2}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)}, \quad \boldsymbol{\mu}^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu} = \left(\frac{(1+\delta)\|\boldsymbol{\mu}\|}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)} \right)^2.$$

Proof. We have that:

$$\begin{aligned}
\bar{\mathbf{Q}} &= \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{1+\delta} + \left(\gamma + \frac{1}{1+\delta} \right) \mathbf{I}_p \right)^{-1} \\
&= (1+\delta) \left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + (1 + \gamma(1+\delta)\mathbf{I}_p) \right)^{-1} \\
&= \frac{1+\delta}{1+\gamma(1+\delta)} \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{1+\gamma(1+\delta)} + \mathbf{I}_p \right)^{-1} \\
&= \frac{1+\delta}{1+\gamma(1+\delta)} \left(\mathbf{I}_p - \frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)} \right) \qquad \text{(lemma 1.2)}
\end{aligned}$$

where the last equality is obtained using Sherman-Morisson's identity (lemma 1.2). Hence,

$$(\bar{\mathbf{Q}})^2 = \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2} \left(\mathbf{I}_p + \frac{(\boldsymbol{\mu}\boldsymbol{\mu}^\top)^2}{(\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta))^2} - \frac{2\boldsymbol{\mu}\boldsymbol{\mu}^\top}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)} \right)$$

First identity:

$$\begin{aligned}\boldsymbol{\mu}^\top \bar{\mathbf{Q}} \boldsymbol{\mu} &= \frac{(1+\delta)}{(1+\gamma(1+\delta))}\left(\|\boldsymbol{\mu}\|^2 - \frac{\|\boldsymbol{\mu}\|^4}{\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta)}\right) \\ &= \frac{(1+\delta)\|\boldsymbol{\mu}\|^2}{\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta)}\end{aligned}$$

Second identity:

$$\begin{aligned}\boldsymbol{\mu}^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu} &= \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2}\left(\|\boldsymbol{\mu}\|^2 + \frac{\|\boldsymbol{\mu}\|^6}{(\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta))^2} - \frac{2\|\boldsymbol{\mu}\|^4}{\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta)}\right) \\ &= \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2}\left(\|\boldsymbol{\mu}\| - \frac{\|\boldsymbol{\mu}\|^3}{\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta)}\right)^2 \\ &= \left(\frac{(1+\delta)\|\boldsymbol{\mu}\|}{\|\boldsymbol{\mu}\|^2+1+\gamma(1+\delta)}\right)^2\end{aligned}$$

□

Lemma 1.5 (Deterministic equivalent of $\mathbf{QAQ}$)**.** *For any positive semi-definite matrix* $\mathbf{A}$*, we have:*

$$\mathbf{QAQ} \leftrightarrow \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{\pi_1}{n(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_1\mathbf{Q}] + \frac{\pi_2}{n(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_2\mathbf{Q}],$$

where $\Sigma_a = \boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{C}_a$*. In particular, if* $\mathbf{C} = \mathbf{I}_p$*, i.e* $\Sigma = \boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p$ *then:*

$$\mathbf{QAQ} \leftrightarrow \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n}\frac{\operatorname{Tr}(\Sigma\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})}{(1+\delta)^2}\mathbb{E}[\mathbf{Q}\Sigma\mathbf{Q}].$$

Proof. Let $\bar{\mathbf{Q}}$ be a d.e. of $\mathbf{Q}$. The following equalities are valid in terms of linear forms. We have that:

$$\begin{aligned}\mathbb{E}[\mathbf{QAQ}] &= \mathbb{E}[\bar{\mathbf{Q}}\mathbf{AQ}] + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}] \\ &= \bar{\mathbf{Q}}(\mathbb{E}[\mathbf{AQ}] + \mathbf{A}\mathbb{E}[\mathbf{Q}-\bar{\mathbf{Q}}]) + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}] \\ &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}]\end{aligned}$$

Using lemma 1.1, we have that:

$$\begin{aligned}\mathbf{Q}-\bar{\mathbf{Q}} &= \mathbf{Q}(\bar{\mathbf{Q}}^{-1}-\mathbf{Q}^{-1})\bar{\mathbf{Q}} \\ &= \mathbf{Q}\left(\pi_1\frac{\Sigma_1}{1+\delta_1} + \pi_2\frac{\Sigma_2}{1+\delta_2} - \frac{1}{n}\mathbf{X}\mathbf{X}^\top\right)\bar{\mathbf{Q}} \\ &= \mathbf{Q}(\mathbf{S} - \frac{1}{n}\mathbf{X}\mathbf{X}^\top)\bar{\mathbf{Q}}\end{aligned}$$

Thus:

$$\begin{aligned}\mathbb{E}[\mathbf{QAQ}] &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[\mathbf{Q}(\mathbf{S}-\frac{1}{n}\mathbf{X}\mathbf{X}^\top)\bar{\mathbf{Q}}\mathbf{AQ}] \\ &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[\mathbf{QS}\bar{\mathbf{Q}}\mathbf{AQ}] - \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}]\end{aligned}$$

We have that:

$$\begin{aligned}\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}] &= \frac{1}{1+\delta}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}] \\ &= \frac{1}{1+\delta_i}\left(\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}_{-i}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_i)}]\right) \\ &= \frac{1}{1+\delta_i}\left(\mathbb{E}[\mathbf{Q}_{-i}\Sigma_i\bar{\mathbf{Q}}\mathbf{AQ}_{-i}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_i)}]\right) \\ &= \frac{1}{1+\delta_i}\left(\mathbb{E}[\mathbf{Q}\Sigma_i\bar{\mathbf{Q}}\mathbf{AQ}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_i)}]\right)\end{aligned}$$

Hence, by replacing in the previous identity, we get:

$$\begin{aligned}
\mathbb{E}[\mathbf{QAQ}] &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n}\sum_{i=1}^{n}\frac{1}{(1+\delta_i)^2}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\frac{1}{n}\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n^2}\sum_{i=1}^{n}\frac{1}{(1+\delta_i)^2}\operatorname{Tr}(\Sigma_i\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n^2}\sum_{i=1}^{n}\frac{1}{(1+\delta_i)^2}\operatorname{Tr}(\Sigma_i\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_i\mathbf{Q}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{\pi_1}{n(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_1\mathbf{Q}] + \frac{\pi_2}{n(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_2\mathbf{Q}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \sum_b\frac{\pi_b}{n(1+\delta_b)^2}\operatorname{Tr}(\Sigma_b\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_b\mathbf{Q}]
\end{aligned}$$

Hence, we conclude that:

$$\mathbf{QAQ} \leftrightarrow \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{\pi_1}{n(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_1\mathbf{Q}] + \frac{\pi_2}{n(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_2\mathbf{Q}]$$

□

2 RMT ANALYSIS OF THE LABEL-PERTURBED CLASSIFIER

Notation: For $a \in \{1,2\}$, we denote by $\mathbb{I}_a = \{i \mid \boldsymbol{x}_i \in \mathcal{C}_a\}$, i.e, the set of indices of vectors belonging to class $\mathcal{C}_a$. Furthermore, we denote $\Sigma_a = \mathbb{E}\left[\boldsymbol{x}\boldsymbol{x}^\top\right]$ for $\boldsymbol{x} \in \mathcal{C}_a$.

