Journal of Machine Learning Research 25 (2024) 1-53 Submitted 6/23; Revised 4/24; Published 4/24

Linear Distance Metric Learning with Noisy Labels

Meysam Alishahi ALISHAHI@CS.UTAH.EDU
Kahlert School of Computing
University of Utah
Salt Lake City, UT 84112, USA

Anna Little LITTLE@MATH.UTAH.EDU
Department of Mathematics, Utah Center For Data Science
University of Utah
Salt Lake City, UT 84112, USA

Jeff M. Phillips JEFFP@CS.UTAH.EDU
Kahlert School of Computing, Utah Center for Data Science
University of Utah
Salt Lake City, UT 84112, USA
and visiting ScaDS.AI, University of Leipzig, and MPI for Math in the Sciences

Editor: Aryeh Kontorovich

Abstract

In linear distance metric learning, we are given data in one Euclidean metric space and the goal is to find an appropriate linear map to another Euclidean metric space which respects certain distance conditions as much as possible. In this paper, we formalize a simple and elegant method which reduces to a general continuous convex loss optimization problem, and for different noise models we derive the corresponding loss functions. We show that even if the data is noisy, the ground truth linear metric can be learned with any precision provided access to enough samples, and we provide a corresponding sample complexity bound. Moreover, we present an effective way to truncate the learned model to a low-rank model that can provably maintain the accuracy in the loss function and in parameters – the first such results of this type. Several experimental observations on synthetic and real data sets support and inform our theoretical results.

Keywords: linear metric learning, Mahalanobis distance, positive semi-definite matrix, low-rank metric learning

1. Introduction

The goal of distance metric learning is to map data in a metric space into another metric space in such a way that the distance between points in the second space optimizes some condition on the data. Early work in this area focuses mostly on the Euclidean to Euclidean setting, and specifically on the case of learning linear transformations. For data $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, it attempts to learn a Mahalanobis distance $\mathrm{d}_{\boldsymbol{M}} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{\geq 0}$ as $\mathrm{d}_{\boldsymbol{M}}(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{M}} = \sqrt{(\boldsymbol{x} - \boldsymbol{y})^t \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{y})}$. This is a metric on the original space $\mathbb{R}^d$ as long as $\boldsymbol{M}$ is positive definite. We can decompose $\boldsymbol{M}$ as $\boldsymbol{M} = \boldsymbol{A}\boldsymbol{A}^t$ for $\boldsymbol{A} \in \mathbb{R}^{d \times d}$. Then $\boldsymbol{A}$ can be used as a linear map so that $\boldsymbol{X}' = \boldsymbol{A}^t \boldsymbol{X}$ is another point set of n points in d dimensions; for $\boldsymbol{x}' = \boldsymbol{A}^t \boldsymbol{x}$ and $\boldsymbol{y}' = \boldsymbol{A}^t \boldsymbol{y}$ in the new space, $\|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{M}}$ is equivalent to the standard Euclidean distance $\|\boldsymbol{x}' - \boldsymbol{y}'\|$.

Linear metric learning has been studied in Weinberger and Saul (2009) for kNN classification, in Torresani and Lee (2006); Wang and Zhang (2007); Xiang et al. (2008) via margin/distance optimization, in Sugiyama (2007); Bar-Hillel et al. (2005) via discriminant analysis, and in Nguyen et al. (2017) via Jeffrey divergence. Many of these linear methods also propose kernelized versions, and kernelized metric learning was also considered in Mika et al. (1999); Roth and Steinhage (1999). But the current state of the art uses arbitrarily complex neural encoders that attempt to optimize the final objective with very little restriction on the form or structure of the mapping (Ermolov et al., 2022). Merging with the area of feature engineering (Nargesian et al., 2017; Shi et al., 2009), these approaches are an integral element of information retrieval (Patel et al., 2021; Ramzi et al., 2022), natural language processing (Mikawa et al., 2011; Li et al., 2017), and image processing (Chopra et al., 2005). To access further details, readers can refer to two well-conducted surveys Bellet et al. (2013); Kulis (2013).

In this work, we revisit *linear* distance metric learning. We posit that there are two useful extremes in this problem, the anything-goes non-linear approaches mentioned above, and the very restrictive linear approaches. The linear approaches exhibit a number of important properties which are essential for certain applications:

- When the original coordinates of the data points have meaning, but for instance are measured in different units (e.g., inches and pounds), then one may want to retain that meaning and interpretability while making the process invariant to the original underlying (and often arbitrary) choice of units.

- Many geometric properties such as linear separability, convexity, straight-line connectivity, vector translation (linear parallel transport) are preserved under affine transformations. If such features are assumed to be meaningful on the original data, then they are retained under a linear transformation.

- Some physical equations, such as those describing ordinary differential equations (ODEs) can be simulated through a linear transform (Sutherland and Parente, 2009). We will demonstrate an example application of this (in Section 5.5) where because of changing units, it is not clear how to measure distance in the original space, and locality based learning can be more effectively employed after a linear transformation.

While several prior works have already explored linear distance metric learning (Weinberger and Saul, 2009; Torresani and Lee, 2006; Sugiyama, 2007; Nguyen et al., 2017), they often reduce to novel optimization settings where specially designed solvers and analysis are required. For instance, Ying and Li (2012) utilize a clever subgradient descent formulation to ensure the learned $\boldsymbol{M}$ retains its positive-definiteness. In our work we provide a simple and natural formulation that converts the linear distance metric learning task into a simple supervised convex gradient descent procedure, basically a standard supervised classification task where any procedure for smooth convex optimization can be employed.

1.1 Formulation

Specifically, we assume N i.i.d. observations $(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathbb{R}^d \times \mathbb{R}^d$ and each pair is given a label $\ell_i \in \{\text{Far}, \text{Close}\}$. Our goal is to learn a positive semi-definite (p.s.d.) matrix $\boldsymbol{M}$ and threshold $\tau \geq 0$ so that $\|\boldsymbol{x}_i - \boldsymbol{y}_i\|^2_{\boldsymbol{M}} \geq \tau$ if $\ell_i = \text{Far}$ and $\|\boldsymbol{x}_i - \boldsymbol{y}_i\|^2_{\boldsymbol{M}} < \tau$ if $\ell_i = \text{Close}$. Towards solving

this we formulate an optimization problem

$$\min_{\substack{\tau \geq 0 \\ \boldsymbol{M} \succeq 0}} R_N(\boldsymbol{M}, \tau) = \min_{\substack{\tau \geq 0 \\ \boldsymbol{M} \succeq 0}} \frac{1}{N} \sum_{i=1}^{N} L(\boldsymbol{x}_i, \boldsymbol{y}_i, \ell_i; \boldsymbol{M}, \tau) \tag{1}$$

where $L(\boldsymbol{x}_i, \boldsymbol{y}_i, \ell_i; \boldsymbol{M}, \tau)$ is a loss function that penalizes the mismatch between the observed label and the model-predicted label. Then we propose to optimize this in an (almost, except for $\tau \geq 0$) unconstrained setting, where we can apply standard techniques like (stochastic) gradient descent:

$$\min_{\substack{\tau \in \mathbb{R} \\ \boldsymbol{A} \in \mathbb{R}^{d \times d}}} R_N(\boldsymbol{A}\boldsymbol{A}^t, \tau)$$

1.2 Our core results

We analyze this simple, flexible, and powerful formulation and show that:

- This optimization problem is convex over $\boldsymbol{M}, \tau$. Moreover, while optimizing over $\boldsymbol{A}$ is not convex, we can leverage an observation of Journée et al. (2010) to show that the minimizer $\boldsymbol{A}^*$ over the unconstrained formulation generates $\boldsymbol{M}^* = \boldsymbol{A}^*(\boldsymbol{A}^*)^t$, which is the minimizer over the convex, but (positive semi-definite) constrained formulation over $\boldsymbol{M}$.

- The sample complexity of this problem is $N_d(\varepsilon, \delta) = O(\frac{1}{\varepsilon^2}(\log \frac{1}{\delta} + d^2 \log \frac{d}{\varepsilon}))$. More specifically, let f be the pdf of the distribution from which difference pairs $\boldsymbol{x} - \boldsymbol{y}$ are drawn, let ℓ be the (noisy) label associated with a pair, and let $R(\boldsymbol{M}, \tau) = \mathbf{E}_{\boldsymbol{x}-\boldsymbol{y} \sim f}[L(\boldsymbol{x}, \boldsymbol{y}, \ell; \boldsymbol{M}, \tau)]$ be the expected loss. Then, given $N_d(\varepsilon, \delta)$ observations, $|R(\hat{\boldsymbol{M}}, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*)| \leq \varepsilon$ with probability at least $1 - \delta$, where $(\hat{\boldsymbol{M}}, \hat{\tau})$ is the minimizer of R_N.

- If the labels ℓ_i are observed with unbiased noise, and the loss function is chosen appropriately to match that noise distribution, then R_N still approximates R, and in fact, the minimizers $\hat{\boldsymbol{M}}, \hat{\tau}$ of R_N converge to the true minimizers $\boldsymbol{M}^*, \tau^*$ of R.

- Returning a low-rank approximation $\hat{\boldsymbol{M}}_k$ to $\hat{\boldsymbol{M}}$ can achieve bounded error with respect to $|R(\hat{\boldsymbol{M}}_k, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*)|$ and $\|\hat{\boldsymbol{M}} - \boldsymbol{M}^*\| + |\hat{\tau} - \tau^*|$, as elaborated on just below. To the best of our knowledge, this is the first such dimensionality reduction result of this kind.

1.3 Reasonable Choice for the Loss Function L: Logistic noise

We assume the labeling of {Close, Far} through the evaluation of $\|\boldsymbol{x}_i - \boldsymbol{y}_i\|^2_{\boldsymbol{M}}$ is noisy. We will prove that if the noise comes from the Logistic distribution, then

$$L(\boldsymbol{x}_i, \boldsymbol{y}_i, \ell_i; \boldsymbol{M}, \tau) = -\log \sigma(\ell_i(\|\boldsymbol{x}_i - \boldsymbol{y}_i\|^2_{\boldsymbol{M}} - \tau))$$

serves as an excellent theoretical choice; here $\sigma(x) = \frac{1}{1+e^{-x}}$ is known as the *Logistic* function. We note that prior work (Guillaumin et al., 2009) also considered this special case of our formulation, and provided an empirical study on face identification; however they did not theoretically analyze this formulation.

As mentioned, our work will show that this form of L is indeed optimal under a Logistic noise model. We also show that if one assumes a different noise model (e.g., Gaussian), then a different loss function would be more appropriate. Furthermore, we show that irrespective to the amount of unbiased noise, we are able to recover the ground truth parameters if we observe enough noisy data, and we provide precise sample complexity bounds. Section 5 also confirms these theoretical results with careful experimental observations and demonstrates that the method is in fact robust to misspecification of the noise model.

1.4 Dimensionality Reduction

It is natural to ask if linear distance metric learning approaches can be used for linear dimensionality reduction. That is, if one restricts to a rank-k positive semi-definite $\boldsymbol{M}_k$, then we can write $\boldsymbol{M}_k = \boldsymbol{A}_k\boldsymbol{A}_k^t$ where $\boldsymbol{A}_k \in \mathbb{R}^{d\times k}$. Hence $\boldsymbol{A}_k$ can be used as a linear map $x' = \boldsymbol{A}_k^t x$ from $\mathbb{R}^d \to \mathbb{R}^k$. Yet optimizing $R_N(\boldsymbol{A}_k\boldsymbol{A}_k^t, \tau)$ with $\boldsymbol{A}_k \in \mathbb{R}^{d\times k}$ is not only non-convex, the optimization has non-optimal local minima (Journée et al., 2010).

Another natural approach is to run the optimization with a full rank $\boldsymbol{A}$, and then truncate $\boldsymbol{A}$ by rounding down its smallest $d-k$ singular values to 0. As far as we know, no previous analysis of a distance metric learning (DML) approach has shown if this is effective; a direction (singular vector of $\boldsymbol{A}$) associated with a small singular value of $\boldsymbol{A}$ could potentially have out-sized relevance towards cost function R_N that we seek to optimize, and this step may induce uncontrolled error in R_N.

In this paper, we detail some reasonable assumptions on the data necessary for this singular value rounding scheme to have provable guarantees. The key assumption is that the width of the support of the data distribution is bounded by $\sqrt{F}$ (for some parameter F), and either this support must include measure in regions which assign labels of both Close and Far, or the unbiased noise must be large enough to generate some of each label.

Specifically we consider the following algorithm.
1) Sample $N = N_d(\varepsilon, \delta)$ pairs $\boldsymbol{x}, \boldsymbol{y}$ from a Lebesgue measurable distribution with the width of support at most $\sqrt{F}$ in each direction.
2) Solve for $\hat{\boldsymbol{A}} \in \mathbb{R}^{d\times d}$ and $\hat{\tau} \geq 0$ in $R_N(\boldsymbol{A}\boldsymbol{A}^t, \tau)$ using any convex gradient descent solver.
3) For positive integer $k \leq d$, set the $d-k$ smallest singular values of $\hat{\boldsymbol{A}}$ to 0, resulting in low-rank matrix $\hat{\boldsymbol{A}}_k$. Let γ be the value of the $(d-k)$-th singular value of $\hat{\boldsymbol{A}}$ (the largest one rounded to 0).
4) Return $\hat{\boldsymbol{M}}_k = \hat{\boldsymbol{A}}_k\hat{\boldsymbol{A}}_k^t$, a rank-$k$ positive semi-definite matrix.

For this algorithm (formalized in Theorems 14 and 15), we claim with probability at least $1-\delta$

$$|R(\hat{\boldsymbol{M}}_k, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*)| \leq \varepsilon + F\gamma^2.$$

Thus, for instance if we draw $\boldsymbol{x} - \boldsymbol{y}$ from a unit ball (so $F = 1$), and set $\varepsilon' = 2\varepsilon$, and assume that $\boldsymbol{M}^*$ has $d-k$ eigenvalues less than $\varepsilon'/4$, then with probability at least $1-\delta$

$$|R(\hat{\boldsymbol{M}}_k, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*)| \leq \varepsilon'.$$

1.5 Outline

In Section 2 we more carefully unroll this model formulation, and in Section 3 we show sample complexity and convergence results, including under noise. In Section 4 we discuss the optimization procedure and show that the unconstrained optimization approach is provably effective, and

useful for dimensionality reduction tasks. Finally, in Section 5 we verify our theory on a variety of synthetic data experiments and demonstrate the utility of this linear DML framework on two real data problems that benefit from a learned Mahalanobis distance.

2. Model and Key Observations

This section contains the model, the underlying assumptions, and some observations used throughout the paper.

2.1 Data and Model Assumptions

We work under the following data model throughout the paper. We assume there exists some positive semi-definite $\boldsymbol{M}^* \in \mathbb{R}^{d\times d}$ that defines a Mahalanobis distance that we seek to discover. We observe pairs of data points $\boldsymbol{x}_i, \boldsymbol{y}_i \in \mathbb{R}^d$, but we only consider the differences of the pairs $\boldsymbol{z}_i = \boldsymbol{x}_i - \boldsymbol{y}_i$. Moreover, we assume that all observations $\boldsymbol{z}_i \in \mathbb{R}^d$ are i.i.d. from some unknown distribution with pdf $f(\boldsymbol{z})$.

We also assume there exists a threshold τ^* which generates labels $\ell_i \in \{\mathrm{Close}, \mathrm{Far}\}$; more specifically, z_i is Close if and only if

$$\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}^*} + \eta_i < \tau^*\,, \qquad \text{(Label Assumption)}$$

where η_i is a noise term. Each η_i is generated i.i.d. from a distribution $\mathrm{Noise}(\eta|0,s)$ which comes from a location-scale family of distributions with location 0 and scale parameter $s>0$. For different distributions, s can have a different meaning. For example, for the normal distribution, s is the standard deviation. For mathematical convenience, we overload "Close" = -1 and "Far" = +1 so $\ell_i \in \{-1,+1\}$. As $\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}^*} + \eta_i < \tau^*$ is equivalent to $\|\boldsymbol{z}_i\|^2_{\frac{\boldsymbol{M}^*}{t}} + \frac{\eta_i}{t} < \frac{\tau^*}{t}$ for any $t>0$, scaling $\boldsymbol{M}^*$, τ^*, and s by a common parameter t does not change the labeling distribution, so w.l.o.g., we remove this degree of freedom in the analysis presented in this paper by setting $s=1$. Thus what the techniques in this paper ultimately recover is actually $\frac{\boldsymbol{M}^*}{s}$ and $\frac{\tau^*}{s}$. In other words, we will see that our model is identifiable if $s=1$. Moreover, the following lemma indicates that for the noiseless case, the model is identifiable provided that $\tau = 1$. The following formalization is proved in Appendix A.

Lemma 1 *Given two pairs $(\boldsymbol{M}_1,\tau_1)$ and $(\boldsymbol{M}_2,\tau_2)$ such that $\boldsymbol{M}_1, \boldsymbol{M}_2 \succeq 0$ and $\tau_1,\tau_2>0$, if the two functions $\boldsymbol{z}\mapsto \mathbb{1}_{\left\{\|\boldsymbol{z}\|^2_{\boldsymbol{M}_1}-\tau_1\geq 0\right\}}$ and $\boldsymbol{z}\mapsto \mathbb{1}_{\left\{\|\boldsymbol{z}\|^2_{\boldsymbol{M}_2}-\tau_2\geq 0\right\}}$ agree for all $\boldsymbol{z}$, then $\frac{\boldsymbol{M}_1}{\tau_1} = \frac{\boldsymbol{M}_2}{\tau_2}$.*

In order to be able to solve our optimization challenges and bound the sample complexity, we need to make a few simple assumptions on $\boldsymbol{M}^*, \tau^*$ and the data distribution. We have the following assumption on the model:

$$\|\boldsymbol{M}^*\|_2 \leq \beta \;\text{ and }\; \tau^* \in [0,B]\,. \qquad \text{(Model Assumption)}$$

Note that as we have assumed that $s=1$, $\boldsymbol{M}^*$ and τ^*, as well as the upper bounds β and B, have been scaled by $1/s$. Since β and B later appear in the sample complexity (see Section 5.3.2), noise affects the sample complexity through these terms.

Next we assume something about the data we observe. Assume $\boldsymbol{z}_1,\ldots,\boldsymbol{z}_N \in \mathbb{R}^d$ are N i.i.d. samples from an unknown distribution with probability density function $f(\boldsymbol{z})$ with respect to the

Lebesgue measure on $\mathbb{R}^d$ (f is Lebesgue measureable whose Lebesgue integral over $\mathbb{R}^d$ is 1). Hence $\int_{\mathbb{R}^d} f(\boldsymbol{z})\mathrm{d}x = 1$ which implies that the set $\{\boldsymbol{z}\colon f(\boldsymbol{z}) \neq 0\}$ has a positive Lebesgue measure. We also assume that the probability density function $f(\boldsymbol{z})$ has bounded support, i.e.,

$$\max\{\|\boldsymbol{z}\|^2\colon f(\boldsymbol{z}) \neq 0\} \leq F. \qquad \text{(Data Assumption)}$$

By this assumption, we know that almost surely $\|\boldsymbol{z}\|^2 \leq F$. Also, by the Data Assumption, for each

$$\boldsymbol{M} \in \mathcal{M}_d = \left\{\boldsymbol{M} \in \mathbb{R}^{d\times d} : \boldsymbol{M} \text{ is p.s.d., } \|\boldsymbol{M}\|_2 \leq \beta\right\},$$

we have $\|\boldsymbol{z}\|^2_{\boldsymbol{M}} \leq \|\boldsymbol{M}\|_2\|\boldsymbol{z}\|^2 \leq \beta F$. When it is clear by context, we sometimes write $\mathcal{M}$ for $\mathcal{M}_d$. Accordingly, for $\boldsymbol{z} \sim f(\boldsymbol{z})$, almost always

$$\left|\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau\right| \leq \max\{B, \beta F\} \qquad \forall(\boldsymbol{M}, \tau) \in \mathcal{M}_d \times [0, B].$$

Since the sign of $\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau$ determines the label of $\boldsymbol{z}$, it is reasonable to assume that

$$B \leq \beta F, \qquad \text{(Meta Assumption)}$$

which implies

$$\left|\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau\right| \leq \beta F \qquad \forall(\boldsymbol{M}, \tau) \in \mathcal{M}_d \times [0, B]. \tag{2}$$

2.2 Convexity

Our core objective is optimizing R_N or R over $(\boldsymbol{M}, \tau)$ such that $\boldsymbol{M} \succeq 0, \tau \geq 0$. In the same vein as regular supervised learning scenarios, we consider loss functions that are convex with respect to the contribution of each data point $(\boldsymbol{z}_i, \ell_i)$. To show the loss functions over the space of valid parameters are convex, we first show that any convex combination of two valid models' parameters is still a valid parameter and for any two models with parameters $(\boldsymbol{M}_1, \tau_1)$ and $(\boldsymbol{M}_2, \tau_2)$ and an interpolation parameter $\lambda \in [0, 1]$, we have

$$\|\boldsymbol{z}\|^2_{\lambda\boldsymbol{M}_1+(1-\lambda)\boldsymbol{M}_2} - (\lambda\tau_1 + (1-\lambda)\tau_2) = \lambda(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_1} - \tau_1) + (1-\lambda)(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_2} - \tau_2).$$

Hence, the convex interpolation of the two models gives us another model with parameter $(\boldsymbol{M} = \lambda\boldsymbol{M}_1 + (1-\lambda)\boldsymbol{M}_2, \tau = \lambda\tau_1 + (1-\lambda)\tau_2)$ which is in the space of the valid parameters. Coupled with a convex loss function, this implies that minimizing R_N or R over $(\boldsymbol{M}, \tau)$ with $\boldsymbol{M} \succeq 0$ and $\tau \geq 0$ is a convex optimization problem. Hence any critical point is a global minimum. However, restricting $\boldsymbol{M} \succeq 0$ under gradient descent is non-trivial since generically the gradient may push the solution out of that. While manifold optimization methods have been developed for other matrix optimization challenges (Balzano et al., 2010; Vandereycken, 2013), we develop a simpler unconstrained approach in this work.

3. Noise Observations and Optimal Loss Functions

Recall that $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N \in \mathbb{R}^d$ are N i.i.d. samples from an unknown distribution with probability density function $f(\boldsymbol{z})$ with bounded support. As the noise distribution $\mathrm{Noise}(\eta|0, 1)$ makes the

labeling probabilistic, for a given $\boldsymbol{z} \sim f(\boldsymbol{z})$, the probability that the corresponding label is 1 can be computed as

$$\begin{aligned} P(\ell = 1|\boldsymbol{z}; \boldsymbol{M}, \tau) &= P(\eta > \tau - \|\boldsymbol{z}\|_{\boldsymbol{M}}^2) \\ &= \int_{-\infty}^{\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau} \mathrm{Noise}(\eta|0,1)\mathrm{d}\eta \\ &= \Phi_{\mathrm{Noise}}(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau), \end{aligned} \tag{3}$$

where $\Phi_{\mathrm{Noise}}(a) = \int_{-\infty}^{a} \mathrm{Noise}(\eta|0,1)\mathrm{d}\eta$. Observe that

$$P(\ell = -1|\boldsymbol{z}; \boldsymbol{M}, \tau) = 1 - P(\ell = 1|\boldsymbol{z}; \boldsymbol{M}, \tau) = \Phi_{\mathrm{Noise}}(-1(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau)).$$

Consequently, we have

$$\boldsymbol{z}, \ell \sim g(\boldsymbol{z}, \ell; \boldsymbol{M}, \tau) = f(\boldsymbol{z})\Phi_{\mathrm{Noise}}\left(\ell(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau)\right),$$

where g is the density of the random variable (z, ℓ) with respect to $\mu_L \otimes \nu$ (the product of the Lebesgue measure μ_L and the counting measure ν). Our theoretical results hold under quite general assumptions on the noise. In particular, the noise should be symmetric, continuous, and $-\log \Phi_{\mathrm{Noise}}$ should be convex, one-to-one, and ζ-Lipschitz on $[-\beta F, \beta F]$. Throughout the paper, we refer to any noise satisfying these assumptions as a *simple noise model*, and specifically consider the Logistic, Normal, Laplace, and Hyperbolic secant (HS) distributions as special cases. Table 1 describes these distributions and their associated model constants (see Appendix E for verification of these constants). Note for the Logistic and Hyperbolic models, $\mathrm{sech}(t) = \frac{1}{\cosh(t)} = \frac{2}{e^t + e^{-t}}$, and in the remainder of this article we will refer to $\mathrm{Noise}(\eta|0,1)$ as $\mathrm{Noise}(\eta)$ for brevity. One could consider other noise models like Cauchy, but then $-\log \Phi_{\mathrm{Noise}}$ is not convex. In Figure 1, we compare the four simple noise distributions when they share the same mean and variance.