Assumption 2.1 (Generalized growth rates)**.** *Suppose that as $p, n \to \infty$:*

$$\textit{1)}\ \tfrac{p}{n} \to \eta \in (0,\infty), \qquad \textit{2)}\ \tfrac{n_a}{n} \to \pi_a \in (0,1), \qquad \textit{3)}\ \|\boldsymbol{\mu}\| = \mathcal{O}(1), \qquad \textit{4)}\ \|\Sigma_a\| = \mathcal{O}(1),$$

$\|\Sigma_a\|$ is the spectral norm of the matrix Σ_a.

We consider the LPC with regularization parameter γ given by:

$$\boldsymbol{w}_\rho = \frac{1}{n}\mathbf{Q}(\gamma)\mathbf{X}\mathbf{D}_\rho\tilde{\boldsymbol{y}}, \quad \mathbf{Q}(z) = \left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_p\right)^{-1}, \tag{1}$$

where $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{p\times n}$ and $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \ldots, \tilde{y}_n)^\top \in \mathbb{R}^n$.

Theorem 2.2 (Gaussianity of LPC generalized)**.** *Let $\boldsymbol{w}_\rho$ be the LPC as defined in equation (5) and suppose that Assumption 2.1 holds. The decision function $\boldsymbol{w}_\rho^\top\boldsymbol{x}$, on some test sample $\boldsymbol{x} \in \mathcal{C}_a$ independent from $\mathbf{X}$, satisfies:*

$$\boldsymbol{w}_\rho^\top\boldsymbol{x} \xrightarrow{\mathcal{D}} \mathcal{N}\left((-1)^a m_\rho,\ \nu_\rho - m_\rho^2\right),$$

where:

$$\begin{aligned}
m_\rho &= \left(\pi_1\frac{(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1} + \pi_2\frac{(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}, \\
\nu_\rho &= \left(\frac{\pi_1(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1} + \frac{\pi_2(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)^2\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} \\
&- \frac{T_1}{1+\delta_1}\left(\left(\frac{\pi_1(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1}\right)^2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} + \frac{\pi_1\pi_2(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right) \\
&+ \frac{\pi_1(4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2)}{(1+\delta_1)^2}T_1 + \frac{\pi_2(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)}{(1+\delta_2)^2}T_2 \\
&- \frac{T_2}{1+\delta_2}\left(\left(\frac{\pi_2(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)^2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} + \frac{\pi_1\pi_2(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right),
\end{aligned}$$

where $T_b = \frac{1}{n}\operatorname{Tr}(\Sigma_b\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])$ for $b \in [2]$ and $\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]$ is computed with Lemma 1.5.

Let $g_\rho(\boldsymbol{x}) = \boldsymbol{w}_\rho^\top \boldsymbol{x}$; by Lyupanov's central limit theorem, the decision function follows a Gaussian distribution as $n \to \infty$. Thus, to prove Theorem 2.2, we only need to compute the expectation and the variance of $g_\rho(\boldsymbol{x})$ which are developed below.

2.1 Test Expectation

Denote by $\tilde{\lambda}_i = \frac{1-\rho_{-\tilde{y}_i}+\rho_{\tilde{y}_i}}{1-\rho_+-\rho_-}$, then $\boldsymbol{w}_\rho = \frac{1}{n}\sum_{i=1}^n \mathbf{Q}(\gamma)\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i$.
We have:

$$\begin{aligned}\mathbb{E}\left[g_\rho(\boldsymbol{x})\right] &= \mathbb{E}\left[\boldsymbol{w}_\rho^\top \boldsymbol{x}\right] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\right] = \frac{1}{n}\sum_{i\in\mathbb{I}_1} \mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\right] + \frac{1}{n}\sum_{i\in\mathbb{I}_2} \mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\right] \\ &= \frac{1}{n}\sum_{i\in\mathbb{I}_1} \mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_1\right] + \frac{1}{n}\sum_{i\in\mathbb{I}_2} \mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_2\right]\end{aligned}$$

Recall that:

$$\lambda_+ = \frac{1-\rho_-+\rho_+}{1-\rho_+-\rho_-}, \quad \lambda_- = \frac{1-\rho_++\rho_-}{1-\rho_+-\rho_-}, \quad \beta = \frac{\lambda_-+\lambda_+}{2} \tag{2}$$

Then:

$$\begin{aligned}\mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_1\right] &= \lambda_+\varepsilon_-\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid y_i = -1\right] - \lambda_-(1-\varepsilon_-)\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid y_i = -1\right] \\ &= ((\lambda_+ + \lambda_-)\varepsilon_- - \lambda_-)\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid y_i = -1\right] \\ &= (2\beta\varepsilon_- - \lambda_-)\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid y_i = -1\right] \\ &= \frac{(2\beta\varepsilon_- - \lambda_-)}{1+\delta_1}\boldsymbol{\mu}_1\bar{\mathbf{Q}}\boldsymbol{\mu}_a\end{aligned}$$

Similarly, we have:

$$\begin{aligned}\mathbb{E}\left[\tilde{\lambda}_i\tilde{y}_i\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_2\right] &= \lambda_+(1-\varepsilon_+)\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_2\right] - \lambda_-\varepsilon_+\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_2\right] \\ &= (\lambda_+ - 2\beta\varepsilon_+)\mathbb{E}\left[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{\mu}_a \mid \boldsymbol{x}_i \in \mathcal{C}_2\right] \\ &= \frac{(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\boldsymbol{\mu}_2\bar{\mathbf{Q}}\boldsymbol{\mu}_a\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbb{E}\left[g_\rho(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{C}_a\right] &= \pi_1\frac{(2\beta\varepsilon_- - \lambda_-)}{1+\delta_1}\boldsymbol{\mu}_1^\top\bar{\mathbf{Q}}\boldsymbol{\mu}_a + \pi_2\frac{(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\boldsymbol{\mu}_2^\top\bar{\mathbf{Q}}\boldsymbol{\mu}_a \\ &= (-1)^a\left(\pi_1\frac{(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1} + \pi_2\frac{(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\end{aligned}$$

2.2 Test Variance

To compute the variance of $g_\rho(\boldsymbol{x})$, it only remains to compute the term: $\mathbb{E}[g_\rho(\boldsymbol{x})^2]$.

$$\begin{aligned}\mathbb{E}[g_\rho(\boldsymbol{x})^2] &= \frac{1}{n^2}\sum_{i,j=1}^n \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x}] \\ &= \frac{1}{n^2}\sum_{i\in\mathbb{I}_1}\sum_{j\in\mathbb{I}_1} \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_1] \\ &+ \frac{2}{n^2}\sum_{i\in\mathbb{I}_1}\sum_{j\in\mathbb{I}_2} \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_2] \\ &+ \frac{1}{n^2}\sum_{i\in\mathbb{I}_2}\sum_{j\in\mathbb{I}_2} \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_2, x_j \in \mathcal{C}_2]\end{aligned}$$

Let us develop each sum.

First sum We need to distinguish two cases here: case $i = j$ and $i \neq j$
- **For** $i \neq j$:

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_1] &= \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid y_i = -1, y_j = -1] \\ &= (\lambda_-^2(1-\varepsilon_-)^2 - 2\lambda_-\lambda_+\varepsilon_-(1-\varepsilon_-) + \lambda_+^2\varepsilon_-^2)\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}] \\ &= (\lambda_-(1-\varepsilon_-) - \lambda_+\varepsilon_-)^2\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}] \\ &= (2\beta\varepsilon_- - \lambda_-)^2\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}]\end{aligned}$$

We have that, knowing $x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_1$ and $i \neq j$

$$\begin{aligned}\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}] &= \mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top\mathbf{Q}\boldsymbol{x}_j] \\ &= \mathbb{E}[x_i^\top\mathbf{Q}\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]\mathbf{Q}\boldsymbol{x}_j] \qquad (\boldsymbol{x} \perp\!\!\!\perp (\boldsymbol{x}_i)_{i=1}^n) \\ &= \mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\Sigma_a\mathbf{Q}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)^2}\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\Sigma_a\mathbf{Q}_{-j}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)^2}\mathbb{E}\left[\boldsymbol{x}_i^\top\left(\mathbf{Q}_{-ij} - \frac{\frac{1}{n}\mathbf{Q}_{-ij}\boldsymbol{x}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-ij}}{1+\delta_1}\right)\Sigma_a\left(\mathbf{Q}_{-ij} - \frac{\frac{1}{n}\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}}{1+\delta_1}\right)\boldsymbol{x}_j\right] \\ &= A_1 - A_2 - A_3 + A_4\end{aligned}$$

Let us compute each term now.

$$\begin{aligned}A_1 &= \frac{1}{(1+\delta_1)^2}\mathbb{E}[\boldsymbol{x}_i\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)^2}\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}]\boldsymbol{\mu} \\ &= \frac{1}{(1+\delta_1)^2}\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu}\end{aligned}$$

Hence

$$A_1 = \frac{1}{(1+\delta_1)^2}\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} \tag{3}$$

And we have that:

$$\begin{aligned}A_2 &= \frac{1}{(1+\delta_1)^3}\mathbb{E}[\frac{1}{n}\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)^3}\frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)^3}\frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\end{aligned}$$

Since:

$$\begin{aligned}\frac{1}{n}\boldsymbol{x}_i\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}\boldsymbol{x}_i &= \frac{1}{n}\mathbb{E}[\boldsymbol{x}_i\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}\boldsymbol{x}_i] \\ &= \frac{1}{n}\mathbb{E}[\operatorname{Tr}(\boldsymbol{x}_i\boldsymbol{x}_i\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij})] \\ &= \frac{1}{n}\operatorname{Tr}(\mathbb{E}[\boldsymbol{x}_i\boldsymbol{x}_i\top\mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}]) \\ &= \frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\end{aligned}$$

And we have that:

$$A_2 = A_3$$

And:

$$A_4 = \mathcal{O}(n^{-1})$$

Thus finally:

$$\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_1] = \frac{1}{(1+\delta_1)^2}\left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{2}{(1+\delta_1)}\frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right) \tag{4}$$

Thus:

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_1] &= \frac{(2\beta\varepsilon_- - \lambda_-)^2}{(1+\delta_1)^2} \\ \times \left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{2}{(1+\delta_1)}\frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\end{aligned} \tag{5}$$

- **For** $i = j$: we have that $\tilde{y}_i^2 = 1$ a.s, then knowing $\boldsymbol{x}_i \in \mathcal{C}_1$

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i^2\tilde{y}_i^2(\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x})^2] &= (\lambda_-^2(1-\varepsilon_-) + \lambda_+^2\varepsilon_-)\mathbb{E}[(\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x})^2] \\ &= (4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2)\mathbb{E}[(\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x})^2]\end{aligned}$$

And

$$\begin{aligned}\mathbb{E}[(\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x})^2] &= \mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top \mathbf{Q}\boldsymbol{x}_i] \\ &= \mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\Sigma_a\mathbf{Q}\boldsymbol{x}_i] \\ &= \frac{1}{(1+\delta_1)^2}\mathbb{E}[\operatorname{Tr}(\boldsymbol{x}_i\boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_a\mathbf{Q}_{-i})] \\ &= \frac{1}{(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\end{aligned}$$

Thus:

$$\mathbb{E}[\tilde{\lambda}_i^2\tilde{y}_i^2(\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x})^2 \mid \boldsymbol{x}_i \in \mathcal{C}_1] = \frac{(4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2)}{(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]) \tag{6}$$

Second sum: Here by definition, $i \neq j$. And we have, knowing $x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_2$:

$$\begin{aligned}&\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_2] \\ &= (\lambda_-^2\varepsilon_+(1-\varepsilon_-) - \lambda_+\lambda_-(1-\varepsilon_-)(1-\varepsilon_+) - \lambda_+\lambda_-\varepsilon_+\varepsilon_- + \lambda_+^2\varepsilon_-(1-\varepsilon_+))\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x}] \\ &= (2\beta\varepsilon_+ - \lambda_+)(\lambda_- - 2\beta\varepsilon_-)\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x}]\end{aligned}$$

And, we have that:

$$\begin{aligned}&\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top \mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_2] \\ &= \mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}\Sigma_a\mathbf{Q}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)(1+\delta_2)}\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_a\mathbf{Q}_{-j}\boldsymbol{x}_j] \\ &= \frac{1}{(1+\delta_1)(1+\delta_2)}\mathbb{E}\left[\boldsymbol{x}_i^\top\left(\mathbf{Q}_{-ij} - \frac{\frac{1}{n}\mathbf{Q}_{-ij}\boldsymbol{x}_j\boldsymbol{x}_j^\top \mathbf{Q}_{-ij}}{1+\delta_2}\right)\Sigma_a\left(\mathbf{Q}_{-ij} - \frac{\frac{1}{n}\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}}{1+\delta_1}\right)\boldsymbol{x}_j\right] \\ &= \frac{1}{(1+\delta_1)(1+\delta_2)}(B_1 - B_2 - B_3 + B_4)\end{aligned}$$

We have that:

$$B_1 = \mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\Sigma_a\mathbf{Q}_{-ij}\boldsymbol{x}_j] = -\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu}$$