Noise	Logistic $\mathrm{L}(\eta\|0,1)$	Normal $\mathcal{N}(\eta\|0,1)$	Laplace $f(\eta\|0,1)$	Hyperbolic $\mathrm{HS}(\eta\|0,1)$
pdf	$\frac{1}{4}\mathrm{sech}^2\left(\frac{\eta}{2}\right)$	$\frac{1}{\sqrt{2\pi}}e^{-\frac{\eta^2}{2}}$	$\frac{1}{2}e^{-\lvert\eta\rvert}$	$\frac{1}{2}\mathrm{sech}(\frac{\pi}{2}\eta)$
std	$\frac{\pi}{\sqrt{3}}$	1	$\sqrt{2}$	1
ζ (cdf is ζ-log-Lipschitz)	1	$O(\beta F)$	1	$\frac{\pi}{2}$
$\omega = \min_{\lvert\eta\rvert \le \beta F} \mathrm{Noise}(\eta)$	$e^{-O(\beta F)}$	$e^{-O(\beta^2 F^2)}$	$e^{-O(\beta F)}$	$e^{-O(\beta F)}$
$T = \max_{\lvert\eta\rvert \le \beta F} -\log \Phi_{\mathrm{Noise}}(\eta)$	$O(\beta F)$	$O((\beta F)^2)$	$O(\beta F)$	$O(\beta F)$

Table 1: Simple noise models and key properties and values.

Note by the Label Assumption,

$$(\boldsymbol{z}_1, \ell_1), \ldots, (\boldsymbol{z}_N, \ell_N) \overset{i.i.d.}{\sim} g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*) = f(\boldsymbol{z})\Phi_{\mathrm{Noise}}\left(\ell(\|\boldsymbol{z}\|_{\boldsymbol{M}^*}^2 - \tau^*)\right). \tag{4}$$

Since we have a probabilistic model, we now use the MLE method to estimate $\boldsymbol{M}^*, \tau^*$. As one of the main contributions of this work, we will prove that this method works and deduce the corresponding

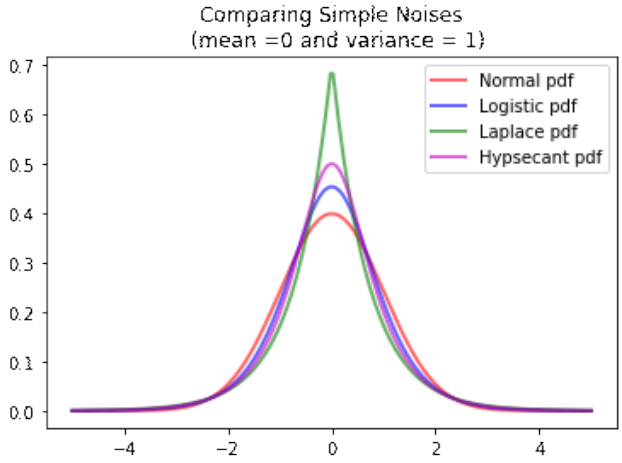

Figure 1: Comparing simple noise models when $\mu = 0$ and $\sigma^2 = 1$.

sample complexity. The average negative log-likelihood of the given data $\boldsymbol{z}_1 \ldots, \boldsymbol{z}_N$ and their labels $\ell_i, \ldots, \ell_N$ as a function of $\boldsymbol{M}$ and τ is

$$\begin{aligned}
\mathrm{NLL}(\boldsymbol{M}, \tau) &= -\frac{1}{N} \sum_{i=1}^{N} \log g(\boldsymbol{z}_i, \ell_i; \boldsymbol{M}, \tau) \\
&= -\frac{1}{N} \sum_{i=1}^{N} [\log f(\boldsymbol{z}_i) + \log p(\ell_i | \boldsymbol{z}_i; \boldsymbol{M}, \tau)] \\
&= \underbrace{-\frac{1}{N} \sum_{i=1}^{N} \log f(\boldsymbol{z}_i)}_{\text{independent of } \boldsymbol{M}, \tau} \underbrace{-\frac{1}{N} \sum_{i=1}^{N} \log p(\ell_i | \boldsymbol{z}_i; \boldsymbol{M}, \tau)}_{\text{the loss function}} .
\end{aligned}$$

Therefore, to solve the MLE, we need to find a p.s.d. matrix $\boldsymbol{M}$ and a $\tau \in [0, B]$ minimizing

$$R_N(\boldsymbol{M}, \tau) = -\frac{1}{N} \sum_{i=1}^{N} \log p(\ell_i | \boldsymbol{x}_i; \boldsymbol{M}, \tau) = -\frac{1}{N} \sum_{i=1}^{N} \log \Phi_{\text{Noise}} \left(\ell_i (\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2 - \tau) \right) .$$

Thus the optimization problem we are dealing with is

$$\min_{\boldsymbol{M} \succeq 0, \tau \geq 0} R_N(\boldsymbol{M}, \tau), \tag{5}$$

where $R_N(\boldsymbol{M}, \tau) = -\frac{1}{N} \sum_{i=1}^{N} \log \Phi_{\text{Noise}} \left(\ell_i (\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2 - \tau) \right)$.

We will justify that solving this optimization problem with high probability ends up in a guaranteed approximation of $(\boldsymbol{M}^*, \tau^*)$. For fixed but arbitrary $\boldsymbol{M}, \tau$, using Chebyshev's inequality (or directly by the Law of Large Numbers), we know that $R_N(\boldsymbol{M}, \tau)$ tends to (in measure induced by

$g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*)$)

$$R(\boldsymbol{M}, \tau) = -\mathbb{E}_{z,\ell\sim g(\boldsymbol{z},\ell;\boldsymbol{M}^*,\tau^*)} \log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}} - \tau)\right) \tag{6}$$
$$= -\int g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*) \log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}} - \tau)\right) d\boldsymbol{z} d\ell$$

provided that $\log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}} - \tau)\right)$ has a bounded variance, which is the case since $f(\boldsymbol{z})$ has bounded support. Thus we can view $R_N(\boldsymbol{M}, \tau)$ as the empirical risk function and $R(\boldsymbol{M}, \tau)$ as the true risk function. However, we only have access to $R_N(\boldsymbol{M}, \tau)$. The rest of this subsection can be summarized as follows:

1. We prove that both functions $R(\boldsymbol{M}, \tau)$ and $R_N(\boldsymbol{M}, \tau)$ are convex, so we are dealing with convex optimization problems.
2. We prove that $R(\boldsymbol{M}, \tau)$ is uniquely minimized at $(\boldsymbol{M}^*, \tau^*)$.
3. We show that $R_N(\boldsymbol{M}, \tau)$ converges uniformly in measure to $R(\boldsymbol{M}, \tau)$. We also bound the corresponding error for an arbitrarily given confidence bound.
4. Combining these results, we conclude that minimizing $R_N(\boldsymbol{M}, \tau)$ is a good proxy for minimizing $R(\boldsymbol{M}, \tau)$.

Theorem 2 *The true loss $R(\boldsymbol{M}, \tau)$ is uniquely minimized at $(\boldsymbol{M}^*, \tau^*)$.*

Proof First, note that

$$\begin{aligned} R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}^*, \tau^*) &= \mathbb{E}_{\boldsymbol{z},\ell\sim g(\boldsymbol{z},\ell;\boldsymbol{M}^*,\tau^*)} \left(\log \frac{\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*} - \tau^*))}{\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}} - \tau))}\right) \\ &= \mathbb{E}_{\boldsymbol{z},\ell\sim g(\boldsymbol{z},\ell;\boldsymbol{M}^*,\tau^*)} \left(\log \frac{f(\boldsymbol{z})\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*} - \tau^*))}{f(\boldsymbol{z})\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}} - \tau))}\right) \\ &= \mathrm{D}_{\mathrm{KL}}(g(\boldsymbol{x}, \ell; \boldsymbol{M}^*, \tau^*) \| g(\boldsymbol{x}, \ell; \boldsymbol{M}, \tau)) \geq 0. \end{aligned}$$

This indicates that $R(\boldsymbol{M}, \tau)$ takes its minimum at $(\boldsymbol{M}^*, \tau^*)$. Moreover, for any $(\boldsymbol{M}^+, \tau^+)$ at which $R(\boldsymbol{M}, \tau)$ attains its minimum, $\mathrm{D}_{\mathrm{KL}}(g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*) \| g(\boldsymbol{z}, \ell; \boldsymbol{M}^+, \tau^+)) = 0$. This implies that $g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*) = g(\boldsymbol{z}, \ell; \boldsymbol{M}^+, \tau^+)$ almost everywhere (according to the probability measure induced by $g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*)$ over $\mathbb{R}^d \times \{-1, 1\}$). Since $\mu_L(\{\boldsymbol{z} \colon f(\boldsymbol{z}) > 0\}) > 0$ (Lebesgue measure) and $\log \Phi_{\text{Noise}}(\cdot)$ is one-to-one on $[-\beta F, \beta F]$, we can conclude that there is a set $S \subseteq \mathbb{R}^d$ such that $\mu_L(S) > 0$ and for every $\boldsymbol{z} \in S$,

$$\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*} - \tau^* = \|\boldsymbol{z}\|^2_{\boldsymbol{M}^+} - \tau^+.$$

By Lemma 16 in Appendix B, we then conclude $\boldsymbol{M}^+ = \boldsymbol{M}^*$ and $\tau^+ = \tau^*$. ■

Although we have proved that the true loss $R(\boldsymbol{M}, \tau)$ is uniquely minimized at $(\boldsymbol{M}^*, \tau^*)$, in reality, we do not have access to the true loss, but only to the empirical loss $R_N(\boldsymbol{M}, \tau)$. Next we will show that $R_N(\boldsymbol{M}, \tau)$ is uniformly close to $R(\boldsymbol{M}, \tau)$ as N gets large, and then conclude that instead of minimizing $R(\boldsymbol{M}, \tau)$, we can minimize $R_N(\boldsymbol{M}, \tau)$ to approximate $(\boldsymbol{M}^*, \tau^*)$. Note that for two given p.s.d. $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$,

$$|\|\boldsymbol{x}\|^2_{\boldsymbol{M}_1} - \|\boldsymbol{x}\|^2_{\boldsymbol{M}_2}| \leq \|\boldsymbol{M}_1 - \boldsymbol{M}_2\|_2 \|\boldsymbol{x}\|^2. \tag{7}$$

For proof, see Appendix D: Observation 3. In the next lemma, using this inequality, we prove that the true loss and empirical loss are both Lipschitz with respect to the metric

$$\mathrm{d}((\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2)) = \|\boldsymbol{M}_1 - \boldsymbol{M}_2\|_2 + |\tau_1 - \tau_2|.$$

Lemma 3 *If* $\log \Phi_{\text{Noise}}(\cdot)$ *is* ζ*-Lipschitz, then, for any given* $(\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2) \in \mathcal{M} \times [0, B]$*,*

1. $|R(\boldsymbol{M}_1, \tau_1) - R(\boldsymbol{M}_2, \tau_2)| < \zeta \max(F, 1)\mathrm{d}((\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2))$,
2. $|R_N(\boldsymbol{M}_1, \tau_1) - R_N(\boldsymbol{M}_2, \tau_2)| < \zeta \max(F, 1)\mathrm{d}((\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2))$.

Proof We start with the proof of the first inequality. In view of Equation 6, we have

$$\begin{aligned}
|R(\boldsymbol{M}_1, \tau_1) - R(\boldsymbol{M}_2, \tau_2)| &= \Big|\mathbb{E}_{\boldsymbol{z},\ell}\Big[\log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_1} - \tau_1)\right) - \log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_2} - \tau_2)\right)\Big]\Big| \\
&\leq \mathbb{E}_{\boldsymbol{z},\ell}\Big|\log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_1} - \tau_1)\right) - \log \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_2} - \tau_2)\right)\Big| \\
(\log \Phi_{\text{Noise}}(\cdot) \text{ is } \zeta\text{-Lipschitz}) \quad &\leq \zeta\mathbb{E}_{\boldsymbol{z}}\Big[\Big|(\|\boldsymbol{z}\|^2_{\boldsymbol{M}_1} - \tau_1) - (\|\boldsymbol{z}\|^2_{\boldsymbol{M}_2} - \tau_2)\Big|\Big] \\
(\text{using 7}) \quad &\leq \zeta\mathbb{E}_{\boldsymbol{z}}\Big|\boldsymbol{z}^t(\boldsymbol{M}_1 - \boldsymbol{M}_2)\boldsymbol{z}\Big| + \zeta\mathbb{E}\Big[|\tau_1 - \tau_2|\Big] \\
&\leq \zeta\|(\boldsymbol{M}_1 - \boldsymbol{M}_2)\|_2\mathbb{E}\Big[\|\boldsymbol{z}\|^2\Big] + \zeta|\tau_1 - \tau_2| \\
(\text{Data Assumption}) \quad &\leq \zeta\|(\boldsymbol{M}_1 - \boldsymbol{M}_2)\|_2 F + \zeta|\tau_1 - \tau_2| \\
&< \zeta \max(F, 1)\mathrm{d}\big((\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2)\big),
\end{aligned}$$

where $(\boldsymbol{z}, \ell) \sim g(\boldsymbol{z}, \ell; \boldsymbol{M}^*, \tau^*)$ (see Equation 4). The same arguments apply to the second result. ■

As $-\log \Phi_{\text{Noise}}(\cdot)$ is a decreasing function, using Equation 2, we have

$$0 \leq -\log \Phi_{\text{Noise}}\left(\ell_i(\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau)\right) \leq -\log \Phi_{\text{Noise}}(-\beta F) = T, \tag{8}$$

which indicates that the random variables $-\log \Phi_{\text{Noise}}\left(\ell_i(\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau)\right)$ are bounded by a value T; see Table 1. In the next theorem, we prove that, with high probability, the empirical loss R_N is everywhere close to the true loss R. We provide a sketch of the proof here; for the complete proof, see Appendix D. The full bound for $N_d(\varepsilon, \delta)$ appears in 14; it is a polynomial function of $F, T, \log(B), \log(\beta)$, but for simplicity of presentation we omit the precise dependence on these constants.

Theorem 4 *Assume that the noise model* $\text{Noise}(\eta)$ *is simple. For any* $\varepsilon, \delta > 0$*, define*

$$N_d(\varepsilon, \delta) = O\left(\frac{1}{\varepsilon^2}\Big[\log\frac{1}{\delta} + d^2 \log\frac{d}{\varepsilon}\Big]\right).$$

If $N > N_d(\varepsilon, \delta)$*, then with probability at least* $1 - \delta$*,*

$$\sup_{(\boldsymbol{M},\tau)\in\mathcal{M}\times[0,B]} |R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| < \varepsilon.$$

Proof Set $\alpha = \frac{\varepsilon}{3\zeta \max(F,1)}$. Consider $\mathcal{E} = \{(\boldsymbol{M}_i, \tau_i); i = 1, \ldots, m = m(\alpha)\}$ as an α-cover for $\mathcal{M} \times [0, B]$. By Lemma 3, for every $(\boldsymbol{M}, \tau)$, there exists an index $i \in [m]$ such that

$$|R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}_i, \tau_i)| < \frac{\varepsilon}{3} \qquad \text{and} \qquad |R_N(\boldsymbol{M}, \tau) - R_N(\boldsymbol{M}_i, \tau_i)| < \frac{\varepsilon}{3}$$

which concludes $|R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| \leq \frac{2\varepsilon}{3} + |R_N(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}_i, \tau_i)|$. Using an appropriate upper bound for $m(\alpha) = \frac{B}{\alpha}(4\beta d\sqrt{d}/\alpha)^{d^2}$ in Lemma 17, and applying a Chernoff-Hoeffding bound and union bound, we conclude the desired result (for full details, see Appendix D). ■

Note that combining Theorem 2 and Theorem 4, we conclude that minimizing $R_N(\boldsymbol{M}, \tau)$ is a good proxy for minimizing $R(\boldsymbol{M}, \tau)$. We restate this result in the next theorem.

Theorem 5 *Assume that the noise model* $\mathrm{Noise}(\eta)$ *is simple. For any given* $\varepsilon, \delta > 0$*, if* $N > N(\frac{\varepsilon}{2}, \delta)$*, then, with probability at least* $1 - \delta$*, for any point* $(\hat{\boldsymbol{M}}, \hat{\tau})$ *minimizing* $R_N(\boldsymbol{M}, \tau)$*, we have*

$$0 < R(\hat{\boldsymbol{M}}, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*) < \varepsilon.$$

Proof As $N > N(\frac{\varepsilon}{2}, \delta)$, by Theorem 4, with probability at least $1 - \delta$, we have

$$|R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| < \frac{\varepsilon}{2} \qquad \text{for all } (\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B].$$

Consequently, with probability at least $1 - \delta$,

$$\begin{aligned} R(\hat{\boldsymbol{M}}, \hat{\tau}) - \frac{\varepsilon}{2} &< R_N(\hat{\boldsymbol{M}}, \hat{\tau}) \\ &\leq R_N(\boldsymbol{M}^*, \tau^*) \qquad (R_N(\boldsymbol{M}, \tau) \text{ minimized at } (\hat{\boldsymbol{M}}, \hat{\tau})) \\ &< R(\boldsymbol{M}^*, \tau^*) + \frac{\varepsilon}{2}, \end{aligned}$$

implying that

$$0 \leq R(\hat{\boldsymbol{M}}, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*) \leq \varepsilon.$$

■

In the next four subsections, we consider the simple noise models determined by the Logistic, Gaussian, Laplace, and Hyperbolic secant distributions.

3.1 Logistic Distribution as the Noise

The probability density function of the Logistic distribution is

$$\mathrm{L}(x|\mu, s) = \frac{1}{4s}\mathrm{sech}^2\left(\frac{x - \mu}{2s}\right),$$

where μ is the mean, s is the scale parameter of this distribution, and $\frac{s^2\pi^2}{3}$ is the variance. The Logistic distribution looks very much like a normal distribution and is sometimes used as an approximation for it. In this subsection, we assume that the noise has a Logistic distribution with $\mu = 0$ and $s = 1$, i.e., $\mathrm{Noise}(\eta) = \mathrm{L}(\eta|0, 1)$ (since scaling $\boldsymbol{M}^*, \tau^*$, and η does not change the

labeling distribution, w.l.o.g. we may assume $s = 1$). The Cumulative distribution function of $\mathrm{L}(x|0,1)$ is the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, i.e., $\Phi_{\mathrm{L}}(x) = \sigma(x)$. As the Logistic noise is simple with $\zeta = 1$ (see Table 1), we can apply Theorem 4. Plugging $\sigma(x)$ in for $\Phi_{\mathrm{Noise}}(x)$ in Optimization Problem 5, we obtain $R_N(\boldsymbol{M}, \tau) = -\frac{1}{N}\sum_{i=1}^{N} \log \sigma\left(\ell_i(\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau)\right)$.

Since in the Logistic case we have a closed form for $\Phi_{\mathrm{L}}(x) = \sigma(x)$, the loss function becomes computationally easier to work with. As the main setting for the paper, we will thus assume that the noise comes from a Logistic distribution, although we consider other noise models and loss functions in our experiments.

3.2 Normal Distribution as the Noise

If we assume the noise has a Normal instead of a Logistic distribution, then $\Phi_{\mathrm{Noise}}(x)$ becomes a probit function instead of the sigmoid function. Indeed, if we set $\mathrm{Noise}(\eta) = \mathcal{N}(\eta|0,1)$, then

$$\Phi(a) = \Phi_{\mathrm{Noise}}(a) = \int_{-\infty}^{a} \mathcal{N}(\eta|0,1)\mathrm{d}\eta,$$

which is known as the *probit function*. Once again plugging Φ into $R_N(\boldsymbol{M}, \tau)$ in Optimization Problem 5, we obtain $R_N(\boldsymbol{M}, \tau) = -\frac{1}{N}\sum_{i=1}^{N} \log \Phi\left(\ell_i(\|\boldsymbol{z}_i\|^2_{\boldsymbol{M}} - \tau)\right)$. Unfortunately, the probit function has no closed-form formula, so optimizing R_N is not as simple as in the Logistic case. However we will observe in Section 5 that since the Logistic and Gaussian distributions are very similar, the Logistic loss does well under Normal noise.

3.3 Laplace Distribution as the Noise

As the third natural option for noise, we assume that

$$\mathrm{Noise}(\eta) = \mathrm{Laplace}(\eta|0,1) = \frac{1}{2}e^{-|\eta|}.$$

In this setting,

$$\Phi_{\mathrm{Laplace}}(a) = \int_{-\infty}^{a} \frac{1}{2}e^{-|\eta|}\mathrm{d}\eta = \begin{cases} \frac{1}{2}e^{a} & a \leq 0 \\ 1 - \frac{1}{2}e^{-a} & a \geq 0. \end{cases}$$

Similar to the Logistic case, we obtain a closed-form formula for $\Phi_{\mathrm{Laplace}}(a)$, so that this setting is also convenient to work with. Note that

$$-\log \Phi_{\mathrm{Laplace}}(a) = \begin{cases} -a + \log 2 & a \leq 0 \\ -\log\left(1 - \frac{1}{2}e^{-a}\right) & a \geq 0 \end{cases},$$

which yields a closed-form formula for R_N.

3.4 Hyperbolic Secant Distribution as the Noise

The hyperbolic secant distribution is a continuous probability distribution whose probability density function is

$$\mathrm{HS}(\eta|\mu,\sigma) = \frac{1}{2\sigma}\mathrm{sech}(\frac{\pi}{2\sigma}(\eta - \mu))$$

and whose Cumulative distribution function is

$$\Phi_{\mathrm{HS}}(\eta|\mu,\sigma) = \frac{2}{\pi}\arctan(\exp(\frac{\pi}{2\sigma}(\eta-\mu))).$$

The mean and variance of this distribution are μ and σ^2 respectively. As the last option for noise, we consider $\mathrm{Noise}(\eta) = \mathrm{HS}(\eta|0,1)$. In this case, we have

$$\Phi_{\mathrm{HS}}(a) = \frac{2}{\pi}\arctan\left(\exp(\frac{\pi}{2}\eta)\right).$$

Plugging Φ_{HS} into $R_N(\boldsymbol{M},\tau)$ in Optimization Problem 5, we obtain

$$R_N(\boldsymbol{M},\tau) = -\frac{1}{N}\sum_{i=1}^{N}\log\left[\arctan\left(\exp\left(\frac{\pi}{2}\left(-\ell_i(\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2-\tau)\right)\right)\right)\right] + \mathrm{Constant}\,,$$

and we can ignore the constant term.

4. Algorithms, Approximation, and Dimensionality Reduction

In this part, we mainly explore some properties of Optimization Problem 5 and consider the potential ways to solve it.