And

$$\begin{aligned}B_2 &= \frac{1}{n(1+\delta_1)}\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\Sigma_a \mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{1}{n(1+\delta_1)}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{-1}{n(1+\delta_1)}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\end{aligned}$$

And,

$$\begin{aligned}B_3 &= \frac{1}{n(1+\delta_2)}\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\boldsymbol{x}_j\boldsymbol{x}_j^\top \mathbf{Q}_{-ij}\Sigma_a \mathbf{Q}_{-ij}\boldsymbol{x}_j] \\ &= \frac{1}{n(1+\delta_2)}\mathbb{E}[\boldsymbol{x}_i^\top \mathbf{Q}_{-ij}\boldsymbol{x}_j]\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]) \\ &= \frac{-1}{n(1+\delta_2)}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\end{aligned}$$

And $B_4 = \mathcal{O}(n^{-1})$
Thus, finally:

$$\begin{aligned}&\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_1, x_j \in \mathcal{C}_2] \\ &= \frac{(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}(\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{1}{n(1+\delta_1)}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} \\ &- \frac{1}{n(1+\delta_2)}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu})\end{aligned}$$

Third sum: We have that
- **For** $i \neq j$:

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_2, x_j \in \mathcal{C}_2] &= \mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid y_i = 1, y_j = 1] \\ &= (\lambda_-^2\varepsilon_+^2 - 2\lambda_-\lambda_+\varepsilon_+(1-\varepsilon_+) + \lambda_+^2(1-\varepsilon_+)^2)\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}] \\ &= (\lambda_-\varepsilon_+ - \lambda_+(1-\varepsilon_+))^2\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}] \\ &= (2\beta\varepsilon_+ - \lambda_+)^2\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x}]\end{aligned}$$

Thus:

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i\tilde{\lambda}_j\tilde{y}_i\tilde{y}_j\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{x}_j^\top\mathbf{Q}\boldsymbol{x} \mid x_i \in \mathcal{C}_2, x_j \in \mathcal{C}_2] =& \frac{(2\beta\varepsilon_+ - \lambda_+)^2}{(1+\delta_2)^2}\times \\ &\left(\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{2}{(1+\delta_2)}\frac{1}{n}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\end{aligned}$$

- **For** $i = j$:

$$\begin{aligned}\mathbb{E}[\tilde{\lambda}_i^2\tilde{y}_i^2(\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x})^2] &= (\lambda_-^2\varepsilon_+ + \lambda_+^2(1-\varepsilon_+))\mathbb{E}[(\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x})^2] \\ &= (4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)\mathbb{E}[(\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x})^2] \\ &= \frac{(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)}{(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])\end{aligned}$$

Thus:

$$\mathbb{E}[\tilde{\lambda}_i^2\tilde{y}_i^2(\boldsymbol{x}_i^\top\mathbf{Q}\boldsymbol{x})^2 \mid \boldsymbol{x}_i \in \mathcal{C}_2] = \frac{(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)}{(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]) \tag{7}$$

Recall that we denoted by $T_1 = \frac{1}{n}\operatorname{Tr}(\Sigma_1\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])$ and $T_2 = \frac{1}{n}\operatorname{Tr}(\Sigma_2\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}])$, we then deduce that:

Grouping all the terms:

$$
\begin{aligned}
\mathbb{E}[g_\rho(\boldsymbol{x})^2] &= \frac{(\pi_1(\lambda_- - 2\beta\varepsilon_-))^2}{(1+\delta_1)^2}\left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{2}{1+\delta_1}T_1\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\\
&+ \frac{\pi_1(4\beta^2\varepsilon_-(\rho_+-\rho_-)+\lambda_-^2)}{(1+\delta_1)^2}T_1\\
&+ \frac{\pi_1\pi_2(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}\left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{1}{1+\delta_1}T_1\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} - \frac{1}{1+\delta_2}T_2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\\
&+ \frac{(\pi_2(\lambda_+ - 2\beta\varepsilon_+))^2}{(1+\delta_2)^2}\left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu} - \frac{2}{1+\delta_2}T_2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\\
&+ \frac{\pi_2(4\beta^2\varepsilon_+(\rho_--\rho_+)+\lambda_+^2)}{(1+\delta_2)^2}T_2\\
&= \left(\frac{\pi_1(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1} + \frac{\pi_2(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)^2\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]\boldsymbol{\mu}\\
&- \frac{T_1}{1+\delta_1}\left(\left(\frac{\pi_1(\lambda_- - 2\beta\varepsilon_-)}{1+\delta_1}\right)^2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} + \frac{\pi_1\pi_2(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)\\
&+ \frac{\pi_1(4\beta^2\varepsilon_-(\rho_+-\rho_-)+\lambda_-^2)}{(1+\delta_1)^2}T_1 + \frac{\pi_2(4\beta^2\varepsilon_+(\rho_--\rho_+)+\lambda_+^2)}{(1+\delta_2)^2}T_2\\
&- \frac{T_2}{1+\delta_2}\left(\left(\frac{\pi_2(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta_2}\right)^2\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu} + \frac{\pi_1\pi_2(\lambda_+ - 2\beta\varepsilon_+)(\lambda_- - 2\beta\varepsilon_-)}{(1+\delta_1)(1+\delta_2)}\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}\right)
\end{aligned}
$$

Remark 2.3. *The expression $\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}]$ can be easily inferred from this identity (obtained using lemma 1.5):*

$$
\mathbb{E}[\mathbf{Q}\Sigma_a\mathbf{Q}] \leftrightarrow \bar{\mathbf{Q}}\Sigma_a\bar{\mathbf{Q}} + \frac{\pi_1}{n(1+\delta_1)^2}\operatorname{Tr}(\Sigma_1\bar{\mathbf{Q}}\Sigma_a\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_1\mathbf{Q}] + \frac{\pi_2}{n(1+\delta_2)^2}\operatorname{Tr}(\Sigma_2\bar{\mathbf{Q}}\Sigma_a\bar{\mathbf{Q}})\mathbb{E}[\mathbf{Q}\Sigma_2\mathbf{Q}] \tag{8}
$$

So we get a system of two linear independent equations on $\mathbb{E}[\mathbf{Q}\Sigma_1\mathbf{Q}]$ and $\mathbb{E}[\mathbf{Q}\Sigma_2\mathbf{Q}]$, and therefore they are uniquely determined.

2.3 Isotropic Case

If $\mathbf{C} = \mathbf{I}_p$, then we have that:

$$
\delta_1 = \delta_2 = \delta, \qquad T_1 = T_2 = \frac{1}{n}\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2) = \frac{\eta(1+\delta)^2}{(1+\gamma(1+\delta))^2} \tag{9}
$$

and using lemma 1.5:

$$
\mathbb{E}[\mathbf{Q}\Sigma\mathbf{Q}] \leftrightarrow \frac{1}{1-\frac{1}{n}\frac{\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2)}{(1+\delta)^2}}\bar{\mathbf{Q}}\Sigma\bar{\mathbf{Q}} = \frac{1}{h}\bar{\mathbf{Q}}\Sigma\bar{\mathbf{Q}} \tag{10}
$$

where :

$$
h = 1 - \frac{1}{n}\frac{\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2)}{(1+\delta)^2} = 1 - \frac{\eta}{(1+\gamma(1+\delta))^2} \tag{11}
$$

Hence, we get the following result:

Corollary 2.4 (Gaussiannity of the label-perturbed classifier)**.** *Let $\boldsymbol{w}_\rho$ be the LPC with parameters $\rho_\pm$, and $\bar{\mathbf{Q}}$ a deterministic equivalent of $\mathbf{Q}$ defined in lemma 3.5. Under the same assumptions of 4.1:*

$$
\boldsymbol{w}_\rho^\top\boldsymbol{x} \xrightarrow{\mathcal{D}} \mathcal{N}\left((-1)^a m_\rho,\, \nu_\rho - m_\rho^2\right),
$$

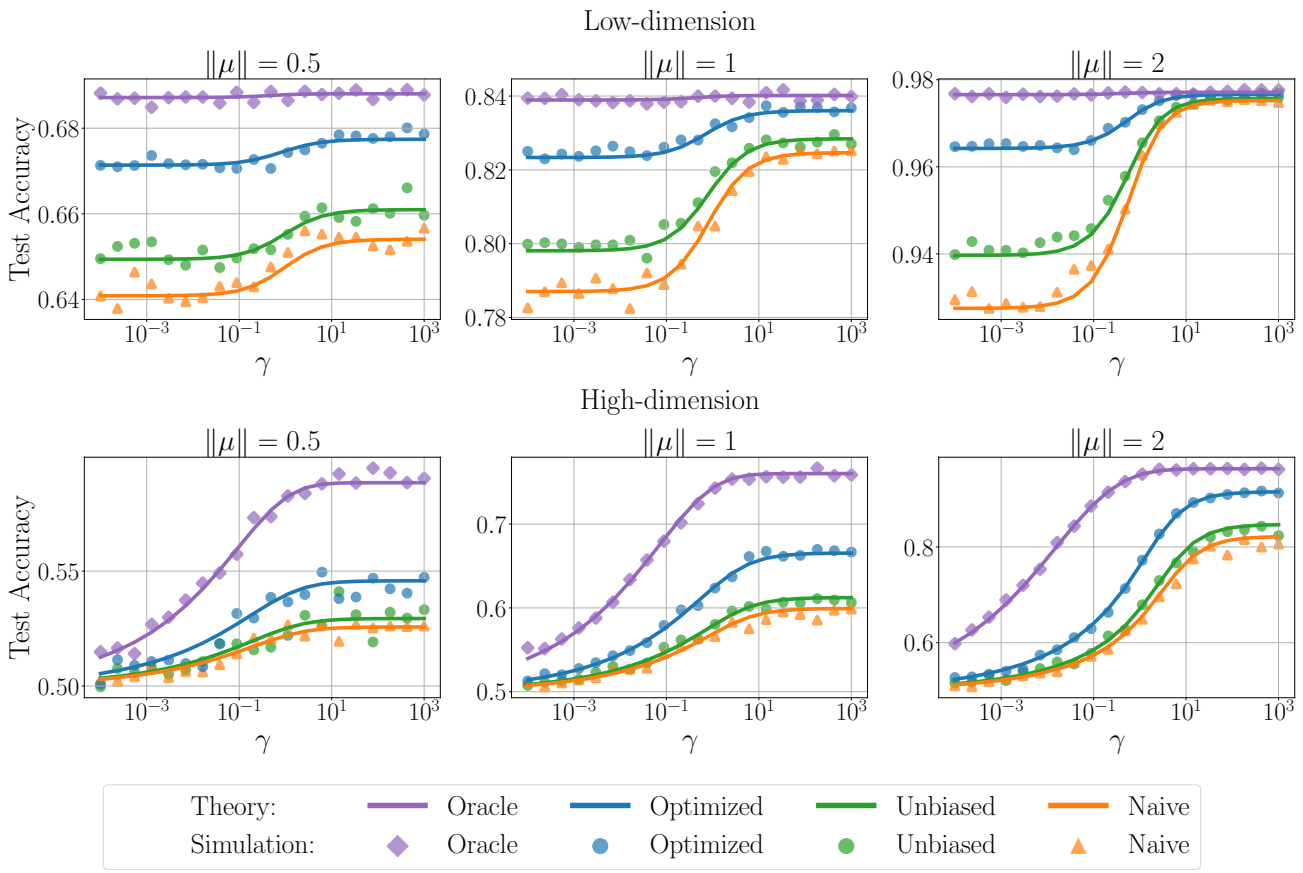

Figure 1: Empirical versus theoretical test accuracy for different variants of LPC. We used $(n, p = 2000, 20)$ for Low-dimensional plot $(n, p = 200, 200)$ and for High-dimensional experiment, $\pi_1 = 0.3$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$ and varied γ.

where:

$$
\begin{aligned}
m_\rho &= \frac{\pi_1(\lambda_- - 2\beta\varepsilon_-) + \pi_2(\lambda_+ - 2\beta\varepsilon_+)}{1+\delta}\boldsymbol{\mu}^\top\bar{\mathbf{Q}}\boldsymbol{\mu}, \\
\nu_\rho &= \frac{(\pi_1(2\beta\varepsilon_- - \lambda_-) + \pi_2(2\beta\varepsilon_+ - \lambda_+))^2}{h(1+\delta)^2}\left(\mu^\top\bar{\mathbf{Q}}\Sigma\bar{\mathbf{Q}}\mu - \frac{2}{(1+\delta)}\frac{1}{n}\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2)\mu^\top\bar{\mathbf{Q}}\mu\right) \\
&+ \frac{1}{hn(1+\delta)^2}\pi_1(4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2)\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2) \\
&+ \frac{1}{hn(1+\delta)^2}\pi_2(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)\operatorname{Tr}((\Sigma\bar{\mathbf{Q}})^2).
\end{aligned}
$$

We finally get our main Theorem 4.2 of the paper by further simplifying the previous expressions using lemma 1.4 and 9:

$$
\begin{aligned}
m_\rho &= \frac{\pi_1(\lambda_- - 2\beta\varepsilon_-) + \pi_2(\lambda_+ - 2\beta\varepsilon_+)}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)}\|\boldsymbol{\mu}\|^2, \\
\nu_\rho &= \frac{(\pi_1(2\beta\varepsilon_- - \lambda_-) + \pi_2(2\beta\varepsilon_+ - \lambda_+))^2}{h(\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta))}\left(\frac{\|\boldsymbol{\mu}\|^2 + 1}{\|\boldsymbol{\mu}\|^2 + 1 + \gamma(1+\delta)} - 2(1-h)\right)\|\boldsymbol{\mu}\|^2 \\
&+ \frac{(1-h)}{h}\left(\pi_1(4\beta^2\varepsilon_-(\rho_+ - \rho_-) + \lambda_-^2) + \pi_2(4\beta^2\varepsilon_+(\rho_- - \rho_+) + \lambda_+^2)\right).
\end{aligned}
$$

Figure 1 shows a consistent estimation of the test accuracy of different LPC variants.