4.1 How to Solve Optimization Problem 5

We will prove that solving Optimization Problem 5 yields close approximations of the parameters $\boldsymbol{M}^*$ and τ^*. We restate the optimization problem here:

$$\min_{\boldsymbol{M}\succeq 0,\tau\geq 0} R_N(\boldsymbol{M},\tau). \tag{9}$$

As we need to enforce $\boldsymbol{M}$ to be p.s.d., using gradient descent directly is difficult. Notice that $\boldsymbol{M}$ is p.s.d. if and only if $\boldsymbol{M} = \boldsymbol{A}\boldsymbol{A}^\top$ for some $\boldsymbol{A}_{d\times k}$ where $k \leq d$; in this case $\boldsymbol{M}$ indeed has rank at most k. Therefore, if we replace $\boldsymbol{M}$ by $\boldsymbol{A}\boldsymbol{A}^\top$ and optimize over $\boldsymbol{A}$, we no longer need to maintain the p.s.d. condition on $\boldsymbol{M}$. Then the optimization problem can be rewritten as follows:

$$\min_{\tau\geq 0,\boldsymbol{A}\in\mathbb{R}^{d\times k}} R_N(\boldsymbol{A}\boldsymbol{A}^\top,\tau). \tag{10}$$

If we set $k = d$, then Optimization 10 is equivalent to Optimization 5. The only downside of this reformulation is that we lose the convexity by this change of variable. So we are dealing with a non-convex optimization, and thus there may be no guarantee that gradient descent will converge to a global minimum. Fortunately, the next theorem proved by Journée et al. (2010) resolves this issue. We remind the reader that for convex optimization problems, global minimums and stationary points are equivalent.

Theorem 6 (Journée et al., 2010) *A local minimizer $\boldsymbol{A}^*$ of Problem 10 provides a stationary point (global minimum) $\boldsymbol{M} = \boldsymbol{A}^*(\boldsymbol{A}^*)^\top$ of Problem 9 if $\boldsymbol{A}^*$ is rank deficient $(\mathrm{rank}(\boldsymbol{A}^*) < k)$. Moreover, if $d = k$, then any local minimizer $\boldsymbol{A}^*$ of Problem 10 provides a stationary point (global minimum) $\boldsymbol{M}^* = \boldsymbol{A}^*(\boldsymbol{A}^*)^\top$ of Problem 9.*

So, we can use gradient descent for $k = d$ to find a local minimum $\boldsymbol{A}^*$ of Problem 9, then using this theorem we know that $\boldsymbol{M}^* = \boldsymbol{A}^*(\boldsymbol{A}^*)^\top$ is a global minimum of Problem 5. Another possible approach is to try some $k < d$, and if $\boldsymbol{A}^*$ is rank deficient, then again $\boldsymbol{M}^* = \boldsymbol{A}^*(\boldsymbol{A}^*)^\top$ is a global minimum of Problem 5. However even if we know that the solution for Problem 9 has rank $r < k$, we might find $\boldsymbol{A}^*$ to be full rank. Indeed, setting $k > r$ does not imply that $\boldsymbol{A}^*(\boldsymbol{A}^*)^\top$ is the global minimum of Problem 9. Moreover, Problem 9 is a generalization of Low-Rank Semi-definite Programming, which is known to be an NP-Hard problem (Anjos and Wolkowicz, 2002) (weighted Max-Cut is a special case of it), which indicates that solving it when $k < d$ might be a difficult task.

4.2 How Well is (M^*, τ^*) Approximated?

Throughout this section, for simplicity of notation and w.l.o.g we assume that $\zeta F = 1$. In this section, we will see that, with high probability, we can approximate $(\boldsymbol{M}^*, \tau^*)$ with any given precision if N is large enough. Recall that Theorem 2 establishes that $R(\boldsymbol{M}, \tau)$ is uniquely minimized at $(\boldsymbol{M}^*, \tau^*)$. Theorem 5 asserts that if N is large enough, the value of the true loss on parameters minimizing the empirical loss, i.e. $R(\hat{\boldsymbol{M}}, \hat{\tau})$, is close to the minimum of the true loss, $R(\boldsymbol{M}^*, \tau^*)$. Although we can infer from this theorem that the error at the ground truth parameters $(\boldsymbol{M}^*, \tau^*)$ is close to the error at $(\hat{\boldsymbol{M}}, \hat{\tau})$, it is still possible that $(\boldsymbol{M}^*, \tau^*)$ and $(\hat{\boldsymbol{M}}, \hat{\tau})$ are far from each other with respect to the metric $\mathrm{d}((\boldsymbol{M}^*, \tau^*), (\hat{\boldsymbol{M}}, \hat{\tau})) = \|\boldsymbol{M}^* - \hat{\boldsymbol{M}}\|_2 + |\tau^* - \hat{\tau}|$.

Recall that the random variable $\boldsymbol{z} \in \mathbb{R}^d$ is generated from an unknown distribution with probability density function $f(\boldsymbol{z})$ with bounded support. Let us define the $L_1(f)$-norm of $(\boldsymbol{M}, \tau)$ as

$$\|(\boldsymbol{M}, \tau)\|_{L_1(f)} = \int f(\boldsymbol{x}) \left|\boldsymbol{x}^t \boldsymbol{M} \boldsymbol{x} - \tau\right| \mathrm{d}\boldsymbol{x}.$$

This is a norm on the vector space $\{(\boldsymbol{M}, \tau) \colon \boldsymbol{M} \in \mathbb{R}^{d\times d} \text{ is symmetric }, \tau \in \mathbb{R}\}$. Note that if $\|(\boldsymbol{M}, \tau)\|_{L_1(f)} = 0$, then Lemma 16 along with the fact that $\mu_L(\{\boldsymbol{x} \colon f(\boldsymbol{x}) > 0\}) > 0$ implies $\boldsymbol{M} = \boldsymbol{0}$ and $\tau = 0$. The other required properties follow by standard reductions. This norm naturally induces the following $L_1(f)$-metric

$$\|(\boldsymbol{M}_1, \tau_1) - (\boldsymbol{M}_2, \tau_2)\|_{L_1(f)} = \int f(\boldsymbol{x}) \left|(\boldsymbol{x}^t \boldsymbol{M}_1 \boldsymbol{x} - \tau_1) - (\boldsymbol{x}^t \boldsymbol{M}_2 \boldsymbol{x} - \tau_2)\right| \mathrm{d}\boldsymbol{x}. \tag{11}$$

Theorem 7 *Assume that the noise model* $\mathrm{Noise}(\eta)$ *is simple and set* $\omega = \min_{|\eta| \le \beta F} \mathrm{Noise}(\eta)$*; see Table 1. Then for all* $(\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B]$

$$R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}^*, \tau^*) \ge \frac{1}{2}\omega^2 \left(\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)}\right)^2.$$

Proof Define $\mu_{(\boldsymbol{M},\tau)}$ to be the measure induced by the probability density function $g(\boldsymbol{x},\ell;\boldsymbol{M},\tau) = f(\boldsymbol{z})\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau))$. Note that

$$\begin{aligned}
R(\boldsymbol{M},\tau)-R(\boldsymbol{M}^*,\tau^*) &= \mathbb{E}_{\boldsymbol{z},\ell\sim g(\boldsymbol{z},\ell;\boldsymbol{M}^*,\tau^*)}\left(\log\frac{\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*))}{\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau))}\right)\\
&= \mathbb{E}_{\boldsymbol{z},\ell\sim g(\boldsymbol{z},\ell;\boldsymbol{M}^*,\tau^*)}\left(\log\frac{f(\boldsymbol{z})\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*))}{f(\boldsymbol{z})\Phi_{\text{Noise}}(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau))}\right)\\
&= \mathrm{D}_{\mathrm{KL}}(g(\boldsymbol{x},\ell;\boldsymbol{M}^*,\tau^*)\|g(\boldsymbol{x},\ell;\boldsymbol{M},\tau))\\
&\geq \frac{1}{2}\left(\left\|\mu_{(\boldsymbol{M}^*,\tau^*)}-\mu_{(\boldsymbol{M},\tau)}\right\|_{\text{TV}}\right)^2
\end{aligned}$$

where $\|\mu_{(\boldsymbol{M}^*,\tau^*)}-\mu_{(\boldsymbol{M},\tau)}\|_{\text{TV}}$ is the total variation of the signed measure $\mu_{(\boldsymbol{M}^*,\tau^*)}-\mu_{(\boldsymbol{M},\tau)}$ and the last line follows from Pinsker's inequality (see Brillinger (1964)). So to find a lower bound for $|R(\boldsymbol{M},\tau)-R(\boldsymbol{M}^*,\tau^*)|$, it suffices to find a lower bound for $\|\mu_{(\boldsymbol{M}^*,\tau^*)}-\mu_{(\boldsymbol{M},\tau)}\|_{\text{TV}}$. To this end,

$$\begin{aligned}
\|\mu_{(\boldsymbol{M}^*,\tau^*)}-\mu_{(\boldsymbol{M},\tau)}\|_{\text{TV}} &= \frac{1}{2}\int f(\boldsymbol{z})\left|\Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)\right)-\Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right)\right|\mathrm{d}\boldsymbol{z}\mathrm{d}\ell\\
\text{(Mean value theorem)}\quad &= \frac{1}{2}\int f(\boldsymbol{z})\left|\Phi'_{\text{Noise}}(\xi(\boldsymbol{z},\ell))\left[\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)-\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right]\right|\mathrm{d}\boldsymbol{z}\mathrm{d}\ell\\
(\Phi'_{\text{Noise}}(\cdot)=\text{Noise}(\cdot))\quad &= \frac{1}{2}\int f(\boldsymbol{z})\text{Noise}(\xi(\boldsymbol{z},\ell))\left|(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)-(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right|\mathrm{d}\boldsymbol{z}\mathrm{d}\ell\\
&\geq \frac{1}{2}\left(\min_{|\xi|\leq\beta F}\text{Noise}(\xi)\right)\times\int f(\boldsymbol{z})\left|(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)-(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right|\mathrm{d}\boldsymbol{z}\mathrm{d}\ell\\
&= \omega\int f(\boldsymbol{z})\left|(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)-(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right|\mathrm{d}\boldsymbol{z}\\
&= \omega\|(\boldsymbol{M},\tau)-(\boldsymbol{M}^*,\tau^*)\|_{L_1(f)},
\end{aligned}$$

where the first step follows from Scheffé's Lemma, which relates the total variation distance to the L_1-norm (see Lemma 2.1 in Tsybakov (2008)), the second step is true since Mean Value Theorem implies that there is a $\xi(\boldsymbol{z},\ell)$ between $\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)$ and $\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)$ such that

$$\begin{aligned}
&\Phi'_{\text{Noise}}(\xi(\boldsymbol{z},\ell))\left[\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)-\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right]\\
&\qquad\qquad = \Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}}-\tau)\right)-\Phi_{\text{Noise}}\left(\ell(\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*}-\tau^*)\right),
\end{aligned}$$

and the fourth step is true since $|\xi(\boldsymbol{z},\ell)|\leq\beta F$. Therefore,

$$R(\boldsymbol{M},\tau)-R(\boldsymbol{M}^*,\tau^*)\geq\frac{1}{2}\omega^2\|(\boldsymbol{M},\tau)-(\boldsymbol{M}^*,\tau^*)\|^2_{L_1(f)}.$$

■

The next corollary is an immediate consequence of Theorems 5 and 7. It indicates that, with high probability, $(\hat{\boldsymbol{M}},\hat{\tau})$ can approximate $(\boldsymbol{M}^*,\tau^*)$ with any given precision with respect to the $L_1(f)$-metric defined in 11 provided that N is large enough with respect to that precision.

Corollary 8 *Assume that the noise model* $\mathrm{Noise}(\eta)$ *is simple. For* $\varepsilon, \delta > 0$*, if* $N > N\left(\frac{1}{2}\varepsilon^2\omega^2, \delta\right)$*, then with probability at least* $1-\delta$

$$\|(\hat{\boldsymbol{M}}, \hat{\tau}) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} \leq \varepsilon.$$

The $L_1(f)$-metric is dependent on the distribution $f(\boldsymbol{z})$, which is unavoidable. The following lemma bounds this norm under a general condition on f, and thus provides a more intuitive result (see Appendix F for the proof).

Lemma 9 *If* $f(\boldsymbol{z}) \geq c > 0$ *for each* $\boldsymbol{z} \in B^d(1) = \{\boldsymbol{x} \in \mathbb{R}^d \colon \|z\|_2 \leq 1\}$*, then for all* $(\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B]$

$$\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} \geq \frac{c\pi^{d/2}}{20\Gamma(d/2+1)} \left(\frac{1}{18}\right)^d \mathrm{d}((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)).$$

In particular, if $f(\boldsymbol{z})$ *is uniform on the unit ball* $B^d(1)$*, then*

$$\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} \geq \frac{1}{20} \left(\frac{1}{18}\right)^d \mathrm{d}((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)).$$

Combining Theorem 7 and Lemma 9, we obtain the following result.

Theorem 10 *For simplicity of notation, set* $C(d) = \frac{c\pi^{d/2}}{\Gamma(d/2+1)}$*. Let* $\mathrm{Noise}(\eta)$ *be a simple noise model and set* $\omega = \min_{|\eta| \leq \beta A} \mathrm{Noise}(\eta)$*. If* $f(\boldsymbol{z}) \geq c > 0$ *for each* $\boldsymbol{z} \in B^d(1)$*, then*

$$R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}^*, \tau^*) \geq \frac{\omega^2 C(d)^2}{800 \times 18^{2d}} \mathrm{d}^2\left((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)\right).$$

In particular, if $f(\boldsymbol{z})$ is uniform on $B^d(1)$, then $C(d) = 1$ and thus

$$R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}^*, \tau^*) \geq \frac{\omega^2}{800 \times 18^{2d}} \mathrm{d}^2((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)).$$

Now, combining Theorems 5 and 10, we obtain the following result.

Theorem 11 *Let* $\mathrm{Noise}(\eta)$ *be a simple noise model and set* $\omega = \min_{|\eta| \leq \beta A} \mathrm{Noise}(\eta)$*. Assume* $f(\boldsymbol{z}) \geq c > 0$ *for each* $\boldsymbol{z} \in B^d(1)$*. For any given* $\varepsilon, \delta > 0$*, if* $N > N(\frac{\omega^2\varepsilon^2 C^2(d)}{800 \times 18^{2d}}, \delta)$*, then with probability at least* $1-\delta$*, any point* $(\hat{\boldsymbol{M}}, \hat{\tau})$ *minimizing* $R_N(\boldsymbol{M}, \tau)$ *satisfies*

$$\mathrm{d}((\hat{\boldsymbol{M}}, \hat{\tau}), (\boldsymbol{M}^*, \tau^*)) < \varepsilon.$$

We remark that we have not attempted to optimize the constants which appear in the sample complexity bound in Theorem 11.

4.3 Rank-Deficient Case

No work prior to the present paper has proved that we can recover the matrix $\boldsymbol{M}^*$ when it is not full rank, or bounds the effect of truncating the derived $\hat{\boldsymbol{M}}$ to a low-rank $\hat{\boldsymbol{M}}_k$. Theorem 11 indicates that for any given $\varepsilon > 0$, $\|\hat{\boldsymbol{M}} - \boldsymbol{M}^*\|_2 < \varepsilon$ and $|\hat{\tau} - \tau^*| < \varepsilon$ if N is large enough. This will guarantee that there will be a small eigenvalue of $\hat{\boldsymbol{M}}$ for every small eigenvalue of $\boldsymbol{M}^*$.

Lemma 12 *Let $\boldsymbol{M}_1, \boldsymbol{M}_2$ be two given $n \times d$ matrices. If $\|\boldsymbol{M}_1 - \boldsymbol{M}_2\|_2 < \varepsilon$, then $|\sigma_i(\boldsymbol{M}_1) - \sigma_i(\boldsymbol{M}_2)| < \varepsilon$, where $\sigma_i(\boldsymbol{M}_j)$ is the i-th singular value of matrix $\boldsymbol{M}_j$ for $j = 1, 2$.*

Proof Set $\boldsymbol{M}_r = \boldsymbol{M}_1 - \boldsymbol{M}_2$. Because of the definition of the spectral norm,

$$\max_{\boldsymbol{x} \neq \boldsymbol{0}} \frac{\|\boldsymbol{M}_r \boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} < \varepsilon.$$

Write $\boldsymbol{M}_2 = \boldsymbol{M}_1 - \boldsymbol{M}_r$ and for each $i \in [d]$, notice

$$\begin{aligned}
\sigma_i(\boldsymbol{M}_2) &= \min_{\dim(W)=n-i+1} \max_{\substack{x \in W \\ \|\boldsymbol{x}\|_2=1}} \|(\boldsymbol{M}_1 - \boldsymbol{M}_r)\boldsymbol{x}\|_2 \\
&\leq \min_{\dim(W)=n-i+1} \max_{\substack{x \in W \\ \|\boldsymbol{x}\|_2=1}} (\|\boldsymbol{M}_1 \boldsymbol{x}\|_2 + \|\boldsymbol{M}_r \boldsymbol{x}\|_2) \\
&\leq \min_{\dim(W)=n-i+1} \max_{\substack{x \in W \\ \|\boldsymbol{x}\|_2=1}} (\|\boldsymbol{M}_1 \boldsymbol{x}\|_2 + \varepsilon) \\
&= \varepsilon + \min_{\dim(W)=n-i+1} \max_{\substack{x \in W \\ \|\boldsymbol{x}\|_2=1}} \|\boldsymbol{M}_1 \boldsymbol{x}\|_2 \\
&= \sigma_i(\boldsymbol{M}_1) + \varepsilon.
\end{aligned}$$

With a similar approach, we can prove that $\sigma_i(\boldsymbol{M}_1) \leq \sigma_i(\boldsymbol{M}_2) + \varepsilon$, which implies that

$$|\sigma_i(\boldsymbol{M}_1) - \sigma_i(\boldsymbol{M}_2)| < \varepsilon \qquad \forall i \in [d],$$

which completes the proof. ■

Using this lemma, we obtain the following lemma

Lemma 13 *Let $\alpha_1 \geq \alpha_2 > 0$ be given and assume the ground truth $\boldsymbol{M}^*$ has k eigenvalues greater than or equal to α_1 and k' eigenvalues less than or equal to $\alpha_2 - 3\varepsilon$ with $k + k' \leq d$. If the noise model $\mathrm{Noise}(\eta)$ is simple and $N > N(\frac{\omega^2 \varepsilon^2 C^2(d)}{800 \times 18^{2d}}, \delta)$, then, with probability at least $1 - \delta$, $\|\hat{\boldsymbol{M}} - \boldsymbol{M}^*\|_2 < \varepsilon$, the number of eigenvalues of $\hat{\boldsymbol{M}}$ which are greater than $\alpha_1 - \varepsilon$ is at least k, and the number of eigenvalues of $\hat{\boldsymbol{M}}$ which are less than $\alpha_2 - 2\varepsilon$ is at least k'.*

Proof Assume that $(\hat{\boldsymbol{M}}, \hat{\tau})$ minimizes $R_N(\boldsymbol{M}, \tau)$. By Theorem 11, with probability at least $1 - \delta$, we have $\|\hat{\boldsymbol{M}} - \boldsymbol{M}^*\|_2 < \varepsilon$. Now, Lemma 12 implies the result. ■

The next theorem gives us a better understanding of the eigenvalue perturbation of $\hat{\boldsymbol{M}}$ when $\boldsymbol{M}^*$ is not full rank.

Theorem 14 *Assume that the noise model* $\mathrm{Noise}(\eta)$ *is simple and* $\boldsymbol{M}^*$ *has rank* $0 < r < d$*. For a given* $\varepsilon, \delta > 0$*, if* $0 < 3\varepsilon < \sigma_r(\boldsymbol{M}^*)$ *and* $N > N(\frac{\omega^2\varepsilon^2 C^2(d)}{800\times 18^{2d}}, \delta)$*, then, with probability at least* $1-\delta$*,* $\hat{\boldsymbol{M}}$ *has exactly* $d-r$ *eigenvalues less than* ε *and the rest* r *eigenvalues are at least* $\frac{2}{3}\sigma_r(\boldsymbol{M}^*)$*. So if we truncate the eigenvalues of* $\hat{\boldsymbol{M}}$ *which are less than* ε *to zero, we obtain* $\hat{\boldsymbol{M}}_k$ *of rank* r *for which* $\|\boldsymbol{M}^* - \hat{\boldsymbol{M}}_r\|_2 < 2\varepsilon$*.*

Proof As Lemma 13 holds with $\alpha_1 = \sigma_r(\boldsymbol{M}^*), \alpha_2 = 3\varepsilon, k = r, k' = n - r$, we have $\|\hat{\boldsymbol{M}} - \boldsymbol{M}^*\|_2 < \varepsilon$, the number of eigenvalues of $\hat{\boldsymbol{M}}$ which are greater than $\alpha_1 - \varepsilon > \frac{2}{3}\sigma_r(\boldsymbol{M}^*)$ is r, and the number of eigenvalues of $\hat{\boldsymbol{M}}$ which are less than $\alpha_2 - 2\varepsilon = \varepsilon$ is $n - r$. Now,

$$\|\boldsymbol{M}^* - \hat{\boldsymbol{M}}_r\|_2 \leq \|\boldsymbol{M}^* - \hat{\boldsymbol{M}}\|_2 + \|\hat{\boldsymbol{M}} - \hat{\boldsymbol{M}}_r\|_2 \leq 2\varepsilon$$

completes the proof. ■

As we are not given $\boldsymbol{M}^*$, in practice we are unaware of its rank or spectral properties. Since we only have access to $\hat{\boldsymbol{M}}$, the next theorem establishes that the loss function still converges under eigenvalue truncation when the corresponding eigenvalues of $\hat{\boldsymbol{M}}$ are small.

Theorem 15 *Assume that the noise model* $\mathrm{Noise}(\eta)$ *is simple. For a given* $\varepsilon, \delta > 0$*, assume that* $(\hat{\boldsymbol{M}}, \hat{\tau})$ *minimizes* $R_N(\boldsymbol{M}, \tau)$ *for* $N > N(\varepsilon/2, \delta)$*. Define threshold* γ *such that* $\hat{\boldsymbol{M}}$ *has* $d - k$ *eigenvalues which are less than* γ*. Let* $\hat{\boldsymbol{M}}_k$ *be the rank* k *matrix obtained from* $\hat{\boldsymbol{M}}$ *by setting these* $d - k$ *eigenvalues to zero. Then, with probability at least* $1 - \delta$*,*

$$0 < R(\hat{\boldsymbol{M}}_k, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*) < \gamma + \varepsilon.$$

Proof As $\|\hat{\boldsymbol{M}} - \hat{\boldsymbol{M}}_k\|_2 < \gamma$, using the proof of the first inequality in Lemma 3, we obtain

$$|R(\hat{\boldsymbol{M}}, \hat{\tau}) - R(\hat{\boldsymbol{M}}_k, \hat{\tau})| < \zeta F\gamma = \gamma,$$

since we have assumed $\zeta F = 1$. On the other hand, by Theorem 5, we have

$$0 < R(\hat{\boldsymbol{M}}, \hat{\tau}) - R(\boldsymbol{M}^*, \tau^*) < \varepsilon.$$

Combining these two inequalities implies the desired inequality. ■

Combining Theorems 10 and 15, we also have $\mathrm{d}\left((\hat{\boldsymbol{M}}_k, \hat{\tau}), (\boldsymbol{M}^*, \tau^*)\right) < \frac{20\sqrt{2}\times 18^d}{\omega C(d)}(\gamma + \varepsilon)$. Thus, if $\gamma \leq \varepsilon\frac{\omega C(d)}{20\sqrt{2}\times 18^d}$, then $\mathrm{d}\left((\hat{\boldsymbol{M}}_k, \hat{\tau}), (\boldsymbol{M}^*, \tau^*)\right) < 2\varepsilon$, implying $\|\hat{\boldsymbol{M}}_k - \boldsymbol{M}^*\|_2 < 2\varepsilon$. Thus spectral truncation of the empirical $\hat{\boldsymbol{M}}$ yields a good approximation of $\boldsymbol{M}^*$.

4.4 Invariance to Changes in Unit of Input

Clearly, the learned $\hat{\boldsymbol{M}}$ and $\hat{\tau}$ are dependent on the units of feature space. So, as an interesting question, we can study the behavior of $\hat{\boldsymbol{M}}$ and $\hat{\tau}$ if we change the units in the original feature space. Mainly, we want to prove that if we change the units in feature space, we do not need to solve a new optimization problem to learn a new $\hat{\boldsymbol{M}}$ and $\hat{\tau}$. Instead, we can recover these parameters from the already solved optimization problem. Assume that we have a non-singular matrix $\boldsymbol{U}_{d\times d}$ which

changes the units and rotates the space of features, and let $\boldsymbol{z}'_i = \boldsymbol{U}\boldsymbol{z}_i$ and $\ell'_i = \ell_i$. We want to solve the following optimization problem

$$\min_{\boldsymbol{M}\succeq 0, \tau \geq 0} R'_N(\boldsymbol{M}, \tau)\,, \tag{12}$$

where

$$\begin{aligned} R'_N(\boldsymbol{M}, \tau) &= -\frac{1}{N}\sum_{i=1}^{N} \log\sigma\left(\ell'_i(\|\boldsymbol{z}'_i\|^2_{\boldsymbol{M}} - \tau)\right) \\ &= -\frac{1}{N}\sum_{i=1}^{N} \log\sigma\left(\ell_i(\|\boldsymbol{z}_i\|^2_{\boldsymbol{U}^\top \boldsymbol{M}\boldsymbol{U}} - \tau)\right) \end{aligned}$$

Since $\boldsymbol{U}$ is non-singular, Optimization Problem 5 is minimized at $\hat{\boldsymbol{M}}, \hat{\tau}$ if and only if Optimization Problem 12 is minimized at $\hat{\boldsymbol{M}}' = {\boldsymbol{U}^{-1}}^\top \hat{\boldsymbol{M}}\boldsymbol{U}^{-1}, \hat{\tau}' = \hat{\tau}$. Hence, the solution is invariant to the choice of units, given knowledge of the conversion.

5. Experimental Results

In Section 3, we described the optimal loss functions for four different noise distributions (Logistic, Normal, Laplace, and Hyperbolic Secant). As the Normal noise model ends up with a probit in the loss function, and the probit function has no closed-form formula, we will not use this model in the experiments. Also, since in practice we generally do not know the noise distribution, we evaluate the performance of each model under a variety of possible noise distributions, thus testing the robustness of each model to misspecification of the noise. We investigate how the resulting accuracy depends on the sample complexity, amount of noise, and noise misspecification.

We start with synthetic data, described in Section 5.1, so we can run precisely controlled experiments which are reported in Sections 5.2 and 5.3. In Section 5.4, we compare our model performance with DML-eig (Ying and Li (2012)). Then in Section 5.5 we apply our methods to some real data experiments well suited to our proposed algorithm. All experimental results are reproducible; see the GitHub repository by Alishahi et al. (2023) containing data and source codes.

5.1 Data Generation

We start with d random positive real values $\lambda_1, \ldots, \lambda_d$ and then we randomly generate a $d \times d$ covariance matrix Σ whose eigenvalues are $\lambda_1, \ldots, \lambda_d$. To this end, we first randomly and uniformly generate an orthonormal matrix $\boldsymbol{U}_{d\times d}$ and then set $\Sigma = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$, where $\boldsymbol{D}$ is the $d \times d$ diagonal matrix whose diagonal entries are $\lambda_1, \ldots, \lambda_d$. We then independently sample $2N$ points $\boldsymbol{x}_1, \boldsymbol{y}_1, \ldots, \boldsymbol{x}_N, \boldsymbol{y}_N$ from $\mathcal{N}(\boldsymbol{0}, \Sigma)$ to generate N pairs $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ for $i = 1 \ldots, N$. Next we select d nonnegative random real values $\gamma_1, \gamma_2, \ldots, \gamma_d \geq 0$ as the eigenvalues of the ground truth $\boldsymbol{M}^*$, and randomly generate $\boldsymbol{M}^*$ to be a random positive semi-definite matrix with eigenvalues $\gamma_1, \ldots, \gamma_d$, as we did for Σ. We have

$$\mathbb{E}\left(\|\boldsymbol{x} - \boldsymbol{y}\|^2_{M^*}\right) = 2\mathrm{tr}\left(\Sigma M^*\right)$$

provided that $\boldsymbol{x}, \boldsymbol{y} \sim h(\boldsymbol{x})$, where $h(\boldsymbol{x})$ is a pdf for which $\mathrm{Cov}_{X\sim h}(X) = \Sigma$. We now choose $\tau^* > 0$ not far from $\mathbb{E}\left(\|\boldsymbol{x} - \boldsymbol{y}\|^2_{M^*}\right)$ so that we obtain a sufficient number of pairs $(\boldsymbol{x}, \boldsymbol{y})$ labeled as both Close and Far. More specifically, in Sections 5.2-5.4, we consider the following setting.

- We assume that the rank of $\boldsymbol{M}^*_{10\times 10}$ is 5 and randomly and uniformly generate 5 nonzero eigenvalues from $[0,1]$. With two-digit precision, we obtain $0.32, 0.89, 0.59, 0.13, 0.14$ as the 5 nonzero eigenvalues of $\boldsymbol{M}^*$.

- We randomly and uniformly select 10 nonzero numbers from $(0,1]$ as the eigenvalues of Σ. With two digit precision, we obtain

$$0.73,\ 0.7,\ 0.68,\ 0.59,\ 0.47,\ 0.45,\ 0.21,\ 0.19,\ 0.11,\ 0.04$$

  as the eigenvalues of Σ.

- As we are dealing with fixed random seeds, we obtain $\mathbb{E}\left(\|\boldsymbol{x}-\boldsymbol{y}\|^2_{M^*}\right) \approx 1.7$.

- To obtain roughly balanced data, we set $\tau^* = 1.3$ and generate 20000 data points. We split the data into 15000 training and 5000 test points.

We now describe the label generation. Note that in the theoretical formulation of the problem, we assume that the noisy labeling process depends on $\|x-y\|^2_{\boldsymbol{M}^*}$. However one could also assume that the noise changes labels directly, independently of $\|x-y\|^2_{\boldsymbol{M}^*}$. We thus study both of the following settings empirically.

- **Noise affects the labeling through $\|x-y\|^2_{\boldsymbol{M}^*}$ (Label Assumption).**
  We consider a noise distribution $\mathrm{Noise}(0,s)$ with zero mean and scale parameter s from the Logistic, Gaussian, Laplace, and Hyperbolic Secant distributions. We then generate $\eta_1,\dots,\eta_N \sim \mathrm{Noise}(0,s)$. For each pair $(\boldsymbol{x}_i,\boldsymbol{y}_i)$, we set $\ell_i = 1$ ("Far") if $\|\boldsymbol{x}_i-\boldsymbol{y}_i\|^2_{\boldsymbol{M}^*}+\eta_i \geq \tau^*$, and we set $\ell_i = -1$ ("Close") if $\|\boldsymbol{x}_i-\boldsymbol{y}_i\|^2_{\boldsymbol{M}^*}+\eta_i < \tau^*$. We save these labels as D_{noisy}. We also save the non-noisy labels to check the model's robustness against noise. However, we do not use these labels during training. Indeed, for each pair $(\boldsymbol{x}_i,\boldsymbol{y}_i)$, we set $\ell^*_i = 1$ if $\|\boldsymbol{x}_i-\boldsymbol{y}_i\|^2_{\boldsymbol{M}^*} \geq \tau^*$ and we set $\ell^*_i = -1$ if $\|\boldsymbol{x}_i-\boldsymbol{y}_i\|^2_{\boldsymbol{M}^*} < \tau^*$. We save these labels as D^*.

- **Noise directly affects the labeling (Noisy Labeling).**
  Here we assume that the noise affects the labels directly by randomly flipping them. We first generate D^* as described in the previous paragraph. Then for each $i=1,\dots,N$, we flip a coin whose head chance is p. If the coin is tails, we set $\ell_i = \ell^*_i$; otherwise, we set $\ell_i \in \{-1,1\}$ randomly with the same chance. We save these labels as D_{noisy}. In expectation, $p/2$ fraction of the labels are mislabeled in D_{noisy}. Although the amount of noise is the same as in the previous setting, i.e. the same number of mistakes are made, this regime is more challenging because in the first case the majority of mistakes occur close to the boundary, while the noisy labeling case results in "big" mistakes. We thus expect performance to be worse.

As a default, in both settings we set the noise parameter so that 10% of the points are mislabeled.

5.2 Logistic Model with Different Noises

Recall the Logistic distribution has density function

$$\mathrm{L}(x|\mu,s) = \frac{1}{4s}\mathrm{sech}^2\left(\frac{x-\mu}{2s}\right).$$

In Subsection 3.1, we saw that if the noise comes from a Logistic distribution, then $R_N(\boldsymbol{M}, \tau) = -\frac{1}{N}\sum_{i=1}^{N} \log \sigma\left(\ell_i(\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2 - \tau)\right)$ serves as an optimal proxy for our objective. In this section, we generate labels with different noise types including noisy labeling. We set the corresponding noise parameter so that the number of mistakes is roughly 10%, and then investigate how the Logistic loss function performs on all these types of noise.

We solve Optimization Problem (10) using gradient descent and setting $learning\_reate = 0.5$, $d = k = 10$, $number\_of\_iterations = 30000$, and $learning\_decay = .95$. We did this 20 times for independent sample observations and summarized the average accuracy in Table 2. Note that the model uses only the noisy labels during training; the non-noisy labels are only used to evaluate the model.

Noise type:	Logistic	Gaussian	Laplace	HS	Noisy Labeling
train acc. w/ noise	89.93% (0.22)	89.51% (0.20)	87.35% (0.28)	85.48% (0.30)	85.73% (0.34)
train acc. w/o noise	98.80% (0.10)	98.79% (0.19)	98.61% (0.13)	98.53% (0.14)	94.68% (0.32)
test acc. w/ noise	89.76% (0.40)	89.34% (0.32)	87.27% (0.51)	85.28% (0.47)	85.57% (0.65)
test acc. w/o noise	98.83% (0.21)	98.82% (0.18)	98.52% (0.21)	98.47% (0.23)	94.51% (0.60)

Table 2: Logistic model average accuracy (std) with different noise types (average over 20 trails).

The Logistic model learned the labeling function very well. With Logistic noise (first column), it reaches about 90% accuracy on noisy labels (as high as possible with 10% misclassification), and almost 99% accuracy with respect to the ground truth labels. This holds on both the training and test data sets, which indicates that the model is not overfitting. We also observe that as the noise becomes more and more different from the Logistic model (Gaussian then Laplace then HS then Noisy Labeling), the accuracy gets worse. This holds for both the noisy and ground truth labels, and on both the training and test data sets. The deterioration is most prominent in the "noisy labeling" setting, where about 5% accuracy is lost in comparison with the Logistic noise.

Next, supporting Theorem 11, we summarize the recovery of the model parameters $\boldsymbol{M}^*/\tau^*$ in Table 3; we report the average recovery error based on 20 independent sample observations. We observe that the error is fairly small with a relative error of about 0.07 for most noise types, but also that the error increases as the misspecification of the noise type increases. For instance noisy labeling achieves only about 0.2-relative error in a Frobenius or spectral sense.

Noise type	Logistic	Gaussian	Laplace	HS	Noisy Labeling
$\left\|\frac{\hat{M}}{\hat{\tau}} - \frac{M^*}{\tau^*}\right\|_2 / \left\|\frac{M^*}{\tau^*}\right\|_2$	0.068 (0.030)	0.069 (0.023)	0.074 (0.030)	0.086 (0.034)	0.231 (0.016)
$\|\frac{\hat{M}}{\hat{\tau}} - \frac{M^*}{\tau^*}\|_F / \|\frac{M^*}{\tau^*}\|_F$	0.070 (0.020)	0.071 (0.015)	0.080 (0.019)	0.088 (0.022)	0.214 (0.019)

Table 3: Average Precisions (std) for Logistic model with different noise types (averaged over 20 trials).

We plot the eigenvalues of $\boldsymbol{M}^*/\tau^*$ and $\hat{\boldsymbol{M}}/\hat{\tau}$ in Figure 2. The large left figure shows the eigenvalue recovery by the Logistic, Laplace, and HS loss functions when the labels are generated from Logistic noise. All do about the same, and capture all eigenvalues fairly well. Four other plots are shown with other types of noise with similar results; the main exception being with noisy labeling, the top eigenvalue is predicted as much smaller than the true value. This experiment

illustrates that although we focus on the Logistic loss function, performance is robust with respect to misspecification of the noise model.

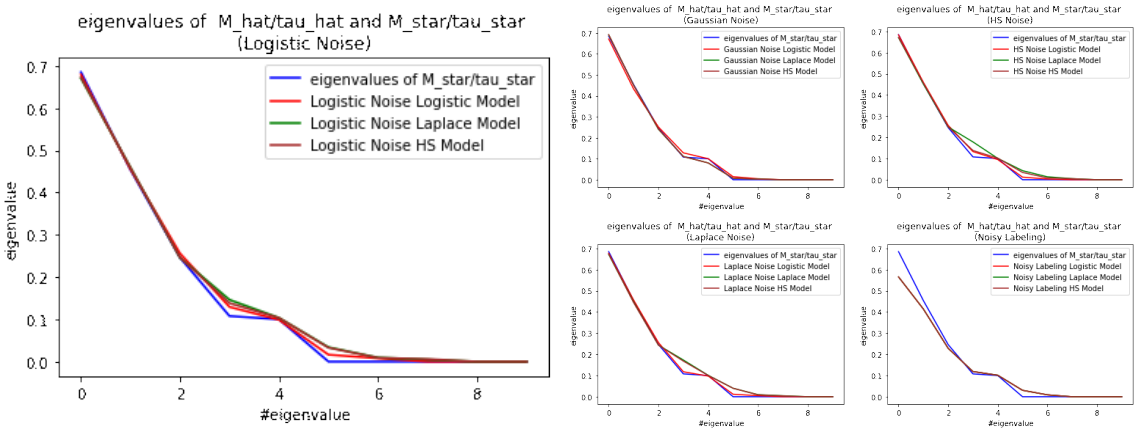

Figure 2: Comparing eigenvalues of $\hat{\boldsymbol{M}}/\hat{\tau}$ and $\boldsymbol{M}^*/\tau^*$.

In Figure 3, we summarize the accuracy of the Logistic model for different noises as the number of iterations increases; the Logistic noise plot is highlighted on the left. Each plot shows the progression of training on the train and test accuracy. As before, there is little difference between test and train accuracy. The accuracy with the noise-induced labels plateaus near 90% which is as good as expected with 10% noise. And the accuracy on the ground truth labels continues to increase (to about 99%) as training continues. The results are similar for other noise types, with convergence to lower plateaus, as expected.

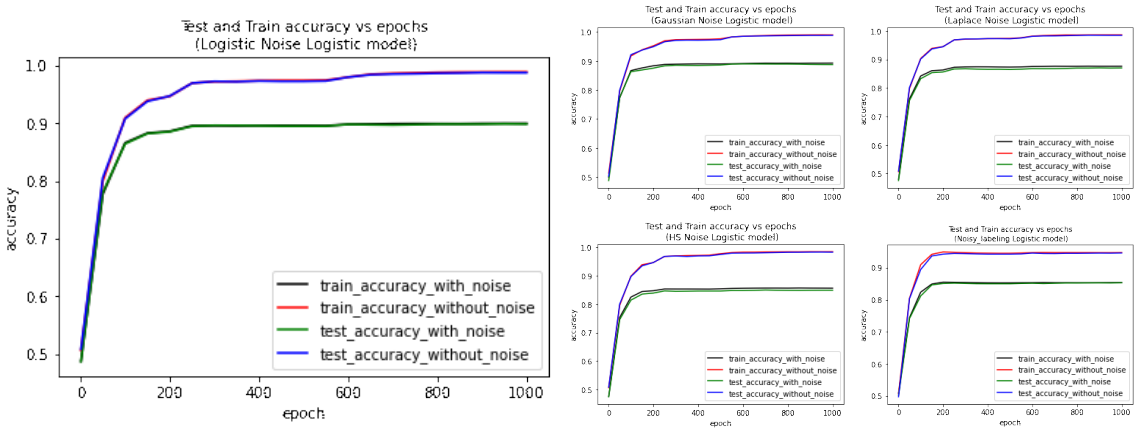

Figure 3: Logistic model's accuracy VS epochs: Logistic noise, Gaussian noise, Laplace noise, HS noise, and Noisy labeling.

To study the sample complexity behavior, we gradually increase the number of training samples and record the accuracy for each case in Figure 4. Again the left figure illustrates the Logistic loss on labels generated with Logistic noise, plotting results for training and test data, with respect to

the noisy and ground truth labels. Note that when the number of training points is too small (100 or less), the training accuracy is 1 while the test accuracy is low; this indicates overfitting. However, when the number of training points increases to around 1000, the overfitting problem vanishes, and the training and test accuracy start to align closely. Between 5000 and 10,000 samples, they become indistinguishable in the plots. Similar results hold for the other noise types, again with somewhat lower overall accuracy depending on how close the noise model is to the Logistic noise.

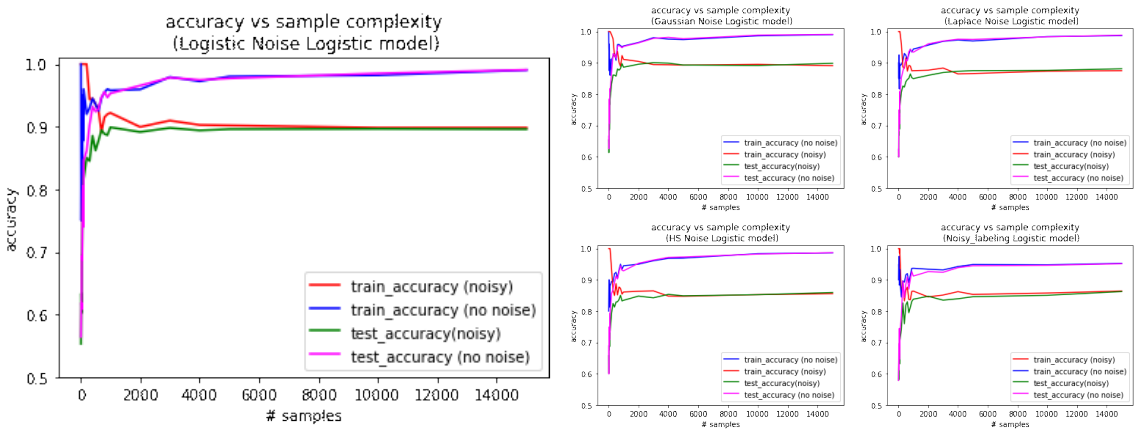

Figure 4: Logistic model with different noises: accuracy VS sample complexity.

5.3 Further Ablation Study

In the following two parts, we experimentally explore the robustness and the sample complexity of the Logistic model when faced with high Logistic noise.

5.3.1 How Much Noise can Break the Model

Theoretically, we proved that the model can recover the ground truth parameters even if the labeling is noisy (under some assumptions). This result is also supported by the experimental evidence in the previous section when the noise causes 10% mislabeling. In this section we increase the effect of the noise and check the model resistance. For a fixed number of training samples (18000 here), we increase the noise variance gradually and log the accuracy of the Logistic loss function when the noise also comes from a Logistic distribution. In Figure 5, the x axis shows the fraction of points that are mislabeled, which depends directly on the variance of the noise distribution, and the y axis indicates accuracy. We generally observe that the model ignores the noise and recovers true labeling even when the noise is high. We can see that the train and test accuracies of the model for the noisy labels are aligned with the line $y = 1 - x$, which is as expected. It indeed indicates that with x amount of noise, the model cannot have better accuracy than $1 - x$ on the noisy labels. However, for the ground truth labeling (described in the legend as "no noise"), we observe that the model is pretty robust against noise, even when the amount of noise is pretty high. For instance, for around 40% mislabeling, we have around 95% of accuracy for unseen data. However, when the noise perturbs 45% of the labels, it starts to collapse. When the noise disturbs 50% of the labels, we might assume that random guessing would achieve the best accuracy, but the model still achieves around 65% accuracy for train and test points with respect to the ground truth labels. Even though

we have 50% mislabeling, the "extreme" examples are correct, so there is more information than purely random labels. We will study this setting in the next paragraph.

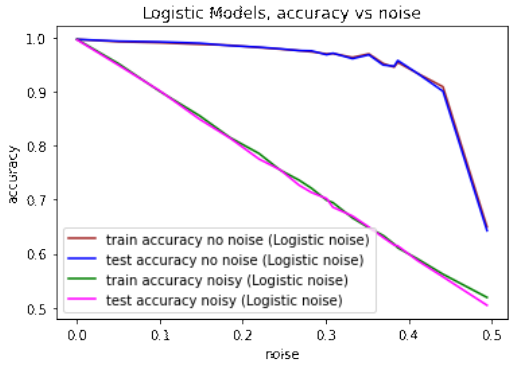

Figure 5: Logistic model with Logistic noises: Accuracy VS Noise.

5.3.2 **Sample Complexity in High Noise Setting**

Now we focus on the setting where the loss function and the noise are compatible. In other words, we only consider the Logistic model for Logistic noise, the Laplace model for Laplace noise, and the HS model for HS noise. As explained in Section 2, the scale parameter s for the noise distribution $\mathrm{Noise}(\eta|0,s)$ directly determines the portion of mislabeling imposed on the data. In Figure 5, we observe that the accuracy drops when the noise gets more intense. However, in theory, we proved that each model could overcome any amount of noise perturbation. The noise scale parameter (variance) affects the sample complexity through the constants β and B (see Section 2: Model Assumption and the discussion after). In Theorems 5 and 7, we proved that irrespective of the amount of noise, we can recover the ground truth parameters if the number of samples is sufficiently large. However in Figure 5, we saw that if the Logistic noise changes around 50% of the labels, then the test accuracy drops to 65% when we have 15000 samples in the training set. Supported by the theoretical results, we should expect more and more accuracy if we increase the number of training points. To verify this, in this experiment we fix the amount of noise at 45% and gradually increase the number of training points to 2×10^5 samples; Figure 6 reports the resulting accuracy. We observe that model accuracy with respect to the ground truth labels is approaching one. With 2×10^5 training samples, we have around 97% accuracy on the test data. This observation adheres to our theoretical results about the recovery power of the method.

For further experiments about the behavior of the loss function and a higher dimensional example, see Appendix G.

5.4 Comparing to DML-eig

Inspired by a work of Xing et al. (2002), Ying and Li (2012) developed an eigenvalue optimization framework (called DML-eig) for learning a Mahalanobis metric. They define an acceptable optimization problem and elegantly reduce it to minimizing the maximal eigenvalue of a symmetric matrix problem (Overton, 1988; Lewis and Overton, 1996). In their formulation, given pairs of similar data points and pairs of dissimilar data points, the goal is to learn a Mahalanobis metric which

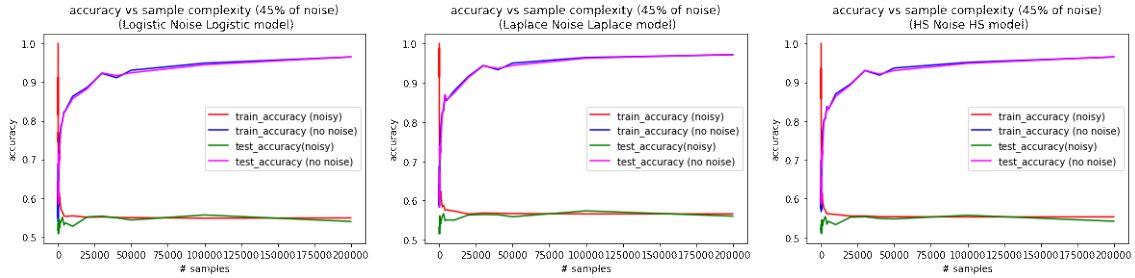

Figure 6: Accuracy VS Sample complexity with 45% noise when the model is aligned with the noise type.

preserves similarity and dissimilarity. More specific, they look for a p.s.d. matrix $\boldsymbol{M}^*$ to maximize the minimal squared distances between dissimilar pairs while the sum of squared distances between similar pairs is at most 1. This setting is comparable to ours since we also look for a matrix $\boldsymbol{M}^*$ (and also a threshold τ^*) to distinguish labeled far pairs of points from labeled close pairs of points. Their work did not study how noise can affect their model, nor if it could potentially recover a "ground truth" model that generates the dissimilar and similar labels. However, we can compare our model to theirs empirically by passing our similarities and dissimilarities to their model and checking whether their model can handle noise or recover ground truth parameters. We can also use the $\hat{\boldsymbol{M}}_{\text{eig}}$ learned by their model with the best $\hat{\tau}$ to see how well they can predict the labels.