3 ADDITIONAL FIGURES

Figure 2 shows the result of estimating ε_+ using our approach as described in Section 4.2. We particularly notice that the estimated value of ε_+ is consistent even for small SNR $\|\boldsymbol{\mu}\|$.

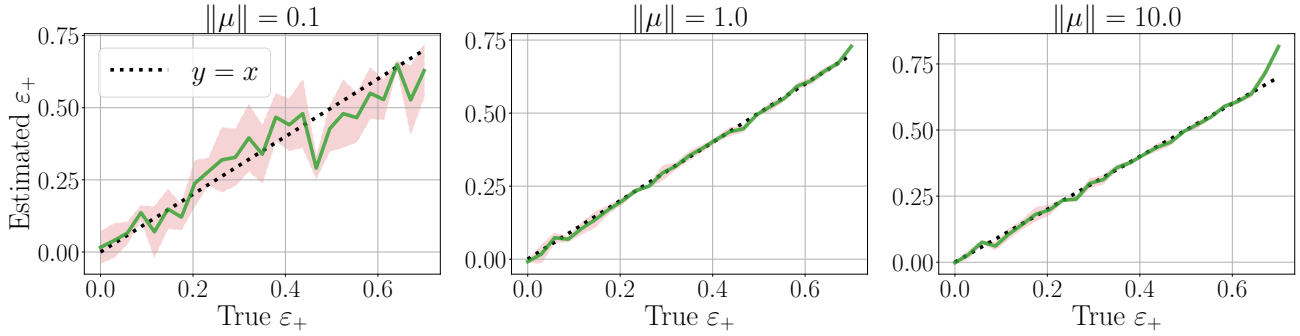

Figure 2: Estimation of the label noise rates as described in Section 4.2. We used $n = 1000$, $p = 100$, $\pi_1 = \frac{1}{3}$, $\varepsilon_- = 0.2$, $(\rho_+^{(1)}, \rho_-^{(1)}) = (0, 0.1)$ and $(\rho_+^{(2)}, \rho_-^{(2)}) = (0, 0.4)$.

4 FINDING OPTIMAL PARAMETERS

We denote by $\pi = \pi_1$ the proportion of data belonging to $\mathcal{C}_1$ (hence $\pi_2 = 1 - \pi$). Our goal is to maximize the theoretical test accuracy with respect to ρ_+ for any fixed ρ_-. This is equivalent to maximizing the term $\frac{m_\rho^2}{\nu_\rho - m_\rho^2}$ since $\varphi(\cdot)$ is a decreasing function. We have that:

$$r(\rho_+) = \frac{m_\rho^2}{\nu_\rho - m_\rho^2} = \frac{N_1(\rho_+)}{D_1(\rho_+)}$$

where:

$$N_1(\rho_+) = -hm_{\text{oracle}}^2 \left(\pi \left(2\epsilon_- + \rho_+ - \rho_- - 1\right) - (\pi - 1)\left(2\epsilon_+ - \rho_+ + \rho_- - 1\right)\right)^2 \left(\rho_+ + \rho_- - 1\right)^2$$

and

$$\begin{aligned} D_1(\rho_+) = &-h\left(\kappa - m_{\text{oracle}}^2\right)\left(\pi \left(2\epsilon_- + \rho_+ - \rho_- - 1\right) - (\pi - 1)\left(2\epsilon_+ - \rho_+ + \rho_- - 1\right)\right)^2 \left(\rho_+ + \rho_- - 1\right)^2 \\ &+ (h - 1)\left(\pi\left(4\epsilon_-\left(\rho_+ - \rho_-\right) + \left(-\rho_+ + \rho_- + 1\right)^2\right) + (\pi - 1)\left(4\epsilon_+\left(\rho_+ - \rho_-\right) - \left(\rho_+ - \rho_- + 1\right)^2\right)\right)\left(\rho_+ + \rho_- - 1\right)^2 \end{aligned}$$

And differentiating r with respect to ρ_+ gives us:

$$r'(\rho_+) = \frac{N_2(\rho_+)}{D_2(\rho_+)}$$

where :

$$\begin{aligned} N_2(\rho_+) = &\, 2hm_{\text{oracle}}^2(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1)) \\ &\times (-(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1)) \\ &\times (h(\kappa - m_{\text{oracle}}^2)(2\pi - 1)(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1))(\rho_+ + \rho_- - 1) \\ &+ h(\kappa - m_{\text{oracle}}^2)(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1))^2 \\ &- (h - 1)(\pi(4\epsilon_-(\rho_+ - \rho_-) + (-\rho_+ + \rho_- + 1)^2) + (\pi - 1)(4\epsilon_+(\rho_+ - \rho_-) - (\rho_+ - \rho_- + 1)^2)) \\ &- (h - 1)(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) + (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1))(\rho_+ + \rho_- - 1)) \\ &+ (h(\kappa - m_{\text{oracle}}^2)(\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1))^2 \\ &- (h - 1)(\pi(4\epsilon_-(\rho_+ - \rho_-) + (-\rho_+ + \rho_- + 1)^2) + (\pi - 1)(4\epsilon_+(\rho_+ - \rho_-) - (\rho_+ - \rho_- + 1)^2))) \\ &\times (\pi(2\epsilon_- + \rho_+ - \rho_- - 1) - (\pi - 1)(2\epsilon_+ - \rho_+ + \rho_- - 1) + (2\pi - 1)(\rho_+ + \rho_- - 1))) \end{aligned}$$

And finally, solving $N_2(\rho_+) = 0$ gives us two solutions:

$$\rho_+^* = \frac{\pi^2\epsilon_-(\epsilon_- - 1) + (1-\pi)^2\epsilon_+(1-\epsilon_+)}{\pi(1-\pi)(1-\epsilon_+-\epsilon_-)} + \rho_-, \qquad \bar{\rho}_+ = \frac{1 - 2\pi\varepsilon_- - 2(1-\pi)\varepsilon_+}{2\pi - 1} + \rho_-.$$

5 LOSS GENERALIZATION

To investigate the extension of our approach to other bounded losses in addition to the squared loss considered in the main paper, we evaluated our LPC trained with the label perturbed loss (equation (3)) using a binary-cross-entropy loss, that is:

$$\ell(s(\boldsymbol{x}), y) = -y \log\left(s(\boldsymbol{x})\right) - (1-y)\log\left(1 - s(\boldsymbol{x})\right), \tag{12}$$

where $s(\boldsymbol{x}) = \frac{1}{1+\exp(-\boldsymbol{w}^\top \boldsymbol{x})}$ and y is in $\{0, 1\}$. Figures 3 and 4 summarize the obtained test accuracies by setting ρ_- to zero and varying ρ_+ on both synthetic and real data respectively. As anticipated theoretically with the squared loss, we remark similar behavior about the existence of an optimal ρ_+^* that maximizes the accuracy beyond the *unbiased* approach.