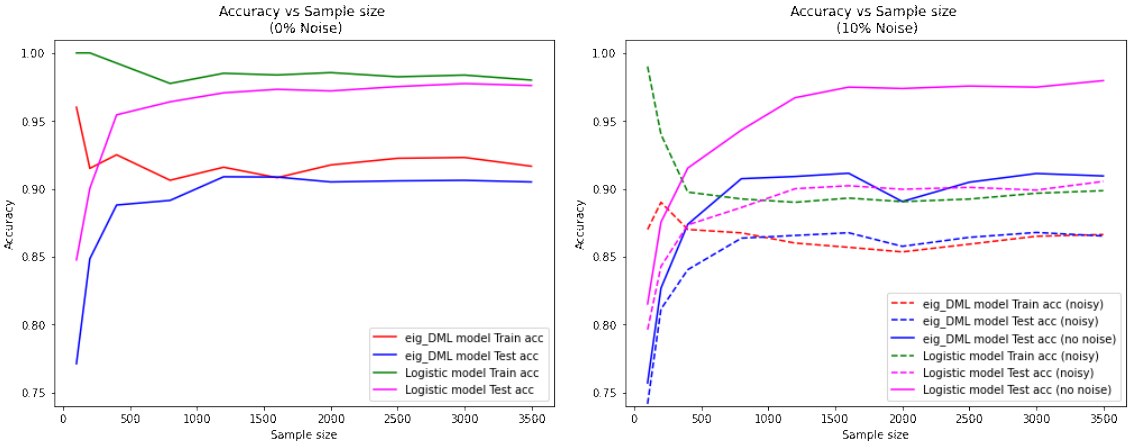

Figure 7: Performance of DML-eig with/without noise vs sample complexity.

In Figure 7, we compare our model with DML-eig using the data generated in Section 5.1. We set the noise level at 0% or at 10% and use the Logistic noise distribution. The number of training points are indicated as the sample size and the test size is always fixed at 5000 points. For the noisy data, we evaluate performance with respect to both the noisy and ground truth labels (described as "no noise"). We can see that our model outperforms DML-eig in each setting and for any sample size. Although the DML-eig model can neutralize the noise (the blue curves in the left and right images are about the same), its accuracy in the noiseless setting is only around 90% at best. In

comparison, our model quickly overcomes the noise and its accuracy approaches 100% (shown as the magenta curves). In the noisy setting, we train on the noisy labels training data, and show results for both techniques. Then we compare against the test data with respect to both the noisy (dashed curves) and ground truth labels, and report the results. Our approach can recover the parameters even under noise (the magenta curve), and the noisy test data matches the noisy training data (so no over fitting). On the other hand, DML-eig only achieves about 90% test accuracy with respect to the ground truth labels, and about 85% train and test accuracy with respect to the noisy labels.

Next we observe that the DML-eig model is far less scalable than our approach. This is because it takes several matrix multiplications and eigenvalue solves for each subgradient step. In Figure 8, we compare its training time to our model. While our model always takes less than a second on a sample size up to 3500, we see that DML-eig quickly surpasses the 20 minute mark (1200 seconds) and starts to become intractable.

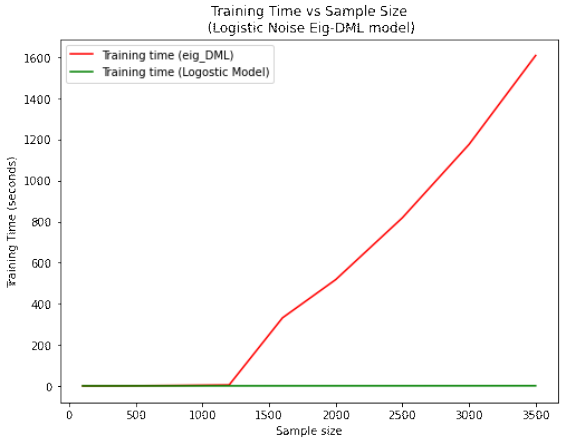

Figure 8: Training Time Complexity comparison between our Logistic and DML-eig.

If we provide our model with $10{,}000$ training points, after 17 seconds, it can reach a test accuracy of 90% with respect to the noisy labels and 99% with respect to the ground truth labels, while DML-eig after 3 hours and 45 minutes cannot do better than 85% and 90% respectively.

5.5 Real Data Experiments

In this section we use our methods to solve some real data challenges which demand or benefit from a learned Mahalanobis distance. The first one is from a physical simulation where we aim to find a reduced order model that needs to pass through a linear projection. Thus the learned scaling must be linear, and is desired to be low rank. The second one is a consumer satisfaction story, and instead of using pairs of data points $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{y}$, it directly uses data points $\boldsymbol{z}$ as the input, where satisfaction is predicted as a Mahalanobis distance from the origin. In both cases we show our method achieves the objectives with high accuracy.

5.5.1 Finding a Low-Rank Metric for Equations of State Combustion Simulation

We first consider a data set generated by Hansen et al. (2020) to represent instances of the equations of state of a thermo-chemical reacting system. The goal is to model a combustion process to

produce more efficient fuels and for easier CO2 capture and sequestration. The data set consists of $30{,}000$ data points in $\mathbb{R}^9$, with 8 dimensions capturing *equations of state*, that is, the fractional composition of various chemicals (like O2, H, CO2) present in the system, and the 9th coordinate being temperature (Zdybał et al., 2022). The goal is to learn a reduced order model (ROM) which should linearly project and scale (Sutherland and Parente, 2009) to a lower dimension, so one can then learn a PDE modeling the physics. The PDE modeling on the ROM is only tractable through a linear transformation (Sutherland and Parente, 2009), so non-linear approaches are not permitted. And choosing an appropriate scaling is crucial since there is a difference of several orders of magnitude in coordinate ranges.

We consider two ways of generating labels to apply our methodology. The first is based on the best known engineered solution (Zdybał et al., 2022) which we attempt to replicate from the training part of a test/train split. The second is via how close data points are on a critical simulation value called "mixture fraction."

For the best known engineered solution, we start with a provided "ground truth" feature transformation matrix $\boldsymbol{A}^* \in \mathbb{R}^{9\times 3}$ as found by Zdybał et al. (2022). We set $\boldsymbol{M}^* = \boldsymbol{A}^*(\boldsymbol{A}^*)^\top$. Given the original and the projected data, we choose a threshold and label the disjoint pairs of the original data points as far and close based on their projected distance and the threshold. Now, we have $15{,}000$ pairs of original points, and we divide them into a train set of size $10{,}000$ and a test set of size $5{,}000$. We only use the train pairs of data points and their labels to recover $\boldsymbol{M}^*$. Recovering the labels, we obtain 99.57% and 99.51% accuracy on the train and test points, respectively. We have summarized the results in Tables 4 and 5.

	Raw data	Normalized data	Covariance normalization
Mean training accuracy	–	99.58% (0.07)	99.82% (0.04)
Mean test accuracy	–	99.49% (0.10)	99.71% (0.09)
$\left\|\frac{\hat{\boldsymbol{M}}}{\hat{\tau}} - \frac{\boldsymbol{M}^*}{\tau^*}\right\|_2 / \left\|\frac{\boldsymbol{M}^*}{\tau^*}\right\|_2$	–	0.744 (0.013)	0.048 (0.019)
$\left\|\frac{\hat{\boldsymbol{M}}}{\hat{\tau}} - \frac{\boldsymbol{M}^*}{\tau^*}\right\|_F / \left\|\frac{\boldsymbol{M}^*}{\tau^*}\right\|_F$	–	0.736 (0.014)	0.053 (0.020)

Table 4: Mean accuracies (std) and precisions (std) for recovery using $\hat{\boldsymbol{M}}$ (average over 20 trails).

Normalizing data. The method does not converge when we input the raw data as provided to our solver; this is due to the different scaling of coordinates. The magnitude of some coordinates of the data are huge (temperature) and some are very small (CO2 percentage). So in the gradient descent, the learning rate is the same for all variables; the algorithm does not show convergence on the variable with very small values, even after a large number of steps. We can solve this problem by doing coordinate-wise normalization of the data, thus scaling all coordinates similarly, and this process is labeled *Normalized data* in Table 4. However, even in this setting, we do not have a good parameter recovery; see the last two rows of the second column of Table 4 with about 0.75 relative error in $\hat{\boldsymbol{M}}/\hat{\tau}$. To analyze this situation, we compute the estimated covariance matrix from the data, we observe that the variance of the data in some directions is almost zero. Note that these directions are not along coordinates, they are a linear combination of the coordinates, so coordinate-wise normalization does not correct for it. Note that in the theoretical results, we have the parameter

recovery only if the support of data distribution has a nonzero Lebesgue measure which is effectively not the case here.

To address the remaining issue, note that if we change the behavior of $\boldsymbol{A}^* : \mathbb{R}^d \longrightarrow \mathbb{R}^k$ only for those directions that the data variance is zero, then the distance in the projected space remains almost always the same and thus the labeling remains the same as well. This implies that there is no unique $\boldsymbol{M}^*$ to recover. However, if we rescale the data and $\boldsymbol{A}^*$ by $\sqrt{C}$ and the data by $(\sqrt{C})^{-1}$, where C is the covariance matrix of the data, we correct for this issue. Indeed, if we set $\boldsymbol{A}^*_{new} = \sqrt{C}\boldsymbol{A}^*$ and $X_{new} = X_{normal}(\sqrt{C})^{-1}$, then, in this setting, $\boldsymbol{M}^*_{new} = \sqrt{C}\boldsymbol{A}^*(\boldsymbol{A}^*)^\top\sqrt{C}$ has a negligible effect in the directions where C has a small variance. As the gradient descent initiates $\boldsymbol{M}$ with very small entries and it will receive no substantial update in those directions, finally we recover $\boldsymbol{M}^*_{new}$ very well, see the third column of Table 4 labeled *Covariance normalization*.

Low-rank recovery. Moreover, we can truncate the matrix $\hat{\boldsymbol{M}}$ to a rank-k matrix $\hat{\boldsymbol{M}}_k$ by setting the last $d-k$ eigenvalues of $\hat{\boldsymbol{M}}$ to zero. In Table 5, we summarize the average accuracies (over 20 independent sample observations) for the case that we use $\hat{\boldsymbol{M}}_k$ instead of $\hat{\boldsymbol{M}}$ for $k = 1, 2, 3, 4$. We can see that the for $k = 1$, we still have 90% accuracy, for $k = 2$ we have 98% accuracy. For $k = 3$, the $\hat{\boldsymbol{M}}_k$ and $\hat{\boldsymbol{M}}$ are indistinguishable. Hence, we can recover the best Mahalanobis distance $\boldsymbol{M}^*$ up to very high classification accuracy and in parameters, even with a desired low-rank solution. We preprocess data here using Covariance normalization.

	$k=1$	$k=2$	$k=3$	$k=4$
Mean training accuracy	89.83% (0.49)	97.83% (0.11)	99.80% (0.04)	99.82% (0.04)
Mean test accuracy	89.93% (0.57)	97.78% (0.18)	99.71% (0.09)	99.71% (0.09)
$\left\\| \frac{\hat{\boldsymbol{M}}_k}{\hat{\tau}} - \frac{\boldsymbol{M}^*}{\tau^*} \right\\|_2 / \left\\| \frac{\boldsymbol{M}^*}{\tau^*} \right\\|_2$	0.301 (0.002)	0.053 (0.012)	0.048 (0.019)	0.048 (0.019)
$\\| \frac{\hat{\boldsymbol{M}}_k}{\hat{\tau}} - \frac{\boldsymbol{M}^*}{\tau^*} \\|_F / \\| \frac{\boldsymbol{M}^*}{\tau^*} \\|_F$	0.294 (0.004)	0.068 (0.016)	0.053 (0.020)	0.053 (0.020)

Table 5: Mean accuracies (std) and precisions (std) for dimension reduction recovery using truncated $\hat{\boldsymbol{M}}_k$ for $k = 1, 2, 3, 4$ (average over 20 trails).

Mixture fraction labeling. One of the features in the data is called the mixture fraction, which takes values between 0 and 1. We remove this feature from the data set and use it to label the points as far and close. We first randomly extract $15{,}000$ disjoint pairs from the data. Then we compute the absolute of their mixture fraction difference, and based on an appropriate threshold, we assign the far and close label to the pairs. We choose a threshold τ such that the generated labels are balanced. We partition the data into $10{,}000$ training points and $5{,}000$ test points. Now, we try to see whether there is a matrix $\boldsymbol{M}^*$ and a threshold τ^* that can replicate this labeling. Indeed, we are able to find $\hat{\boldsymbol{M}}$ and $\hat{\tau}$ for which we have 99.68% accuracy for the test set, which is basically as good as our recovery of the best engineered solution from Zdybał et al. (2022). Notably, we again only have this accuracy if we normalize the data. If we work with the raw data directly, the best performance is about 70%. We summarize the corresponding results in Table 6.

5.5.2 Airline Passenger Satisfaction

We consider a data set containing a training set of around $100{,}000$ points and a test set of around $26{,}000$ points from the Airline Passenger Satisfaction (Air, 2020) data set. Each data point contains

	Raw data	Normalized data
Training accuracy	71.34%	99.83
Test accuracy	70.74%	99.78

Table 6: Accuracy for the mixture fraction labeling.

24 features; 20 are real-valued, and 4 are categorical features containing **Gender:** Female, Male, **Customer Type:** Loyal, disloyal, **Type of Travel:** Business, Personal, **Class:** Eco, Eco Plus, Business. As the first three are binary, we simply convert them to 0 and 1. The fourth one is also ordinal, and we convert it to 0, 1, and 2, respectively. The satisfaction of each passenger is given as either "Neutral or dissatisfied" or "Satisfied". We simply interpret "Neutral or dissatisfied" as 0 (Close) and "Satisfied" as 1 (Far). The train and test sets contain about 45% of satisfied passengers which means that the data is almost balanced. Here we assume that passenger satisfaction is determined by a Mahalanobis norm (distance from the origin). Indeed, we model the problem as if there is some matrix $\boldsymbol{M}^*$ and threshold τ^*, so for each data point $\boldsymbol{z}$, $\|\boldsymbol{z}\|^2_{\boldsymbol{M}^*} + \text{Noise}_{\boldsymbol{z}} \geq \tau^*$, for some unknown unbiased noise $\text{Noise}_{\boldsymbol{z}}$, if and only if the corresponding passenger is satisfied with the airline. We would like to find the generating $\boldsymbol{M}^*$ and τ^*. Compared to the theoretical setting, here we are given $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{y}$ or in other words, we can assume that $\boldsymbol{y} = \boldsymbol{0}$. Using each of the Logistic, Laplace, and HS models ($learning\_rate = 0.045$ *and* $number\_iterations = 20{,}000$), we can recover the satisfaction labeling with 93% accuracy on the training and test data. It is notable that, similar to previous observations, we obtain these accuracies only for normalized data. Other methods, such as random forest, can boost the labeling accuracy to 96% (see Air, 2020). However, our method gives us a feature transformation $\hat{A}$, which is more interpretable compared to something like the random forest. We can find the most important directions for $\hat{A}$ to see what combination of features has the greatest effect on the satisfaction level of passengers.

While this can be modeled as a standard classification task, we believe it is worthwhile to consider it as a linear distance metric learning one. For one, it shows that we can achieve near-state-of-the-art accuracy against classic classification approaches. But moreover, real satisfaction can be thought of as a continuous value; then we can use an underlying metric to determine this satisfaction value by measuring the distance (based on that metric) of a data point from the origin. There is also a threshold such that if the satisfaction value is greater, then the passenger is happy with the service. We here assumed that such a metric and threshold can be modeled via our linear distance metric learning approach. More-so, that threshold can vary among different persons, resulting in a noise labeling process which our formulation accounts for. Thus, the linear distance metric learning approach provides not just a classifier, but also an interpretable statistical likelihood model that adds extra transparency to the task solution; our model reports a value in $[0, 1]$ indicating how likely a customer is to be "satisfied."

We ought to mention that the way we model this Airline Passenger Satisfaction problem by assuming $\boldsymbol{y} = \boldsymbol{0}$ is also related to the SVDD problem (Tax and Duin, 2004) and more specifically, its variant allowing ellipses (Wang et al., 2010). These seek the minimum ball, or in the more general case Mahalanobis ball, which contains all positive examples and none of the negative examples. These formulations allow for some outliers with a Hinge loss, but do not explicitly model the negative examples. They also treat the center of the ball as an optimization parameter, not setting it to $\boldsymbol{0}$ as we do here. Extending the noise and generalization analysis we explore in this paper to that setting would be an interesting future direction.

5.5.3 **Breast Cancer Wisconsin (Diagnostic)**

We here use the Breast Cancer Wisconsin Diagnostic Data Set which is publicly available through the University of California Irvine (UCI) Machine Learning Repository (Wolberg et al., 1995). This data has been created based on 669 samples collected from January 1989 to November 1991. There are 30 real-valued features assigned to each sample which are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. This data set contains 241 instances as malignant, and 458 instances as benign. So, this is the first unbalanced data set we explore (see Unbalanced labeling part in Appendix G.3 for an ablation study on the balance). This data set has been explored in the literature mostly as a classification task where the best performance has been observed by ensembles of ANN and SVM with a 100% accuracy (see Salod and Singh (2020)). Similar to the airline satisfaction task, we here assume that there is an underlying metric and a threshold such that comparing the distance from the origin based on the underlying metric with the threshold determines whether a sample is benign or malignant. Modeling this setting via linear DML (HS noise model), we shuffle and split the data into a train set of size 450 and a test set of size 119 and observed the train and test accuracies for all the samples, the benign samples, and the malignant samples separately. We did this experiment 20 times and recorded the (weighted) average accuracies as in Table 7. Even though the data set is unbalanced, we can see that the model does pretty well on each class of samples. As an advantage over the ensembling classification approaches, we here have a feature transformation $\hat{A}$ which gives us a linear interpretable view of the features.

	All samples	Benign samples	malignant samples
Training accuracy	99.04% (0.30)	99.68% (0.38)	97.97% (0.20)
Test accuracy	97.23% (1.43)	98.60% (0.52)	94.92% (2.82)

Table 7: Different accuracies (std) for the Breast Cancer Wisconsin Diagnostic Data Set.

6. Related Work on Linear DML

A considerable amount of works have been devoted to distance metric learning (Bar-Hillel et al. (2005); Nguyen et al. (2017); Sugiyama (2007); Torresani and Lee (2006); Wang and Zhang (2007); Xiang et al. (2008)). Although recent work has focused primarily on nonlinear distance metric learning, the works most relevant to this article are more classic linear approaches. The method we developed can be categorized as fully supervised linear metric learning in which the scalability is in terms of both number of examples and dimension. Ours has bounded sample complexity is $O(\frac{d^2}{\varepsilon^2}\log\frac{d}{\varepsilon})$ in the dimension d for ε error in our loss function, and in practice we run gradient descent, where each iteration is linear in the data size N and has quadratic dependence for dimension d. Related works do not provide sample complexity bounds. Note that our method learns a Mahalanobis distance (a positive semi-definite matrix $\boldsymbol{M}$) and a threshold τ. We next compare with the most similar prior work.

6.1 Constrained Optimization Approaches

Xing et al. (2002) provide the first method to learn a Mahalanobis distance by maximizing the sum of distances between points in the dissimilarity set ($\mathcal{D}$) under the constraint that the sum of squared

distance between points in the similarity set ($\mathcal{S}$) is upper-bounded:

$$\max_{\boldsymbol{M}\succeq 0} \sum_{(\boldsymbol{x}_i,\boldsymbol{x}_j)\in\mathcal{D}} d_{\boldsymbol{M}}(x_i,\boldsymbol{x}_j)$$
$$\text{s.t.} \sum_{(\boldsymbol{x}_i,\boldsymbol{x}_j)\in\mathcal{S}} d^2_{\boldsymbol{M}}(\boldsymbol{x}_i,\boldsymbol{x}_j) \leq 1.$$

It can be shown that this is a convex optimization which was solved by a proximal gradient ascent which, in each step, takes a gradient ascent step of the objective function, then projects back to the set of constraints, the cone of p.s.d. matrices. The projection to the p.s.d. cone uses full eigen-decomposition with $O(d^3)$ time complexity. So, as the dimension gets large it quickly gets intractable, while in our method we only deal with computing the Mahalanobis distance which takes $O(d^2)$ time complexity.

Note that the model proposed by Xing et al. (2002) takes into account all the information of similar and dissimilar pairs by aggregating all similarity constraints together as well as all dissimilarity constraints. In contrast, the DML-eig method proposed by Ying and Li (2012) maximizes the minimum distance between dissimilar pairs instead of maximizing the sum of their distances. They develop a subgradient ascent procedure to optimize their formulation which does not require a projection, but still uses an $O(d^3)$ eigendecomposition step. Intuitively, this model prioritizes separating dissimilar pairs over keeping similar pairs close. Experimentally, they show that their method outperforms Xing et al. (2002). In Subsection 5.4 we experimentally compare our model with DML-eig showing that our approach works better in terms of performance, accuracy, and dealing with noise.

6.2 Unconstrained Optimization over A

Taking into account the fact that any p.s.d matrix $\boldsymbol{M}$ can be decomposed into $\boldsymbol{M} = \boldsymbol{A}\boldsymbol{A}^\top$, Goldberger et al. (2004) define the expected leave-one-out error of a stochastic nearest neighbor classifier in the projection space induced by $\boldsymbol{A}$. They defined the probability that $\boldsymbol{x}_i$ is similar (close) to $\boldsymbol{x}_j$ as

$$p_{ij}(\boldsymbol{A}) = \frac{\exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2_{\boldsymbol{M}}\right)}{\sum_{k\neq i}\exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2_{\boldsymbol{M}}\right)}, \quad p_{ii} = 0$$

and the probability that $\boldsymbol{x}_i$ is correctly classified as

$$p_i = \sum_{\{j:(\boldsymbol{x}_i,\boldsymbol{x}_j)\in\mathcal{S}\}} p_{ij}.$$

To learn $\boldsymbol{A}$, they solve $\arg\max_{\boldsymbol{A}} \sum_i p_i$. This is not a convex optimization and thus leads to a local maximum rather than a global one.

Using an MLE approach, Guillaumin et al. (2009) define a Logistic loss function which matches a special case of our loss function when the noise comes from a Logistic distribution. They start with the assumption that the similarity is determined via a Bernoulli distribution whose success probability (being similar) is $\sigma(\tau - d_{\boldsymbol{M}}(x_i, x_j))$. Maximizing the likelihood under this probabilistic model then yields an approximation of $\boldsymbol{M}$ and τ. This is exactly the approach we take in Subsection 3.1. However, instead of directly assuming such a model, we derive our model from certain noise assumptions and thus obtain the best theoretical model possible under these assumptions. We also

prove that alternate (nonLogistic) formulations of the loss function are the MLE solution under different noise assumptions. Moreover, we conclude that each of these models is capable of parameter recovery with a sufficiently large sample size.

In each of the above settings, we can rewrite the optimization function in terms of $\boldsymbol{A}$, where $\boldsymbol{M} = \boldsymbol{A}\boldsymbol{A}^\top$. This makes the problem unconstrained which is a good advantage since it eliminates expensive sub-gradient projection and moreover, we can restrict $\boldsymbol{A}$ to be rectangular inducing a low-rank $\boldsymbol{M}$. The main limitation of this formulation (even when $\boldsymbol{A}$ is a square matrix) is that it is non-convex and thus subject to local maxima. However, in our formulation, a result by Journée et al. (2010) resolves this issue.

6.3 Empirical Loss Minimization Framework

There are some other related works proposing an empirical loss minimization framework; for a thorough review see Chapter 8 of the book by Bellet et al. (2015). The prediction performance of the learned metric has been studied in some works such as Balcan et al. (2008); Jin et al. (2009); Bellet et al. (2011, 2012a,b); Guo and Ying (2014); Cao et al. (2016). Broadly speaking, for an unknown data probability distribution, they considered different meaningful constrained cost functions as the true risk and then they studied the convergence of the empirical risk to the true risk. As their setting is a bit different from ours, we recall their data assumptions here. Given labeled examples $\{(\boldsymbol{x}_i, y_i) \colon i = 1, \dots, N\}$ where $\boldsymbol{x}_i \in \mathbb{R}^d$, $\|\boldsymbol{x}_i\| \leq F$ and $y_i \in \{1, \dots, m\}$, they define a similar (close) pair set $\mathcal{S}$ and a dissimilar (far) pair set $\mathcal{D}$ as follows:

$$\mathcal{S} = \{(\boldsymbol{x}_i, \boldsymbol{x}_j \colon y_i = y_j\} \quad \text{and} \quad \mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{x}_j \colon y_i \neq y_j\}.$$

Note that the pairs here are not i.i.d. as in our formulation. From $O(n)$ given data points, we can feed our cost function with only n i.i.d. pairs while they can do it with $O(n^2)$ pairs. It might give the impression that these methods should allow for much stronger convergence results, but it is not the case.

In the following we briefly review some of their results and then compare them to ours. The primary outcome of these findings is to demonstrate the overall reliability of a metric learning approach, rather than offering precise estimations of the generalization loss.

Following the idea of maximum margin classifiers, Jin et al. (2009) adapted the uniform stability framework (Bousquet and Elisseeff (2002) and McDiarmid's inequality) to metric learning to obtain a generalization bound. They considered

$$C(\boldsymbol{M}) = \frac{2c}{n(n-1)} \sum_{i<j} L\left(y_{ij}(1 - \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2_{\boldsymbol{M}})\right) + \frac{1}{2}\|\boldsymbol{M}\|^2_F \tag{13}$$

as the regularized empirical cost function where $y_{ij} = 1$ if $(i,j) \in \mathcal{S}$ and $y_{ij} = -1$ if $(i,j) \in \mathcal{D}$ and $L(z)$ is a standard loss function which is ζ-Lipschitz. As a main result, they proved that the empirical loss $C(\boldsymbol{M})$ converges in probability measure to the true cost $\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}',y} L\left(y(1 - \|\boldsymbol{x} - \boldsymbol{x}'\|^2_{\boldsymbol{M}})\right) + \frac{1}{2}\|\boldsymbol{M}\|^2_F$ with the sample complexity $O\left(\frac{s(d)^2 \ln 1/\delta}{\varepsilon^2}\right)$ where $s(d)$ comes from a constraint $\operatorname{trace}(\boldsymbol{M}) \leq s(d)$, where the hidden constant depends on ζ, F. Note that if $s(d)$ is considered a constant, this provides a sample complexity independent of dimension; but it may be that the best $\boldsymbol{M}$ minimizing the cost function does not follow this constraint. Comparing to our sample complexity (Theorem 4), both bounds share almost the same dominant part (assuming d is fixed). However, they use $O(n^2)$

pairs in their optimization while we only use n; thus our optimization framework is more scalable in n. In our setting, we can sample $O(n)$ disjoint pairs from the $O(n^2)$ pairs to simulate our required i.i.d. samples. It is worth mentioning that the cost function $C(\boldsymbol{M})$ can also be interpreted as a U-statistics of degree two. In their work, Clémençon et al. (2016) investigated the computational complexity of U-statistics and found that this empirical risk can be substituted with much simpler Monte Carlo estimates, known as incomplete U-statistics. These estimates are based on just $O(n)$ terms while $C(\boldsymbol{M})$ has $O(n^2)$ terms, significantly reducing computational complexity without compromising the learning rate.

Bian and Tao (2011) considered a similar loss (without the regularizer term). They theoretically and empirically studied the convergence of the empirical risk in this setting for some appropriate choice of L, focusing on log loss and hinge loss. They proved that the empirical loss converges (in probability measure) to the corresponding true loss with the rate $O(1/\sqrt{n})$. This bound does not resolve the sample complexity since it is presented as a function of n without working out the dependencies to the other parameters. They also concluded that the minimizer of the empirical loss $(\hat{\boldsymbol{M}}, \hat{\tau})$ converges in measure to the minimizer of the true loss $(\boldsymbol{M}^*, \tau^*)$ as n goes to infinity, but does not identify specific conditions that must hold for this to be true, or finite sample bounds, as our work does.

In a sequel, Bian and Tao (2012) examined a data assumption that closely resembled ours, along with empirical and accurate losses like our own. They demonstrated that the empirical loss converges to the true loss on the optimal model, with a sample complexity comparable to ours in terms of ε, δ, and d. However, it is worth noting that their study did not include noise in their setting, it was not proven that their optimization model is theoretically optimal under some generating model parameterized by $\boldsymbol{M}^*$ as is done in this article, and they did not investigate the recovery of ground truth parameters.

Extending the robustness framework (Xu and Mannor (2012)) to metric learning, Bellet and Habrard (2015) studied the deviation between the true and empirical loss. The cost function they worked with is again similar to the one in Equation 13. They proved that the empirical loss converges in probability measure to the corresponding true loss. We can simplify their result to the sample complexity bound $O(\frac{s(d)+\ln(1/\delta)}{\varepsilon^2})$ where $s(d)$ can be exponentially large in terms of d. It should be noted that the constant appearing in their data assumption impacts the constant in this sample complexity bound. For a fixed d, comparing our complexity bound with theirs, we again can see that the dominant parts are almost the same while their algorithm operates on n^2 distances for a set of n points. Afterwards Guo and Ying (2014); Cao et al. (2016), employing a different similarity learning optimization problem, established a comparable error bound in terms of Rademacher average which is upper bounded by a function of data bound F. More precisely, under our data assumption, we can translate their error bound to a sample complexity bound which depends linearly on d and has the dominant term $O(\frac{\ln(1/\delta)}{\varepsilon^2})$. It is similar to the other above-mentioned bounds.

In the following we briefly recall some advantages of our model compared to the above mentioned methods.

- All methods discussed above model metric learning as an optimization problem which penalizes mismatches, including using constrained optimization on $\boldsymbol{M}$. However, they do not prove that these optimization problems are theoretically optimal under some generating model parameterized by $\boldsymbol{M}^*$ as is done in this article.

- These algorithms deal with a more expensive optimization problem which uses $O(n^2)$ pairs for n points while our method uses only $O(n)$ pairs. Despite the extra information, these algorithms do not lead to a more favorable scaling between the sample size n and the error ε compared to our method.

- These other methods do not provide recovery guarantees on generating parameters as we do. This in turn allows us to provide low-rank approximation and dimensionality reduction results, since we can bound the effects of truncating small model parameters.

- Furthermore, since we derive the loss functions from various noise models, we can recover these model parameters even in the presence of (correctly modeled) noise.

6.4 Information Theoretic Modeling

Nguyen et al. (2017) assume that $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{y} | \boldsymbol{x}, \boldsymbol{y} \in \mathcal{S} \sim \mathcal{N}(0, \Sigma_{\mathcal{S}})$ and $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{y} | \boldsymbol{x}, \boldsymbol{y} \in \mathcal{D} \sim \mathcal{N}(0, \Sigma_{\mathcal{D}})$. For any linear transformation $\boldsymbol{x}' = \boldsymbol{A}^\top \boldsymbol{x}$, this can be written

$$\boldsymbol{z}' = \boldsymbol{x}' - \boldsymbol{y}' | \boldsymbol{x}, \boldsymbol{y} \in \mathcal{S} \sim \mathcal{N}(0, \boldsymbol{A}^\top \Sigma_{\mathcal{S}} \boldsymbol{A}) = f_{\boldsymbol{A}}(\boldsymbol{z}')$$

and

$$\boldsymbol{z}' = \boldsymbol{x}' - \boldsymbol{y}' | \boldsymbol{x}, \boldsymbol{y} \in \mathcal{D} \sim \mathcal{N}(0, \boldsymbol{A}^\top \Sigma_{\mathcal{D}} \boldsymbol{A}) = g_{\boldsymbol{A}}(\boldsymbol{z}').$$

Their goal is to find $\boldsymbol{A}$ maximizing Jeffrey divergence, i.e., to solve the following optimization problem;

$$\max_{\boldsymbol{A} \in \mathbb{R}^{d \times k}} \mathrm{KL}(f_{\boldsymbol{A}}, g_{\boldsymbol{A}}) + \mathrm{KL}(g_{\boldsymbol{A}}, f_{\boldsymbol{A}}),$$

where KL stands here for Kullback-Leibler divergence. As both distributions $g_{\boldsymbol{A}}$ and $f_{\boldsymbol{A}}$ are multivariate Gaussian, one can compute $\mathrm{KL}(f_A, g_A) + \mathrm{KL}(g_A, f_A)$ as a function of $\boldsymbol{A}$ and reduce the optimization problem to

$$\max_{\boldsymbol{A} \in \mathbb{R}^{d \times k}} \operatorname{tr}\left((\boldsymbol{A}^\top \Sigma_{\mathcal{S}} \boldsymbol{A})^{-1} (\boldsymbol{A}^\top \Sigma_{\mathcal{D}} \boldsymbol{A}) + (\boldsymbol{A}^\top \Sigma_{\mathcal{D}} \boldsymbol{A})^{-1} (\boldsymbol{A}^\top \Sigma_{\mathcal{S}} \boldsymbol{A}) \right).$$

Setting the derivative of this objective function to zero, they present a solution to this non-convex optimization problem. In practice, they use MLE to replace $\Sigma_{\mathcal{S}}$ and $\Sigma_{\mathcal{D}}$ by their sample estimations, and since the formulation is not convex, the identified answer may be a local optimum.

6.5 Low-Rank Metric Learning

As explained in Section 1, linear distance metric learning approaches can be used for linear dimensionality reduction (see Wang and Sun, 2015). When we are dealing with high dimensional space or a huge number of data points, solving Optimizations 5 (or Optimizations 10 for $k = \Theta(d)$) will be costly. As reducing the matrix size in these optimizations reduces the complexity of search spaces, to resolve this issue, we can think of adding some low-rank constraint to these optimizations, e.g., $\operatorname{rank}(\boldsymbol{M}) \ll d$ in Optimizations 5 or $k \ll d$ in Optimizations 10. As a downside, it turns these optimization problems non-convex and thus the regular approaches such as gradient decent tend to fail easily (see Mu, 2016; Wen and Yin, 2013). Liu et al. (2019), dealing with this challenge, introduced a fast low-rank metric learning method that worked well for several data points as benchmarks.

Although they still have to face a non-convex optimization, as a standard way, they employed manifold optimization methods (Shukla and Anand, 2015; Balzano et al., 2010; Vandereycken, 2013) to overcome the issue. As they worked with a different input setting as in linear DML, we are not able to present a direct comparison between their approach and ours.

7. Conclusion and Discussion

In this paper we provide and analyze a simple and elegant method to solve the linear distance metric learning problem. That is, given a set of iid pairs of points labeled close or far, our method learns a Mahalanobis distance that maintains these labels for some threshold. This arises when in data analysis one needs to learn how to compare various coordinates, which may be in different units, but not introduce non-linearity for reasons of interpretability, equation preservation, or maintaining linear structure. Our method reduces to unconstrained gradient descent, has a simple sample complexity bound, and shows convergence in a loss function and in parameter space. In fact, this convergence holds even under noisy observations.

Moreover, our method is the first approach to show that the learned Mahalanobis distance can be truncated to a low-rank model that can provably maintain the accuracy in the loss function and in parameters.

Finally, we demonstrate that this method works empirically as well as the theory predicts. We can obtain high accuracy (over 99%) and parameter recovery (less than 1.01 multiplicative error) on noiseless and noisy data, and on synthetic and real data. For instance, even if 45% of the data is mislabeled we can with very high accuracy recover the true model parameters. Additionally we show this simple solution nearly matches the best engineered solutions on two real world data challenges.

7.1 Limitations

In our formulation of linear DML, we assume that we are given N i.i.d. observations of pair $(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathbb{R}^d \times \mathbb{R}^d$ and each pair is given a label $\ell_i \in \{\mathrm{Far}, \mathrm{Close}\}$. We discuss the Airline Passenger Satisfaction modeling problem in Section 5.5.2 where by always setting $\boldsymbol{y}_i$ to the origin, this is a direct and natural modeling: each observation generates one pair. However, in other real-world settings, we are only provided with n observations with the ability to query if any pair has label Far or Close. This can induce $\Theta(n^2)$ pairs, but they are not i.i.d. In practice, one would like to use all of these pairs, or perhaps restricted to some local constraints in a neighborhood. But since these are not i.i.d., our analysis does not apply. What we can do is randomly partition the data into $n/2$ pairs; now if the original n observations are i.i.d., then these pairs are also i.i.d. from some distribution, and our analysis holds. While this only generates $N = O(n)$ pairs, not $O(n^2)$, our analysis shows error converges at a rate of roughly $1/\sqrt{N}$. This basically matches the convergence rate for known methods which use all $\Theta(n^2)$ pairs, and converge at a rate $1/\sqrt{n}$. Managing such dependency, and potentially improving to a $1/n$ convergence rate, is a challenging direction to consider for future work.

Another limitation in our modeling in the noisy setting, is that we prove strong convergence and parameter recovery, only when the loss function corresponds with the noise model generating the data. In practice one does not always know the noise model, and in fact there may not be one well-defined noise model. As a result, a user must chose a loss function. In this case, we generically recommend the Logistic loss. It is widely used, has a closed form, the cdf is 1-log-Lipschitz, and as

discussed, it is fairly similar to other common noise models. Moreover, we show empirically that it performs nearly as well under other noise types we considered as it does under Logistic noise.

Finally, the analysis requires several clearly stated assumptions on the model Model Assumption (that the parameters $\boldsymbol{M}^*$ and τ^* are bounded) and the data Data Assumption (that the 2-norm of the data is bounded). If we do not have such bounds, then our analysis does not have a guarantee. In fact, we observe in the experiment on Equations of State for Combustion Simulation in Section 5.5.1 that without properly normalizing, the algorithm performs poorly. This normalization has the effect of properly shaping the data, and as a result the optimal model, so that it satisfies these assumptions. In this case our algorithm converges quickly to small loss and recovers the near-optimal parameters.

Acknowledgments

JP thanks NSF IIS-1816149, CCF-2115677, and CDS&E-1953350; AL thanks NSF DMS-2309570 and NSF DMS-2136198. The authors are deeply grateful to the reviewers for their thorough examination and contributions to the refinement of the paper.

Appendix A. Proof of Lemma 1

Proof To prove Lemma 1, it suffices to show that $\|\boldsymbol{z}\|_{\boldsymbol{M}_1}^2 = \|\boldsymbol{z}\|_{\boldsymbol{M}_2}^2$ for every $\boldsymbol{z} \in \mathbb{R}^d$. W.l.o.g. we may assume that $\tau_1 = \tau_2 = 1$. For an arbitrary $\boldsymbol{z}_0 \in \mathbb{R}^d$, consider the two following different cases.

- $\|\boldsymbol{z}_0\|_{\boldsymbol{M}_1}^2 = 0$. In this case, if $\|\boldsymbol{z}_0\|_{\boldsymbol{M}_2}^2 > 0$, then for sufficiently large α, we should have $\|\alpha\boldsymbol{z}_0\|_{\boldsymbol{M}_2}^2 > 1$ while $\|\alpha\boldsymbol{z}_0\|_{\boldsymbol{M}_1}^2 = 0 < 1$ contradicting the fact that the two functions $\boldsymbol{z} \mapsto \mathbb{1}_{\left\{\|\boldsymbol{z}\|_{\boldsymbol{M}_1}^2 - \tau_1 \geq 0\right\}}$ and $\boldsymbol{z} \mapsto \mathbb{1}_{\left\{\|\boldsymbol{z}\|_{\boldsymbol{M}_2}^2 - \tau_2 \geq 0\right\}}$ agree for all $\boldsymbol{z}$.
- $\|\boldsymbol{z}_0\|_{\boldsymbol{M}_1}^2 > 0$ and $\|\boldsymbol{z}_0\|_{\boldsymbol{M}_2}^2 > 0$. Consider $\alpha, \beta > 0$ such that $\|\alpha\boldsymbol{z}_0\|_{\boldsymbol{M}_1}^2 = \|\beta\boldsymbol{z}_0\|_{\boldsymbol{M}_2}^2 = 1$. We need to show that $\alpha = \beta$. For a contradiction, consider $c > 0$ such that $\alpha < c < \beta$. Now it is clear that

$$\mathbb{1}_{\left\{\boldsymbol{z}\colon \|\boldsymbol{z}\|_{\boldsymbol{M}_1}^2 - \tau_1 \geq 0\right\}}(c\boldsymbol{z}_0) = 1 \qquad \text{and} \qquad \mathbb{1}_{\left\{\boldsymbol{z}\colon \|\boldsymbol{z}\|_{\boldsymbol{M}_2}^2 - \tau_2 \geq 0\right\}}(c\boldsymbol{z}_0) = 0,$$

  a contradiction.

■

Appendix B. Basic Properties Related to Optimization Problem 5

In this part, we derive some basic properties related to Optimization Problem 5. In particular, we will see that it is a convex optimization. Using this as the main result of this subsection, we prove that the true loss is uniquely minimized at the ground truth parameters.

First, note that since any convex combination of two p.s.d. matrices is still p.s.d, using triangle inequality for spectral norm, we conclude that the search space $\mathcal{M} \times [0, B]$ is convex.

Observation 1 *For every fixed $\boldsymbol{z} \in \mathbb{R}^d$ and $\ell \in \{-1, 1\}$, $-\log \sigma(\ell(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau))$ as a function of $(\boldsymbol{M}, \tau)$ is convex.*

Proof To prove the assertion, it suffices to show that $\log \sigma(\ell(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau))$ is concave. Consider arbitrary $\boldsymbol{M}_1, \boldsymbol{M}_2 \in \mathcal{M}$ and $\tau_1, \tau_2 \in [0, B]$. We remind that $\log \sigma(\cdot)$ is a concave function. Also, for each $\lambda \in [0, 1]$,

$$\|\boldsymbol{z}\|_{\lambda \boldsymbol{M}_1 + (1-\lambda)\boldsymbol{M}_2}^2 - (\lambda\tau_1 + (1-\lambda)\tau_2) = \lambda(\|\boldsymbol{z}\|_{\boldsymbol{M}_1}^2 - \tau_1) + (1-\lambda)(\|\boldsymbol{z}\|_{\boldsymbol{M}_2}^2 - \tau_2).$$

Combining these two facts implies $\log \sigma(\ell(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - b))$ as a function of $(\boldsymbol{M}, \tau)$ is concave, completing the proof. ■

This observation immediately implies that both $R_N(\boldsymbol{M}, \tau)$ and $R(\boldsymbol{M}, \tau)$ are convex functions as well. Thus

$$\min_{(\boldsymbol{M},\tau) \in \mathcal{M} \times [0,B]} R(\boldsymbol{M}, \tau) \qquad and \qquad \min_{(\boldsymbol{M},\tau) \in \mathcal{M} \times [0,B]} R_N(\boldsymbol{M}, \tau)$$

are both convex optimization problems. Although the following observation is quite technical, it is necessary for the proof of succeeding results.

Observation 2 *Let $S \subseteq \mathbb{R}_{\geq 0}^d$ be a set with zero Lebesgue measure. Then $S^{\frac{1}{2}} = \{\boldsymbol{x} \in \mathbb{R}^d \colon \boldsymbol{x}^2 \in S\}$ has also zero Lebesgue measure.*

Proof For each $I = \{i_1, \ldots, i_k\} \subseteq [d]$, define

$$S_I^{\frac{1}{2}} = \left\{\boldsymbol{z} \colon \exists\, \boldsymbol{x} \in S \; s.t. \; z_i = \sqrt{x_i} \text{ for } i \in [n] \setminus I \text{ and } z_i = -\sqrt{x_i} \text{ for } i \in I\right\}.$$

It is clear that

$$S^{\frac{1}{2}} = \bigcup_{I \subseteq [d]} S_I^{\frac{1}{2}}.$$

Accordingly, it suffices to show $\mu(S_I^{\frac{1}{2}}) = 0$ for each $I \subseteq [d]$. For an arbitrary $I \subseteq [d]$, define $f_i : \mathbb{R}_{\geq 0}^d \longrightarrow \mathbb{R}^d$ such that

$$f_I(\boldsymbol{x}) = (l_1\sqrt{x_1}, \ldots, l_d\sqrt{x_d}) \qquad \text{where} \qquad l_i = \begin{cases} -1 & i \in [I] \\ 1 & i \notin [I]. \end{cases}$$

It is clear that f_I is a one-to-one continuous function and $f_I(S) = S_I^{\frac{1}{2}}$. To fulfill the proof, we prove that for each $L \in \mathbb{N}$, $\mu_L\left(S_I^{\frac{1}{2}} \cap \left[-\sqrt{L}, \sqrt{L}\right]^d\right) = 0$. This implies that

$$\begin{aligned} \mu_L\left(S_I^{\frac{1}{2}}\right) &= \mu_L\left(\bigcup_{L=1}^{\infty}\left(S_I^{\frac{1}{2}} \cap \left[-\sqrt{L}, \sqrt{L}\right]^d\right)\right) \\ &\leq \sum_{L=1}^{\infty} \mu_L\left(S_I^{\frac{1}{2}} \cap \left[-\sqrt{L}, \sqrt{L}\right]^d\right) \\ &= 0 \end{aligned}$$

Let L be a fixed positive integer and consider an arbitrary $\varepsilon > 0$. For each $i \in [d]$, consider the interval

$$J_i = [0, L] \times \cdots \times \underbrace{[0, \frac{\varepsilon^2}{2^{2d} L^{d-1} d^2}]}_{\text{the } i\text{-th interval}} \times \cdots \times [0, L].$$

It is clear that the volume of each J_i is $\frac{\varepsilon^2}{2^{2d}d^2}$. Also, for the image of each J_i, we have

$$f_I(J_i) \subset [-\sqrt{L},\sqrt{L}] \times \cdots \times \underbrace{[-\frac{\varepsilon}{2^d L^{\frac{d-1}{2}} d}, \frac{\varepsilon}{2^d L^{\frac{d-1}{2}} d}]}_{\text{the } i\text{-th interval}} \times \cdots \times [-\sqrt{L},\sqrt{L}].$$

This implies that the volume of $f_I(J_i)$ is at most $\frac{\varepsilon}{d}$. Since, $f_I(\cdot)$ is Lipschitz over $[0,L] \times \cdots \times [0,L] \setminus \bigcup_{i\in[d]} J_i$, the zero Lebesgue measure sets will be mapped to zero Lebesgue measure sets by f_I. Therefore,

$$\mu_L\left(f_I\left(S \cap ([0,L]^d \setminus \cup_{i\in[d]} J_i)\right)\right) = 0.$$

Consequently,

$$\begin{aligned}\mu_L\left(S_I^{\frac{1}{2}} \cap \left[-\sqrt{L},\sqrt{L}\right]\right) &= \mu_L\left(f_I\left(S \cap ([0,L]^d \setminus \cup_{i\in[d]} J_i)\right)\right) + \mu_L\left(f_I\left(S \cap (\cup_{i\in[d]} J_i)\right)\right)\\ &\leq 0 + d\frac{\varepsilon}{d} = \varepsilon.\end{aligned}$$

Since ε is arbitrary, $\mu_L\left(S_I^{\frac{1}{2}} \cap \left[-\sqrt{L},\sqrt{L}\right]^d\right) = 0$, completing the proof. ■

Using this observation, we can prove the following useful lemma.

Lemma 16 *Let M_1, M_2 be two symmetric matrices and $c \in \mathbb{R}$. If there is $Q \subseteq \mathbb{R}^d$ such that $\mu_L(Q) > 0$ (Lebesgue measure) and*

$$\|\boldsymbol{x}\|^2_{\boldsymbol{M}_1} = \|\boldsymbol{x}\|^2_{\boldsymbol{M}_2} + c \qquad \textit{for each } \boldsymbol{x} \in Q,$$

then $\boldsymbol{M}_1 = \boldsymbol{M}_2$ and $c = 0$.