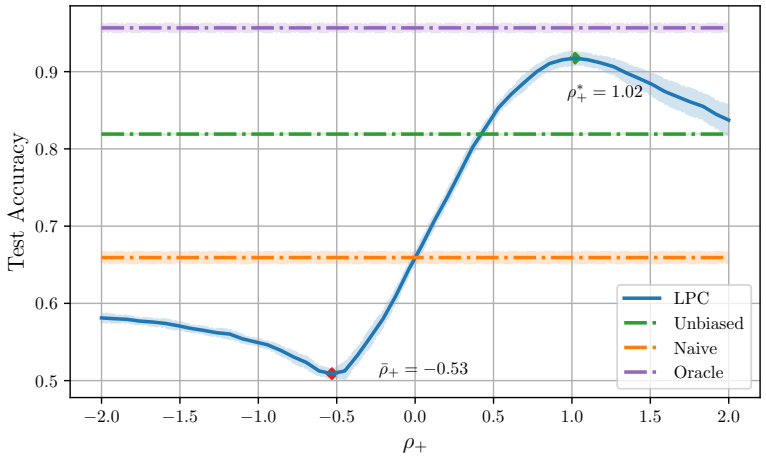

Figure 3: Test Accuracy on Synthetic data with classifiers obtained through minimizing the binary-cross-entropy loss using gradient descent. We used the parameters $n = 1000$, $p = 1000$, $\pi_1 = 0.3$, $\|\boldsymbol{\mu}\| = 2$, $\varepsilon_+ = 0.4$, $\varepsilon_- = 0.3$ and a learning rate of 0.1.

6 MULTI-CLASS EXTENSION: MULTI-LPC

In this section, we provide some evidence to show that our setting can be further extended to multi-class classification by considering the following settings.

6.1 Setting

We consider having a set of n i.i.d p-dimensional vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n \in \mathbb{R}^p$ and corresponding labels $y_1, y_2, ..., y_n \in \{1, ..., k\}$ such that the $\boldsymbol{x}_i$'s are sampled from a Gaussian mixture of k clusters $\mathcal{C}_1, ..., \mathcal{C}_k$ with, $a \in \{1, ..., k\}$:

$$\boldsymbol{x}_i \in \mathcal{C}_a \quad \Leftrightarrow \quad \boldsymbol{x}_i = \boldsymbol{\mu}_a + \boldsymbol{z}_i,$$

where $\boldsymbol{\mu}_a \in \mathbb{R}^p$ and $\boldsymbol{z}_i \in \mathcal{N}(0, \mathbf{I}_p)$. We consider that the true labels are flipped randomly to get $\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_n$ such that for $a, b \in \{1, ..., k\}$:

$$\mathbb{P}(\tilde{y}_i = a \mid y_i = b) = \varepsilon_{a,b}, \quad \sum_{b=1}^{k} \varepsilon_{a,b} < 1. \tag{13}$$

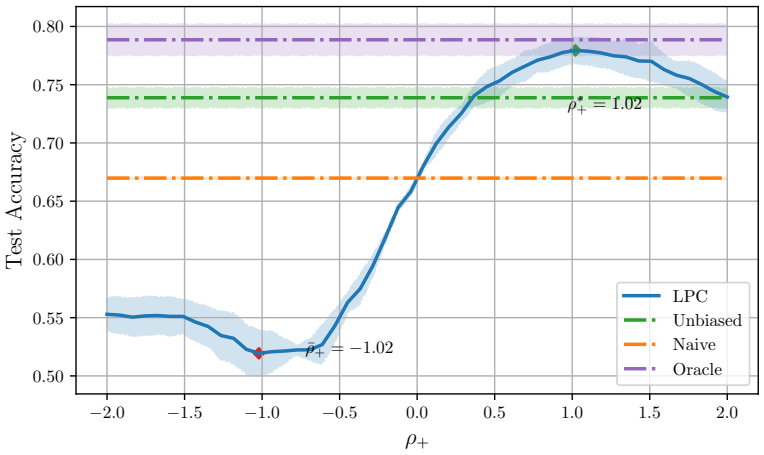

Figure 4: Test Accuracy on `Dvd` Amazon dataset with classifiers obtained through minimizing the binary-cross-entropy loss using gradient descent. We used the parameters $n = 1600$, $p = 400$, $\pi_1 = 0.3$, $\|\boldsymbol{\mu}\| = 2$, $\varepsilon_+ = 0.3$, $\varepsilon_- = 0.2$ and a learning rate of 0.1.

6.2 Linear model

Let $\boldsymbol{y}_i \in \mathbb{R}^k$ denote the one-hot encoding of the label y_i, i.e., if $\boldsymbol{x}_i \in \mathcal{C}_a$:

$$\boldsymbol{y}_{i,j} = \begin{cases} 1 & \text{if} \quad j = a, \\ 0 & \text{otherwise.} \end{cases}$$

Denote the data matrix $\mathbf{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$ and labels matrix $\mathbf{Y} = [\boldsymbol{y}_1, ..., \boldsymbol{y}_n] \in \mathbb{R}^{k \times n}$.

Naive approach: We consider a linear model that consists of minimizing the following regularized squared loss:

$$\mathcal{L}(\mathbf{W}) = \frac{1}{n} \sum_{i=1}^{n} \|\tilde{\boldsymbol{y}}_i - \mathbf{W}^\top \boldsymbol{x}_i\| + \gamma \|\mathbf{W}\|_F^2, \tag{14}$$

where $\gamma \geq 0$ is a regularization parameter, and $\|.\|_F$ denotes the Frobenius norm of a matrix. The minimizer of this equation reads explicitly as:

$$\mathbf{W} = \frac{1}{n} \mathbf{Q}(\gamma) \mathbf{X} \tilde{\mathbf{Y}}^\top, \quad \mathbf{Q}(\gamma) = \left(\frac{1}{n} \mathbf{X}\mathbf{X}^\top + \gamma \mathbf{I}_p \right)^{-1}. \tag{15}$$