Proof For each $\boldsymbol{z} \in Q$, we clearly have $\boldsymbol{x}_i^t(\boldsymbol{M}_1 - \boldsymbol{M}_2)\boldsymbol{x}_i = c$. Since $\boldsymbol{M} = \boldsymbol{M}_1 - \boldsymbol{M}_2$ is a real value symmetric metric, $\boldsymbol{M} = \boldsymbol{U}^t\boldsymbol{D}\boldsymbol{U}$ where $\boldsymbol{U}$ is an orthonormal matrix and $\boldsymbol{D}$ is a diagonal $d\times d$ matrix whose (i,i) element is a_i. To prove the assertion, it suffices to show that $\boldsymbol{D} = 0$. For a contradiction, suppose that $\boldsymbol{D} \neq 0$. Set $Q' = \{\boldsymbol{U}\boldsymbol{x} : \boldsymbol{x} \in Q\}$ and $S = \left\{\boldsymbol{y} \in \mathbb{R}^d_{\geq 0} : \langle \boldsymbol{y}, (a_1,\ldots,a_d)\rangle = c\right\}$. For each $\boldsymbol{z} = (z_1,\ldots,z_d) \in Q'$,

$$\begin{aligned}\sum_{i=1}^d z_i^2 a_i &= \boldsymbol{z}^t\boldsymbol{D}\boldsymbol{z}\\ &= \boldsymbol{x}^t\boldsymbol{U}^t\boldsymbol{D}\boldsymbol{U}\boldsymbol{x}\\ &= \boldsymbol{x}^t\boldsymbol{M}\boldsymbol{x} = c,\end{aligned}$$

which concludes that

$$Q' \subseteq \left\{\boldsymbol{z} \in \mathbb{R}^d : \langle \boldsymbol{z}^2, (a_1,\ldots,a_d)\rangle = c\right\} = S^{\frac{1}{2}}.$$

Using Observation 2, since we know $\mu_L(S) = 0$ we obtain $\mu_L(S^{\frac{1}{2}}) = 0$ and thus $\mu_L(Q') = 0$. On the other hand, $\mu_L(Q) = \mu_L(Q')$ which implies $\mu_L(Q) = 0$, a contradiction. ■

Appendix C. ε-Cover (ε-Net) for $\mathcal{M} \times [0, B]$

As we are going to prove a uniform convergence theorem between empirical and true losses, we need to define the ε-cover of a metric space. For a metric space $(\mathcal{X}, d)$, an ε-cover $\mathcal{E}$ is a subset of $\mathcal{X}$ such that for each $x \in \mathcal{X}$, there is some $y \in \mathcal{E}$ with $d(x, y) < \varepsilon$. In the following, we introduce an ε-cover for $\mathcal{M} \times [0, B]$. However, we should first define a metric over $\mathcal{X} = \mathcal{M} \times [0, B]$. For each $(\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2) \in \mathcal{X}$, we define

$$\mathrm{d}\left((\boldsymbol{M}_1, \tau_1), (\boldsymbol{M}_2, \tau_2)\right) = \|\boldsymbol{M}_1 - \boldsymbol{M}_2\|_2 + |\tau_1 - \tau_2|.$$

Lemma 17 *There exists an ε-cover $\mathcal{E}$ of $\mathcal{M} \times [0, B]$ under metric d of size*

$$\frac{B}{\varepsilon}\left(\frac{4\beta d\sqrt{d}}{\varepsilon}\right)^{d^2}.$$

Proof The inequality

$$\|\boldsymbol{M}\|_F \leq \sqrt{d}\|\boldsymbol{M}\|_2 \leq \beta\sqrt{d}$$

indicates that $\mathcal{M}$ is a subset of the d^2 dimensional Euclidean ball of the radius $\beta\sqrt{d}$ centered at the origin, i.e.,

$$\mathcal{M} = \{\boldsymbol{M}_{d\times d} \colon \boldsymbol{M} \text{ is p.s.d. and } \|\boldsymbol{M}\|_2 \leq \beta\} \subseteq \{\boldsymbol{M} \in \mathbb{R}^{d^2} \colon \|\boldsymbol{M}\|_F \leq \beta\sqrt{d}\}.$$

It is known that a k-dimentional Euclidean ball of radius r can be covered by at most $\left(\frac{2r\sqrt{k}}{\varepsilon}\right)^k$ number of balls of radius ε. So, $\mathcal{M}$ has an $\frac{\varepsilon}{2}$-cover $\mathcal{E}_2$ of size at most $\left(\frac{4\beta d\sqrt{d}}{\varepsilon}\right)^{d^2}$. As an $\frac{\varepsilon}{2}$-cover $\mathcal{E}_2$ for $[0, B]$ (with respect to L_1-norm), we can partition $[0, B]$ into $\frac{B}{\varepsilon}$ intervals of length ε and consider the end points of those intervals as the $\frac{\varepsilon}{2}$-cover. Now the cartesian product of these two $\frac{\varepsilon}{2}$-covers, $\mathcal{E}_1 \times \mathcal{E}_2$, is an ε-cover of size

$$\frac{B}{\varepsilon}\left(\frac{4\beta d\sqrt{d}}{\varepsilon}\right)^{d^2}.$$

for $\mathcal{X} = \mathcal{M} \times [0, B]$ with respect to metric d. ■

Appendix D. Uniform Convergence of R_N to R

Although we have proved in Theorem 2 that the true loss $R(\boldsymbol{M}, \tau)$ is uniquely minimized at $(\boldsymbol{M}^*, \tau^*)$, in reality, we do not have the true loss. Indeed, we only have access to the empirical loss $R_N(\boldsymbol{M}, \tau)$. In this part, broadly speaking, we will show that $R_N(\boldsymbol{M}, \tau)$ is uniformly close to $R(\boldsymbol{M}, \tau)$ as N gets large, and then, we conclude, instead of minimizing $R_N(\boldsymbol{M}, \tau)$, we can minimize $R_N(\boldsymbol{M}, \tau)$ to approximately find $(\boldsymbol{M}^*, \tau^*)$.

In the next lemma, we will see that if the two p.s.d. matrices are close via spectrum norm, then the Mahalanobis norm defined based on these two matrices are also close.

Observation 3 (Equation 7) *For given two p.s.d. $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$,*

$$|\|\boldsymbol{x}\|_{\boldsymbol{M}_1}^2 - \|\boldsymbol{x}\|_{\boldsymbol{M}_2}^2| \leq \|\boldsymbol{M}_1 - \boldsymbol{M}_2\|_2\|\boldsymbol{x}\|^2.$$

Proof Using Cauchy–Schwarz inequality and the definition of the spectral norm, we obtain

$$\begin{aligned} |\|\boldsymbol{x}\|_{\boldsymbol{M}_1}^2 - \|\boldsymbol{x}\|_{\boldsymbol{M}_2}^2| &= |\boldsymbol{x}^\top(\boldsymbol{M}_1 - \boldsymbol{M}_2)\boldsymbol{x}| \\ &= \langle \boldsymbol{x}^\top, \boldsymbol{x}^\top(\boldsymbol{M}_1 - \boldsymbol{M}_2)\rangle \\ &\leq \|\boldsymbol{x}\|\|(\boldsymbol{M}_1 - \boldsymbol{M}_2)\boldsymbol{x}\| \\ &\leq \|(\boldsymbol{M}_1 - \boldsymbol{M}_2)\|_2\|\boldsymbol{x}\|^2, \end{aligned}$$

concluding the inequality. ■

As $-\log \Phi_{\text{Noise}}(\cdot)$ is a decreasing function, using Equation 2, we have

$$0 \leq -\log \Phi_{\text{Noise}}\left(\ell_i(\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2 - \tau)\right) \leq -\log \Phi_{\text{Noise}}(-\beta A) = T,$$

which indicates that the random variables $\boldsymbol{z}_i$'s are bounded. Whenever we are dealing with a summation of bounded i.i.d. random variables, one strong concentration inequality to use is Chernoff-Hoeffding bound. This inequality states if $X_1, ..., X_N$ are N independent random variables such that $X_i \in [a_i, b_i]$ almost surely for all i, and $S_N = \dfrac{X_1 + \cdots + X_N}{N}$, then

$$\mathrm{P}\left(|S_n - \mathrm{E}\left[S_n\right]| \geq \alpha\right) \leq 2\exp\left(-\frac{2N^2\alpha^2}{\sum_{i=1}^N (b_i - a_i)^2}\right).$$

Since,

$$R_N(\boldsymbol{M}, \tau) = \frac{1}{N}\sum_{i=1}^{N} -\log \Phi_{\text{Noise}}\left(\ell_i(\|\boldsymbol{z}_i\|_{\boldsymbol{M}}^2 - \tau)\right)$$

and $\mathbb{E}(R_N(\boldsymbol{M}, \tau)) = R(\boldsymbol{M}, \tau)$, we can use Chernoff-Hoeffding bound to control the probability that $|R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)|$ is large.

Lemma 18 *If $E = \{(\boldsymbol{M}_i, \tau_i) \colon i = 1, \ldots, m = m(\alpha)\}$ is an α-cover for $\mathcal{M} \times [0, B]$, then*

$$P\big(|R_N(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}_i, \tau_i)| \geq \alpha \text{ for some } i \in [m]\big) \leq 2me^{-\frac{2N\alpha^2}{T^2}}.$$

Proof Consider a fixed $i \in [m]$. For simplicity of notation, set $Z = -\log \sigma\left(\ell(\|\boldsymbol{x}\|_{\boldsymbol{M}_i} - \tau_i)\right)$ and $Z_j = -\log \sigma\left(\ell_j(\|\boldsymbol{x}_j\|_{\boldsymbol{M}_i} - \tau_i)\right)$. As we explained above, $Z \in [0, T]$, see Table 1, and, using Chernoff-Hoeffding Inequality, we obtain

$$\begin{aligned} P\big(|R_N(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}_i, \tau_i)| \geq \alpha\big) &= P\Big(\Big|\frac{1}{N}\sum_{j=1}^{N} Z_j - E(Z)\Big| \geq \alpha\Big) \\ &\leq 2e^{-\frac{2N\alpha^2}{T^2}}. \end{aligned}$$

Now, using union bound, we have the desired inequality. ■

In the next theorem, we prove that, with high probability, the empirical loss R_N is everywhere close to the true loss R.

Theorem 19 *[Theorem 4, Restated] For any $\varepsilon, \delta > 0$, assume parameters B, F, and β are constants, define*

$$N_d(\varepsilon, \delta) = O\left(\frac{1}{\varepsilon^2}\Big[\log\frac{1}{\delta} + d^2 \log\frac{d}{\varepsilon}\Big]\right).$$

If $N > N_d(\varepsilon, \delta)$, then with probability at least $1 - \delta$,

$$\sup_{(\boldsymbol{M},\tau)\in\mathcal{M}\times[0,B]} |R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| < \varepsilon.$$

Proof To prove the assertion, we can equivalently prove

$$P\left(\sup_{(\boldsymbol{M},\tau)\in\mathcal{M}\times[0,B]} |R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| \geq \varepsilon\right) < \delta.$$

Set $\alpha = \frac{\varepsilon}{3\zeta \max(F,1)}$. Consider $\mathcal{E} = \{(\boldsymbol{M}_i, \tau_i); i = 1, \ldots, m = m(\alpha)\}$ as an α-cover for $\mathcal{M}\times[0, B]$. For an arbitrary $(\boldsymbol{M}, \tau)$, there is an index $i \in [m]$ such that

$$d\left((\boldsymbol{M}, \tau) - (\boldsymbol{M}_i, \tau_i)\right) < \alpha.$$

Consequently, using Lemma 3,

$$|R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}_i, \tau_i)| < \max(F, 1)\zeta\alpha = \frac{\varepsilon}{3}$$

and

$$|R_N(\boldsymbol{M}, \tau) - R_N(\boldsymbol{M}_i, \tau_i)| < \max(F, 1)\zeta\alpha = \frac{\varepsilon}{3}.$$

So far, we have proved that for every $(\boldsymbol{M}, b)$, there exists an index $i \in [m]$ such that

$$|R(\boldsymbol{M}, \tau) - R(\boldsymbol{M}_i, \tau_i)| < \frac{\varepsilon}{3} \qquad \text{and} \qquad |R_N(\boldsymbol{M}, \tau) - R_N(\boldsymbol{M}_i, \tau_i)| < \frac{\varepsilon}{3}.$$

Using triangle inequality, it concludes

$$\begin{aligned}|R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| \leq& |R_N(\boldsymbol{M}, \tau) - R_N(\boldsymbol{M}_i, \tau_i)| + |R_N(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}_i, \tau_i)| \\ &+ |R(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}, \tau)| \\ \leq& \frac{2\varepsilon}{3} + |R_N(\boldsymbol{M}_i, \tau_i) - R(\boldsymbol{M}_i, \tau_i)|.\end{aligned}$$

Via Lemma 17, there is an α-cover of size

$$\begin{aligned}m(\alpha) &= \frac{B}{\alpha}\left(\frac{4\beta d\sqrt{d}}{\alpha}\right)^{d^2} \\ &= \frac{3\zeta \max(F, 1)B}{\varepsilon}\left(\frac{12\zeta \max(F, 1)\beta d\sqrt{d}}{\varepsilon}\right)^{d^2}\end{aligned}$$

for $\mathcal{X} = \mathcal{M} \times [0, B]$ with respect to metric d. On the other hand,

$$\frac{T^2}{2\alpha^2} \log \frac{2m}{\delta} = \frac{9\zeta \max(F,1)^2 T^2}{2\varepsilon^2} \Big[\log \frac{6\zeta \max(F,1) B}{\varepsilon\delta} + d^2 \log \frac{12\zeta \max(F,1)\beta}{\varepsilon} + \frac{3}{2} d^2 \log d \Big] \tag{14}$$

$$= O\left(\frac{1}{\varepsilon^2} \Big[\log \frac{1}{\delta} + d^2 \log \frac{d}{\varepsilon} \Big] \right)$$

As setting

$$\begin{aligned} N &> \frac{T^2}{2\alpha^2} \log \frac{2m}{\delta} \\ &= O\left(\frac{1}{\varepsilon^2} \Big[\log \frac{1}{\delta} + d^2 \log \frac{d}{\varepsilon} \Big] \right) \end{aligned}$$

implies $2me^{-\frac{2N\alpha^2}{T^2}} < \delta$, using Lemma 18, we obtain, with probability at least $1 - \delta$,

$$|R_N(\boldsymbol{M}_j, \tau_j) - R(\boldsymbol{M}_j, \tau_j)| < \alpha = \frac{\varepsilon}{3\zeta \max(F,1)} < \frac{\varepsilon}{3} \qquad \text{for all } j \in [m].$$

Therefore, if $N > O\left(\frac{1}{\varepsilon^2} \Big[\log \frac{1}{\delta} + d^2 \log \frac{d}{\varepsilon} \Big] \right)$, with probability at least $1 - \delta$, for all $(\boldsymbol{M}, \tau)$ we have

$$|R_N(\boldsymbol{M}, \tau) - R(\boldsymbol{M}, \tau)| \le \varepsilon,$$

as desired. ■

Appendix E. Simple Noise Properties in Table 1

In Subsections 3.1, 3.2, 3.3, 3.4, we derived some properties of $\Phi_{\mathrm{Noise}}(\cdot)$ when noise is one of the simple noises listed in Table 1. This section can be seen as a complementary section for those sections. For noises listed in Table 1, one can verify that each of those noise distributions is simple. Here, we verify some other information listed in that table. When $\mathrm{Noise}(\eta)$ is simple, $-\log \Phi_{\mathrm{Noise}}(\eta)$ is a decreasing convex function which implies that $\frac{\mathrm{d}}{\mathrm{d}\eta}\left(-\log \Phi_{\mathrm{Noise}}(\eta)\right) = -\frac{\mathrm{Noise}(\eta)}{\Phi_{\mathrm{Noise}}(\eta)}$ is a negative increasing function. Therefore, $-\log \Phi_{\mathrm{Noise}}(\eta)$ is ζ-Lipschitz over $[-\beta F, \beta B]$ for

$$\zeta = \frac{\mathrm{Noise}(-\beta F)}{\Phi_{\mathrm{Noise}}(-\beta F)}.$$

Also, from Equation 8, we know

$$T = -\log \Phi_{\mathrm{Normal}}(-\beta F).$$

In what follows, we approximate ζ and T for each noise choice.

- **Logistic noise.** In this case, it is easy to see that

  $$\zeta = \frac{\sigma(-\beta F)\sigma(\beta F)}{\sigma(-\beta F)} = \sigma(\beta F) < 1$$

  and $T = -\log \sigma(-\beta F) = \log(1 + e^{\beta F}) \le 1 + \beta F$. Thus, $\Phi_{\mathrm{Logistic}}(\eta)$ in 1-log-Lipschitz and $T = O(\beta F)$.

- **Normal noise.** We start with an approximation of $\Phi_{\text{Normal}}(\eta)$. The derivation of the lower bound is sourced from the online content authored by John D. Cook (refer to Cook). We set $\Phi^c_{\text{Normal}}(\eta) = 1 - \Phi_{\text{Normal}}(\eta) = \Phi_{\text{Normal}}(-\eta)$.

  Set $g(t) = \Phi^c(t) - \frac{1}{\sqrt{2\pi}} \frac{t}{t^2+1} e^{-t^2/2}$. As $g(0) > 0$, $g'(t) = -\frac{2}{\sqrt{2\pi}} \frac{e^{-t^2/2}}{(t^2+1)^2} < 0$ and $\lim_{t\to+\infty} g(t) = 0$, we obtain $\Phi^c(t) \geq \frac{1}{\sqrt{2\pi}} \frac{t}{t^2+1} e^{-t^2/2}$. Using the above formula for ζ, we have

$$\begin{aligned}
\zeta &= \frac{1}{\sqrt{2\pi}} \frac{e^{-\frac{(\beta F)^2}{2}}}{\Phi(-\beta F)} \\
&= \frac{1}{\sqrt{2\pi}} \frac{e^{-\frac{(\beta F)^2}{2}}}{\Phi^c(\beta F)} \\
&\leq \frac{(\beta F)^2 + 1}{\beta F} = O(\beta F).
\end{aligned}$$

  Also,

$$T = -\log \Phi_{\text{Normal}}(-\beta F) = -\log \Phi^c_{\text{Normal}}(\beta F) = O((\beta F)^2).$$

- **Laplace noise.** In this setting,

$$\zeta = \frac{\text{Noise}(-\beta F)}{\Phi_{\text{Noise}}(-\beta F)} = \frac{\frac{1}{2} e^{-\beta F}}{\frac{1}{2} e^{-\beta F}} = 1$$

  and $T = -\log \Phi_{\text{Normal}}(-\beta F) = \beta F \log 2 = O(\beta F)$.

- **Hyperbolic secant noise.**

  Remind that $\Phi^c(t) = \int_t^\infty \frac{1}{2} \text{sech}(\frac{\pi}{2}\eta) \mathrm{d}\eta$. Set

$$g(t) = \Phi^c(t) - \frac{1}{\pi} \text{sech}(\frac{\pi}{2} t)$$

  Also, note that $g(0) > 0$, $\lim_{t\to+\infty} g(t) = 0$, and

$$\begin{aligned}
g'(t) &= -\frac{1}{2} \text{sech}(\frac{\pi}{2} t) + \frac{1}{2} \tanh(\frac{\pi}{2} t) \, \text{sech}(\frac{\pi}{2} t) \\
&= \frac{1}{2} \text{sech}(\frac{\pi}{2} t) \left(-1 + \tanh(\frac{\pi}{2} t) \right) < 0 \qquad \forall t \in \mathbb{R}.
\end{aligned}$$

  This implies that, for each $t \in \mathbb{R}$,

$$\Phi^c(t) > \frac{1}{\pi} \text{sech}(\frac{\pi}{2} t).$$

  Using this inequality, we obtain

$$\zeta = \frac{\text{Noise}(-\beta F)}{\Phi_{\text{Noise}}(-\beta F)} = \frac{\frac{1}{2} \text{sech}(-\frac{\pi}{2} \beta F)}{\Phi^c_{\text{Noise}}(\beta F)} \leq \frac{\pi}{2}.$$

  Furthermore,

$$\begin{aligned}
T &= -\log \Phi_{\text{Normal}}(-\beta F) \leq -\log \frac{1}{\pi} \text{sech}(\frac{\pi}{2} \beta F) \\
&= \log \pi + \log \cosh(\frac{\pi}{2} \beta F) = O(\beta F).
\end{aligned}$$

Appendix F. Connection Between $L_1(f)$ Norm and Spectral Norm

In Theorem 7 and Corollary 8, we study the connection between $L_f(f)$ norm and the sample complexity of our problem. However, as the $L_1(f)$-metric is dependent on the distribution $f(\boldsymbol{z})$, which is unavoidable, it is not very intuitive. We indeed prefer some more informative norms such as spectral norm. However, to this end, we must restrict the distribution $f(\boldsymbol{z})$.

- We here assume that there is a constant $c > 0$, such that $f(\boldsymbol{z}) \geq c$ for each $\boldsymbol{z} \in B^d(1)$; recall we assume almost surely $\|\boldsymbol{z}\|^2 \leq F = 1$ in this section, and $B^d(1) = \{\boldsymbol{z} \in \mathbb{R}^d \mid \|\boldsymbol{z}\| \leq 1\}$.

We next prove a statement similar to Corollary 8 but in terms of the d-metric instead of the $L_1(f)$-metric. The following definition is also needed for the following two results. For $0 \leq a < b \leq 1$, where z_1 is the first coordinate of $\boldsymbol{z}$, define

$$\mathrm{Cone}(a,b) = \left\{\boldsymbol{z} = (z_1, \ldots, z_d) \mid 3z_1^2 \geq 2\|\boldsymbol{z}\|_2^2 \text{ and } a \leq z_1 \leq b\right\}. \tag{15}$$

Lemma 20 *If $f(\boldsymbol{z}) \geq c > 0$ for each $\boldsymbol{z} \in B^d(1)$, then for all $(\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B]$*

$$\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} \geq \frac{c\pi^{d/2}}{20\Gamma(d/2+1)}(\frac{1}{18})^d \mathrm{d}\left((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)\right).$$

In particular, if $f(\boldsymbol{z})$ is uniform on unit disk, then for all $(\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B]$

$$\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} \geq \frac{1}{20}(\frac{1}{18})^d \mathrm{d}\left((\boldsymbol{M}, \tau), (\boldsymbol{M}^*, \tau^*)\right).$$

Proof We remind that

$$\begin{aligned}
\|(\boldsymbol{M}, \tau) - (\boldsymbol{M}^*, \tau^*)\|_{L_1(f)} &= \int f(\boldsymbol{z}) \left|(\|\boldsymbol{z}\|_{\boldsymbol{M}}^2 - \tau) - (\|\boldsymbol{z}\|_{\boldsymbol{M}^*}^2 - \tau^*)\right| \mathrm{d}\boldsymbol{z} \\
&= \int f(\boldsymbol{z}) \Big|\boldsymbol{z}^\top (\underbrace{\hat{\boldsymbol{M}} - \boldsymbol{M}^*}_{=\bar{\boldsymbol{M}}})\boldsymbol{z} - \underbrace{(\tau - \tau^*)}_{=\bar{\tau}}\Big| \mathrm{d}\boldsymbol{z} \\
&= \int f(\boldsymbol{z}) \left|\boldsymbol{z}^\top \bar{\boldsymbol{M}} \boldsymbol{z} - \bar{\tau}\right| \mathrm{d}\boldsymbol{z}.
\end{aligned}$$

Note that $\bar{\boldsymbol{M}}$ is a symmetric matrix. So, there are a real value orthonormal matrix $\boldsymbol{U}$ and a real value diagonal matrix $\boldsymbol{D}$ such that $\bar{\boldsymbol{M}} = \boldsymbol{U}^\top \boldsymbol{D} \boldsymbol{U}$. Let the vector $\boldsymbol{\lambda}$ denote the diagonal of $\boldsymbol{D}$. Without loss of generality, we assume that $\lambda_1 = \max\{|\lambda_1|, \ldots, |\lambda_d|\}$. One can verify that $\lambda_1 = \|\bar{\boldsymbol{M}}\|_2$. As $\boldsymbol{U}$ is orthonormal, we have

$$\begin{aligned}
\int f(\boldsymbol{z}) \left|\boldsymbol{z}^\top \bar{\boldsymbol{M}} \boldsymbol{z} - \bar{\tau}\right| \mathrm{d}\boldsymbol{z} &= \int f(\boldsymbol{z}) \left|(\boldsymbol{U}\boldsymbol{z})^\top \boldsymbol{D} (\boldsymbol{U}\boldsymbol{z}) - \bar{\tau}\right| \mathrm{d}\boldsymbol{z} \\
&= \int f(\boldsymbol{U}^\top \boldsymbol{z}) \left|\boldsymbol{z}^\top \boldsymbol{D} \boldsymbol{z} - \bar{\tau}\right| \mathrm{d}\boldsymbol{z} \\
&= \int f(\boldsymbol{U}^\top \boldsymbol{z}) \left|\sum_{i=1}^d z_i^2 \lambda_i - \bar{\tau}\right| \mathrm{d}\boldsymbol{z}
\end{aligned}$$

Note that $\sum_{i=1}^d z_i^2\lambda_i \le \lambda_1\|\boldsymbol{z}\|^2$ and, for each $\boldsymbol{z} \in \mathrm{Cone}(0,1)$,

$$\begin{aligned}\sum_{i=1}^d z_i^2\lambda_i &\ge \lambda_1 z_1^2 - \sum_{i=2}^d \lambda_1 z_i^2 \\ &= \lambda_1\left(2z_1^2 - \sum_{i=1}^d z_i^2\right) \\ &= \lambda_1(2z_1^2 - \|\boldsymbol{z}\|_2^2) \ge \frac{\lambda_1}{2}z_1^2\end{aligned}$$

Set $q = \lambda_1 + |\bar{\tau}| = \mathrm{d}\left((\bar{\boldsymbol{M}}, \bar{\tau}), (\boldsymbol{M}^*, \tau^*)\right)$. We next consider two cases based on $|\bar{\tau}|$ and q.

- $|\bar{\tau}| \le 0.1q$. It implies $\lambda_1 \ge 0.9q$ and thus, if $\boldsymbol{z} \in \mathrm{Cone}(\frac{1}{\sqrt{3}}, 1)$, then

$$\begin{aligned}\sum_{i=1}^d z_i^2\lambda_i - \bar{\tau} &\ge \frac{\lambda_1}{2}z_1^2 - \bar{\tau} \\ &\ge q\left(\frac{9}{20}z_1^2 - \frac{1}{10}\right) \\ &\ge \frac{1}{20}q.\end{aligned}$$

  Therefore,

$$\begin{aligned}\int f(\boldsymbol{U}^\top\boldsymbol{z})\Big|\sum_{i=1}^d z_i^2\lambda_i - \bar{\tau}\Big|\mathrm{d}\boldsymbol{z} &\ge \frac{q}{20}\int_{\boldsymbol{z}\in\mathrm{Cone}(\frac{1}{\sqrt{3}},1)} f(\boldsymbol{U}^\top\boldsymbol{z})\mathrm{d}\boldsymbol{z} \\ &= \frac{q}{20}\mu_f(\boldsymbol{U}^\top\mathrm{Cone}(\frac{1}{\sqrt{3}},1)) \qquad (16) \\ &\ge \frac{cq}{20}\int_{\boldsymbol{z}\in\mathrm{Cone}(\frac{1}{\sqrt{3}},1)\cap B^d(1)}\mathrm{d}\boldsymbol{z} \\ &= \frac{cq}{20}\times\mathrm{Volume}\left(\mathrm{Cone}(\frac{1}{\sqrt{3}},1)\cap B^d(1)\right) \\ &\ge \frac{cq}{20}\times\mathrm{Volume}\left(\mathrm{Cone}(\frac{1}{\sqrt{3}},\sqrt{\frac{2}{3}})\right) \\ &= \frac{cq}{20d}\left[\sqrt{\frac{2}{3}}V^{d-1}(\sqrt{\frac{1}{3}})) - \sqrt{\frac{1}{3}}V^{d-1}(\sqrt{\frac{1}{6}})\right] \\ &= \frac{cq}{20}\frac{\pi^{\frac{d-1}{2}}}{d3^{\frac{d}{2}}\Gamma(\frac{d+1}{2})}\left[\sqrt{2} - \frac{1}{2^{\frac{d-1}{2}}}\right]. \qquad (17)\end{aligned}$$

- $|\bar{\tau}| > 0.1q$. It implies $\lambda_1 < 0.9q$ and thus

$$\begin{aligned}\left|\bar{\tau} - \sum_{i=1}^{d} z_i^2 \lambda_i\right| &\geq |\bar{\tau}| - \lambda_1 \|\boldsymbol{z}\|_2^2 \\ &\geq q\left(\frac{1}{10} - \frac{9}{10}\|\boldsymbol{z}\|_2^2\right) \\ &\geq \frac{q}{20} \quad \text{if } \|\boldsymbol{z}\|_2^2 \leq \frac{1}{18}\end{aligned}$$

Therefore,

$$\begin{aligned}\int f(\boldsymbol{U}^\top \boldsymbol{z})\Big|\sum_{i=1}^{d} z_i^2\lambda_i - \bar{\tau}\Big|\mathrm{d}\boldsymbol{z} &\geq \frac{q}{20}\int_{\boldsymbol{z}\in B^q(\frac{1}{18})} f(\boldsymbol{U}^\top \boldsymbol{z})\mathrm{d}\boldsymbol{y} \\ &= \frac{q}{20}\mu_f(B^q(\frac{1}{18})) \qquad (18) \\ &\geq \frac{cq}{20} \times \text{Volume}\left(B^q(\frac{1}{18})\right) \\ &= \frac{cq}{20}\frac{\pi^{d/2}}{\Gamma(d/2+1)}(\frac{1}{18})^d \qquad (19)\end{aligned}$$

Now Lower bounds (17) and (19) implies the proof. ■

If $f(\boldsymbol{z})$ is a rotationally symmetric pdf, then $\mu_f(\boldsymbol{U}^\top B) = \mu_f(B)$ for any measurable set B. Therefore, combining two Lower bounds (16) and (18), we obtain the following lemma

Lemma 21 *If $f(\boldsymbol{z})$ is a rotationally symmetric pdf, then for all $(\boldsymbol{M}, \tau) \in \mathcal{M} \times [0, B]$*

$$\frac{\|(\boldsymbol{M},\tau) - (\boldsymbol{M}^*,\tau^*)\|_{L_1(f)}}{\mathrm{d}\left((\boldsymbol{M},\tau),(\boldsymbol{M}^*,\tau^*)\right)} \geq \frac{1}{20}\max\left(\mu_f(\mathrm{Cone}(\frac{1}{\sqrt{3}},1), \mu_f(B^d(\frac{1}{18}))\right).$$

Appendix G. Further Experimental Study

This section can be seen as a complementary section for Section 5.

G.1 Loss Function Behavior

In Subsection 5.2, we experimentally studied the Logistic model with different noises. In Figures 2 and 3, we evaluate the model for different noise in terms of eigenvalue recover and the accuracies. As a complementary information to these figures, in Figure 9, as the iteration increases, we compare the values of the loss function R_N on $(\hat{\boldsymbol{M}}, \hat{\tau})$, compared with $(\boldsymbol{M}^*/s, \tau^*/s)$ which we do not expect to surpass. Observe that the loss on $(\boldsymbol{M}^*/s, \tau^*/s)$ is a constant red line at the bottom at around 0.23. When considering Logistic noise (blue) we reach this loss around 700 iterations, and nearly do when considering Gaussian noise. For other types of noise, the method does worse; Noisy labeling only achieves a loss value around 0.5.

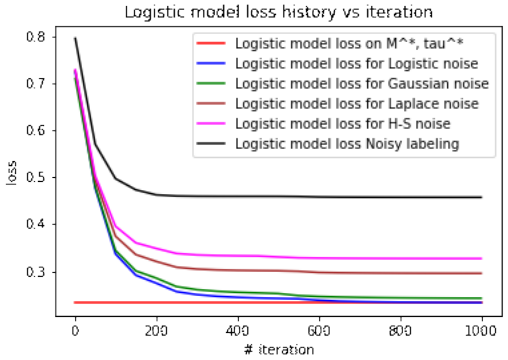

Figure 9: Loss history for Logistic model and different noises.

G.2 Larger Dimension

In Section 5, we dealt with small value for dimension d ($d = 10$ for synthetic and $d = 9, 24$ for real data). In this section, following the same approach as in Subsection 5.1, we generate synthetic data with $d = 100$ and $\text{rank}(\boldsymbol{M}^*) = 30$. We also set the level of noise at 20%. In Figure 10, we observe that how the sample complexity can be affected by the dimension. When the sample size is less than $30K$, the model overfits, which is completely natural as our model has $d^2 + 1$ parameters. But for the larger values, we can see that the model starts to neutralize the noise and the non-noisy accuracy approaches to 1 (blue and magenta curves). We have $97.59\%, 97.56\%$ non-noisy train and test accuracy for $600K$ sample size.

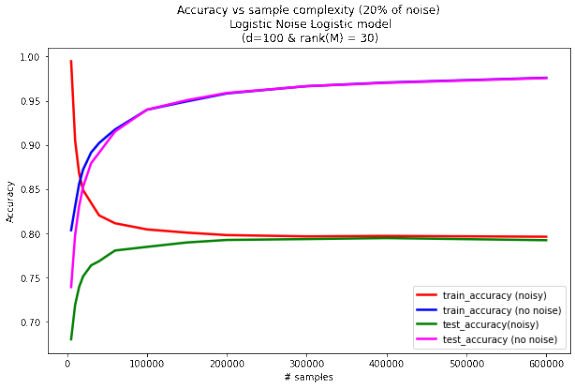

Figure 10: Sample complexity for $d = 100$.

G.3 Unbalanced Labeling

In prior experiments, the parameters are set to ensure a balanced data set, which happens often in real world scenarios. Here we experimentally study the robustness of our model against unbalanced data sets generated according to the same data generation schema explained in Section 5.1. Then we gradually increase τ^* from 0.1 to 6.1 (for 30 values) and record the performance of our model in predicting the ground truth no-noisy labels for each of the classes of Far and Close pairs separately.

The number of generated pairs is 60,000, of which 20,000 are reserved for testing. At the starting state with $\tau^* = 0.1$, we have around 5% of pairs as Close (the rests are Far) while at the end with $\tau^* = 6.1$, around 98% of the pairs are labeled Close. Because of noise, we cannot expect all the pairs to have the same label for any positive τ^*. The results are shown in Figure 11. We can see that the accuracy of the model when measured on the whole data set (see magenta curve) is always very good, irrespective of the distribution of Close and Far pairs. The performance of the model on Far pairs (green curve) in the worst case drops to 93% for the test set. The model on Close pairs (blue curve) drops to 78% at the worst performance. When there are Close pairs are between about 10% and 98% the algorithm recovers above 90% accuracy on all labeled subsets of the data.

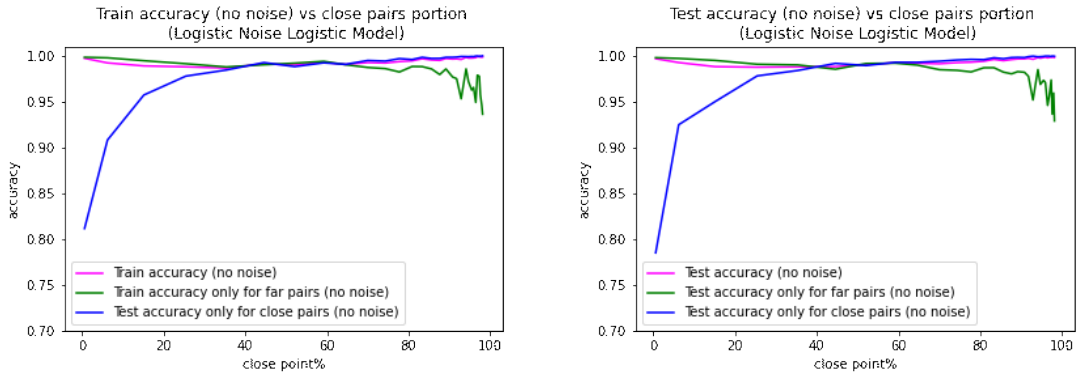

Figure 11: Performance of Logistic noise Logistic model with unbalanced data.

References

Airline Passenger Satisfaction, 2020. URL https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction.

Meysam Alishahi, Anna Little, and Jeff M. Phillips. Linear Distance Metric Learning. https://github.com/meysamalishahi/Linear-Distance-Metric-Learning, 2023. GitHub repository.

Miguel F. Anjos and Henry Wolkowicz. Geometry of semidefinite max-cut relaxations via matrix ranks. *Journal of Combinatorial Optimization*, 6(3):237–270, 2002. doi: 10.1023/A:1014895808844. URL https://doi.org/10.1023/A:1014895808844.

Maria-Florina Balcan, Avrim Blum, and Nathan Srebro. Improved guarantees for learning via similarity functions. In Rocco A. Servedio and Tong Zhang, editors, *COLT*, pages 287–298. Omnipress, 2008. URL http://dblp.uni-trier.de/db/conf/colt/colt2008.html#BalcanBS08.

Laura Balzano, Robert Nowak, and Benjamin Recht. Online identification and tracking of subspaces from highly incomplete information. In *2010 48th Annual allerton conference on communication, control, and computing (Allerton)*, pages 704–711. IEEE, 2010.

Aharon Bar-Hillel, Tomer Hertz, Noam Shental, and Daphna Weinshall. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6(32):937–965, 2005. URL http://jmlr.org/papers/v6/bar-hillel05a.html.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. Learning good edit similarities with generalization guarantees. In Dimitrios Gunopulos, Thomas Hofmann, Donato Malerba, and Michalis Vazirgiannis, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 188–203, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. Good edit similarity learning by loss minimization. *Machine Learning*, 89(1):5–35, 2012a. doi: 10.1007/s10994-012-5293-8. URL https://doi.org/10.1007/s10994-012-5293-8.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. Similarity learning for provably accurate sparse linear classification. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012b. URL http://icml.cc/2012/papers/919.pdf.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. *Metric Learning*. Morgan & Claypool Publishers, 2015.

Aurélien Bellet and Amaury Habrard. Robustness and generalization for metric learning. *Neurocomputing*, 151:259–267, 2015. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2014.09.044. URL https://www.sciencedirect.com/science/article/pii/S092523121401248X.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. A survey on metric learning for feature vectors and structured data. *CoRR*, abs/1306.6709, 2013. URL http://dblp.uni-trier.de/db/journals/corr/corr1306.html#BelletHS13.

Wei Bian and Dacheng Tao. Learning a distance metric by empirical loss minimization. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 1186–1191. IJCAI/AAAI, 2011. doi: 10.5591/978-1-57735-516-8/IJCAI11-202. URL https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-202.

Wei Bian and Dacheng Tao. Constrained empirical risk minimization framework for distance metric learning. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1194–1205, 2012. doi: 10.1109/TNNLS.2012.2198075.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526, 2002. ISSN ISSN 1533-7928. URL http://www.jmlr.org/papers/v2/bousquet02a.html.

David R. Brillinger. Information and Information Stability of Random Variables and Processes. *Journal of the Royal Statistical Society Series C*, 13(2):134–135, June 1964. doi: 10.2307/2985711. URL https://ideas.repec.org/a/bla/jorssc/v13y1964i2p134-135.html.

Qiong Cao, Zheng-Chu Guo, and Yiming Ying. Generalization bounds for metric and similarity learning. *Machine Learning*, 102(1):115–132, 2016. doi: 10.1007/s10994-015-5499-7. URL https://doi.org/10.1007/s10994-015-5499-7.

S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546 vol. 1, 2005. doi: 10.1109/CVPR.2005.202.

Stephan Clémençon, Igor Colin, and Aurélien Bellet. Scaling-up empirical risk minimization: Optimization of incomplete u-statistics. *Journal of Machine Learning Research*, 17(76):1–36, 2016. URL http://jmlr.org/papers/v17/15-012.html.

John D. Cook. Upper and lower bounds for the normal distribution function. URL https://www.johndcook.com/blog/norm-dist-bounds/.

A. Ermolov, L. Mirvakhabova, V. Khrulkov, N. Sebe, and I. Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7399–7409, Los Alamitos, CA, USA, jun 2022. IEEE Computer Society. doi: 10.1109/CVPR52688.2022.00726. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.00726.

Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004. URL https://proceedings.neurips.cc/paper_files/paper/2004/file/42fe880812925e520249e808937738d2-Paper.pdf.

Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Is that you? metric learning approaches for face identification. In *2009 IEEE 12th International Conference on Computer Vision*, pages 498–505, 2009. doi: 10.1109/ICCV.2009.5459197.

Zheng-Chu Guo and Yiming Ying. Guaranteed Classification via Regularized Similarity Learning. *Neural Computation*, 26(3):497–522, 03 2014. ISSN 0899-7667. doi: 10.1162/NECO_a_00556. URL https://doi.org/10.1162/NECO_a_00556.

Mike Hansen, Elizabeth Armstrong, James Sutherland, Josh McConnell, John Hewson, and Robert Knaus. Spitfire. https://github.com/sandialabs/Spitfire, 2020.

Rong Jin, Shijun Wang, and Yang Zhou. Regularized distance metric learning:theory and algorithm. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper_files/paper/2009/file/a666587afda6e89aec274a3657558a27-Paper.pdf.

M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre. Low-rank optimization on the cone of positive semidefinite matrices. *SIAM Journal on Optimization*, 20(5):2327–2351, 2010. doi: 10.1137/080731359. URL https://doi.org/10.1137/080731359.

Brian Kulis. Metric learning: A survey. *Foundations and Trends® in Machine Learning*, 5(4): 287–364, 2013. ISSN 1935-8237. doi: 10.1561/2200000019. URL http://dx.doi.org/10.1561/2200000019.

Adrian S. Lewis and Michael L. Overton. Eigenvalue optimization. *Acta Numerica*, 5:149–190, 1996. doi: 10.1017/S0962492900002646.

Jianan Li, Yunchao Wei, Xiaodan Liang, Fang Zhao, Jianshu Li, Tingfa Xu, and Jiashi Feng. Deep attribute-preserving metric learning for natural language object retrieval. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 181–189, 2017.

Han Liu, Zhizhong Han, Yu-Shen Liu, and Ming Gu. Fast low-rank metric learning for large-scale and high-dimensional data. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/0d0fd7c6e093f7b804fa0150b875b868-Paper.pdf.

S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.R. Mullers. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No.98TH8468)*, pages 41–48, 1999. doi: 10.1109/NNSP.1999.788121.

Kenta Mikawa, Takashi Ishida, and Masayuki Goto. A proposal of extended cosine measure for distance metric learning in text classification. In *2011 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1741–1746, 2011. doi: 10.1109/ICSMC.2011.6083923.

Yadong Mu. Fixed-rank supervised metric learning on riemannian manifold. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Feb. 2016. doi: 10.1609/aaai.v30i1.10246. URL https://ojs.aaai.org/index.php/AAAI/article/view/10246.

Fatemeh Nargesian, Horst Samulowitz, Udayan Khurana, Elias B Khalil, and Deepak S Turaga. Learning feature engineering for classification. In *IJCAI*, volume 17, pages 2529–2535, 2017.

Bac Nguyen, Carlos Morell, and Bernard De Baets. Supervised distance metric learning through maximization of the jeffrey divergence. *Pattern Recognition*, 64:215–225, 2017. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2016.11.010. URL https://www.sciencedirect.com/science/article/pii/S0031320316303600.

Michael L. Overton. On minimizing the maximum eigenvalue of a symmetric matrix. *SIAM Journal on Matrix Analysis and Applications*, 9(2):256–268, 1988. doi: 10.1137/0609021. URL https://doi.org/10.1137/0609021.

Yash Patel, Giorgos Tolias, and Jiri Matas. Recall@k surrogate loss with large batches and similarity mixup. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7492–7501, 2021.

Elias Ramzi, Nicolas Audebert, Nicolas Thome, Clément Rambour, and Xavier Bitot. Hierarchical Average Precision Training for Pertinent Image Retrieval. In *ECCV 2022*, Tel-Aviv, Israel, October 2022. URL https://hal.science/hal-03712933.

Volker Roth and Volker Steinhage. Nonlinear discriminant analysis using kernel functions. *Advances in neural information processing systems*, 12, 1999.

Zakia Salod and Yashik Singh. A five-year (2015 to 2019) analysis of studies focused on breast cancer prediction using machine learning: A systematic review and bibliometric analysis. *J Public Health Res*, 9(1):1792, Jun 2020. ISSN 2279-9028 (Print); 2279-9036 (Electronic); 2279-9028 (Linking). doi: 10.4081/jphr.2020.1772.

Qinfeng Shi, James Petterson, Gideon Dror, John Langford, Alex Smola, and SVN Vishwanathan. Hash kernels for structured data. *Journal of Machine Learning Research*, 10(11), 2009.

Ankita Shukla and Saket Anand. *Distance Metric Learning by Optimization on the Stiefel Manifold*. 01 2015. doi: 10.5244/C.29.DIFFCV.7.

Masashi Sugiyama. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *Journal of Machine Learning Research*, 8(37):1027–1061, 2007. URL http://jmlr.org/papers/v8/sugiyama07b.html.

James C Sutherland and Alessandro Parente. Combustion modeling using principal component analysis. *Proceedings of the Combustion Institute*, 32(1):1563–1570, 2009.

David M. J. Tax and Robert P. W. Duin. Support vector data description. *Machine Learning*, 54(1): 45–66, 2004. doi: 10.1023/B:MACH.0000008084.60811.49. URL https://doi.org/10.1023/B:MACH.0000008084.60811.49.

Lorenzo Torresani and Kuang-chih Lee. Large margin component analysis. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper/2006/file/dc6a7e655d7e5840e66733e9ee67cc69-Paper.pdf.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition, 2008. ISBN 0387790519.

Bart Vandereycken. Low-rank matrix completion by riemannian optimization. *SIAM Journal on Optimization*, 23(2):1214–1236, 2013.

Fei Wang and Jimeng Sun. Survey on distance metric learning and dimensionality reduction in data mining. *Data Mining and Knowledge Discovery*, 29(2):534–564, 2015.

Fei Wang and Changshui Zhang. Feature extraction by maximizing the average neighborhood margin. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. doi: 10.1109/CVPR.2007.383124.

Zhe Wang, Daqi Gao, and Zhisong Pan. An effective support vector data description with relevant metric learning. In Liqing Zhang, Bao-Liang Lu, and James Kwok, editors, *Advances in Neural Networks - ISNN 2010*, pages 42–51, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-13318-3.

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(9):207–244, 2009. URL http://jmlr.org/papers/v10/weinberger09a.html.

Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1):397–434, 2013. doi: 10.1007/s10107-012-0584-1. URL https://doi.org/10.1007/s10107-012-0584-1.

William Wolberg, Olvi Mangasarian, Nick Street, and W. Street. Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository, 1995. DOI: https://doi.org/10.24432/C5DW2B.

Shiming Xiang, Feiping Nie, and Changshui Zhang. Learning a mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, 41(12):3600–3612, 2008. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2008.05.018. URL https://www.sciencedirect.com/science/article/pii/S0031320308002057.

Eric Xing, Michael Jordan, Stuart J Russell, and Andrew Ng. Distance metric learning with application to clustering with side-information. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002. URL https://proceedings.neurips.cc/paper_files/paper/2002/file/c3e4035af2a1cde9f21e1ae1951ac80b-Paper.pdf.

Huan Xu and Shie Mannor. Robustness and generalization. *Machine Learning*, 86(3):391–423, 2012. doi: 10.1007/s10994-011-5268-1. URL https://doi.org/10.1007/s10994-011-5268-1.

Yiming Ying and Peng Li. Distance metric learning with eigenvalue optimization. *The Journal of Machine Learning Research*, 13:1–26, 2012.

Kamila Zdybał, James C. Sutherland, and Alessandro Parente. Manifold-informed state vector subset for reduced-order modeling. *Proceedings of the Combustion Institute*, 2022. ISSN 1540-7489. doi: https://doi.org/10.1016/j.proci.2022.06.019. URL https://www.sciencedirect.com/science/article/pii/S1540748922000153.