Multi-LPC : Let us sort the data vectors $(\boldsymbol{x}_i)_{i=1}^n$ in $\mathbf{X}$ and their labels $(\tilde{\boldsymbol{y}}_i)_{i=1}^n$ in their matrices $\mathbf{X}$ and $\tilde{\mathbf{Y}}$ such that we put the vectors of class $\mathcal{C}_1$ in the first columns, then those of class $\mathcal{C}_2$, and so on.
Let $\tilde{\mathbf{Y}}^\top = [\boldsymbol{u}_1, ..., \boldsymbol{u}_k]$, each vector $\boldsymbol{u}_i$ is defined in the following way:

$$\boldsymbol{u}_{i,j} = \begin{cases} 1 & \text{if} \quad \sum_{a=1}^{i-1} \tilde{n}_a \leq j < \sum_{a=1}^{i} \tilde{n}_a \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where $\tilde{n}_a$ is the number of noisy samples belonging to class $\mathcal{C}_a$, i.e., the cardinality of this set $\{i \in \{1, ..., n\} \mid \tilde{y}_i = a\}$. Now let $\alpha_1, ..., \alpha_k, \beta_1, ..., \beta_k \in \mathbb{R}$. Our Multi-LPC approach consists of considering the following label matrix:

$$\mathbf{Y}_{\alpha,\beta}^\top = [\alpha_1 \boldsymbol{u}_1 + \beta_1(\mathbf{1}_n - \boldsymbol{u}_1), ..., \alpha_k \boldsymbol{u}_k + \beta_k(\mathbf{1}_n - \boldsymbol{u}_k)] \tag{17}$$

$$= \tilde{\mathbf{Y}}^\top \mathbf{D}(\alpha) + (\mathbf{M}_1 - \tilde{\mathbf{Y}}^\top) \mathbf{D}(\beta) \tag{18}$$

where $\mathbf{M}_1 \in \mathbb{R}^{n\times k}$ is the matrix containing 1 in all its entries, and $\mathbf{D}(\alpha) \in \mathbb{R}^{k\times k}$ (resp. $,\mathbf{D}(\beta) \in \mathbb{R}^{k\times k}$) is a diagonal matrix containing the coefficients $\alpha_1, ..., \alpha_k$ (resp. $\beta_1, ..., \beta_k$) in its diagonal. Thus the multi-class LPC classifier is defined as:

$$\mathbf{W} = \frac{1}{n}\mathbf{Q}(\gamma)\mathbf{X}\tilde{\mathbf{Y}}_{\alpha,\beta}^{\top}. \tag{19}$$

Our aim is to show the existence of parameters $(\alpha_i^*)_{i=1}^k$ and $(\beta_i^*)_{i=1}^k$ that maximize the accuracy of the classifier.

Remark 6.1. *Remark that we can recover the Naive classifier in (14) by taking $\alpha_i = 1$ and $\beta_i = 0$ for all $i \in \{1, ..., k\}$.*

6.3 Experiments

We tested our extension for $k = 3$ and $k = 4$ classes using synthetic data by taking:

For 3 classes ($k = 3$): We considered the following noise parameters matrix $\boldsymbol{\varepsilon}$ and the proportions $\boldsymbol{\pi}$ of data in each class ($\boldsymbol{\pi}_i$ is the proportion of data belonging to class $\mathcal{C}_i$):

$$\boldsymbol{\varepsilon} = \begin{pmatrix} 0 & 0.3 & 0 \\ 0 & 0 & 0.4 \\ 0.5 & 0 & 0 \end{pmatrix} \qquad \boldsymbol{\pi} = (0.3, 0.3, 0.4)$$

We also considered class $\mathcal{C}_3$ of mean vector $\boldsymbol{\mu}_3$ of norm $\|\boldsymbol{\mu}_3\| = 2$, class $\mathcal{C}_1$ of mean $\boldsymbol{\mu}_1 = -\boldsymbol{\mu}_3$ and a centered class $\mathcal{C}_2$ (zero norm mean).

For 4 classes ($k = 4$): We considered the parameters:

$$\boldsymbol{\varepsilon} = \begin{pmatrix} 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.3 \\ 0 & 0.4 & 0 & 0 \\ 0.3 & 0 & 0 & 0 \end{pmatrix} \qquad \boldsymbol{\pi} = (0.3, 0.2, 0.3, 0.2)$$

We also considered classes $\mathcal{C}_3$ and $\mathcal{C}_4$ of mean vectors $\boldsymbol{\mu}_3$ and $\boldsymbol{\mu}_4$ respectively such that: $\|\boldsymbol{\mu}_3\| = 2$ and $\|\boldsymbol{\mu}_3\| = 6$, and considered $\mathcal{C}_1$ of mean $\boldsymbol{\mu}_1 = -\boldsymbol{\mu}_4$ and $\mathcal{C}_2$ of mean $\boldsymbol{\mu}_2 = -\boldsymbol{\mu}_3$.

For each number of classes k, we found the optimal parameters (in terms of accuracy) $\boldsymbol{\alpha}^* = (\alpha_i^*)_{i=1}^k$ and $\boldsymbol{\beta}^* = (\beta_i^*)_{i=1}^k$ and also the worst ones $\bar{\boldsymbol{\alpha}} = (\bar{\alpha}_i)_{i=1}^k$ and $\bar{\boldsymbol{\beta}} = (\bar{\beta})_{i=1}^k$ within a grid of $G = 5000$ parameters, using Monte Carlo simulation. To visualize the results, we report the accuracy of the Multi-LPC approach with the parameters $\boldsymbol{\alpha}_\tau = \tau\boldsymbol{\alpha}^* + (1-\tau)\bar{\boldsymbol{\alpha}}$ and $\boldsymbol{\beta}_\tau = \tau\boldsymbol{\beta}^* + (1-\tau)\bar{\boldsymbol{\beta}}$ by varying the parameter $\tau \in (0, 1)$. Figure 5 summarizes the obtained results and we clearly observe improved accuracy for $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ even approaching the oracle classifier.

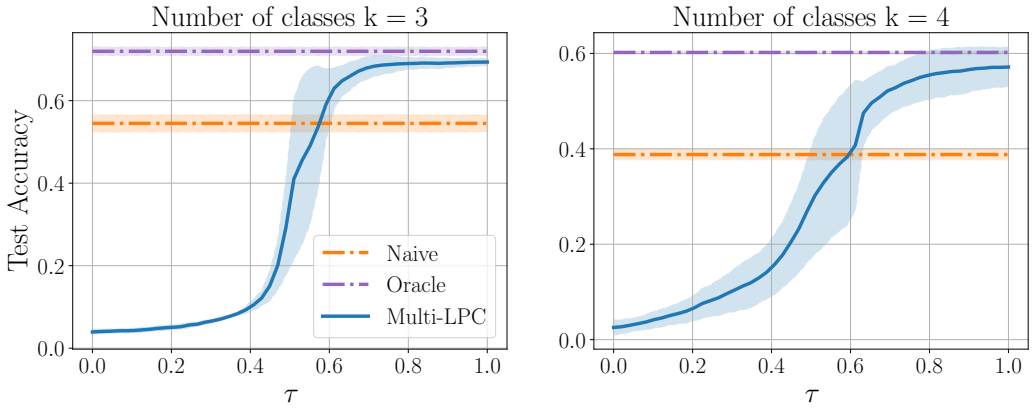

Figure 5: Multi-class classification with $n = 2000$, $p = 200$ evaluated on 3 random seeds.