Generalization Bound of Gradient Descent for Non-Convex Metric Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Metric learning aims to learn a distance measure that can benefit distance-based methods such as the nearest neighbor (NN) classifier. While considerable efforts have been made to improve its empirical performance and analyze its generalization ability by focusing on the data structure and model complexity, an unresolved question is how choices of algorithmic parameters such as training time affect metric learning as it is typically formulated as an optimization problem and nowadays more often as a non-convex problem. In this paper, we theoretically address this question and prove the agnostic Probably Approximately Correct (PAC) learnability for metric learning algorithms with non-convex objective functions optimized via gradient descent (GD); in particular, our theoretical guarantee takes training time into account. We first show that the generalization PAC bound is a sufficient condition for agnostic PAC learnability and this bound can be obtained by ensuring the uniform convergence on a densely concentrated subset of the parameter space. We then show that, for classifiers optimized via GD, their generalizability can be guaranteed if the classifier and loss function are both Lipschitz smooth, and further improved by using fewer iterations. To illustrate and exploit the theoretical findings, we finally propose a novel metric learning method called *S*mooth *M*etric and representative *I*nstance *LE*arning (SMILE), designed to satisfy the Lipschitz smoothness property and learned via GD with an early stopping mechanism for better discriminability and less computational cost of NN.

1 Introduction

A good measure of distance between instances is important to many machine learning algorithms, such as the nearest neighbor (NN) classifier and k-means clustering. As it is difficult to handcraft an optimal distance for each task, metric learning appears as an appealing technique to learn the distance metric automatically and directly from the data. The most widely studied metric is the Mahalanobis distance and it is often learned as an optimization problem [46, 16, 44]. To enhance the discriminability of the learned metric, various loss functions have been designed, considering the local property of heterogeneous data [14, 42, 21, 4, 33, 49, 39, 11] and the nonlinear geometry of the sample space [22, 53, 7]. Meanwhile, to achieve good generalization and robustness, different regularizations have been imposed to control the model complexity [27, 24, 45, and references therein]. In addition to methodological advances, theoretical guarantees of metric learning algorithms, as well as guarantees of metric-based classifiers [2, 18], have been provided. In particular, generalization bounds have been founded on the complexity measure of the model class [50, 3, 6, 41, 29, 48], algorithmic stability [25, 18, 15], and algorithmic robustness [1]. The intrinsic complexity of the dataset has also been considered in recent works [41, 29].

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

While the data structure and model complexity play a vital role in metric learning, an equally important but as yet poorly understood factor is the choice of optimization algorithms and the associated parameters [37]. For example, when metric learning is formulated as a non-convex problem and optimized by using the gradient descent algorithm, its solution is inevitably influenced by factors such as the learning rate and the number of training iterations; different local minima will then exhibit different generalization behavior.

Therefore, the goal of this paper is to provide a new route to theoretical exploration and exploitation of the effect of the gradient descent (GD) algorithm on metric learning methods. To this end, we provide a generalization bound which suggests that early stopping, smooth classifier and smooth loss function have crucial influence on the generalization error. We highlight that our results are obtained without using any property of convex optimization, and hence are applicable to non-convex metric learning methods. The contributions of this paper are fourfold.

1. We show that the generalization Probably Approximately Correct (PAC) bound, which is a weaker notion than the uniform convergence condition, is a sufficient condition for a parametric hypothesis class to be agnostic PAC learnable (Theorem 1).

2. To facilitate the derivation of the generalization PAC bound of a hypothesis class, we propose a new decomposition theorem to decompose the bound into two terms that can be easily guaranteed (Theorem 2). The first term constrains the space of the estimated parameters of the hypothesis, reducing it from the entire parameter space to a high-confidence subset of the parameter space. The second term considers the uniform convergence condition of the concentrated subset.

3. Based on the decomposition theorem, we obtain the generalization PAC bound for classifiers learned with the gradient descent algorithm (Theorem 3). The bound shows that the generalization gap increases over iterations, thus providing a theoretical support for the practical use of early stopping. Moreover, it shows that a Lipschitz smooth (i.e. Lipschitz continuous of the gradient) classifier and a Lipschitz smooth loss function are necessary for generalization guarantee.

4. We propose a novel metric learning method as a concrete example of using the generalization PAC bound. When classifying a test instance, the NN classifier has to store the entire training set and calculate its distances to all training instances, thereby incurring high storage and computational costs. To reduce these costs and improve the generalization performance, we propose to simultaneously learn the distance metric and few representative instances which serve as the reference points for testing; the new method is called *S*mooth *M*etric and representative *I*nstance *LEearning* (SMILE). More specific, to ensure good test performance, SMILE adopts a Lipschitz smooth classifier and loss function and is optimized via GD with a designed early stopping mechanism. The method is evaluated on 12 datasets and shows competitive performance against existing methods.

1.1 Related work

Generalization bound of GD with early stopping Early stopping in regularizing the model complexity and its effect on the generalization ability have been extensively studied for a wide range of methods, such as perceptron algorithm [8], kernel regression [47], and deep neural networks [31]. Our algorithm-dependent PAC bound is motivated by [20], which proves the generalizability for models learned with stochastic GD. The main difference between [20] and our work is that [20] studies the expected generalization gap, which is not a sufficient condition for agnostic PAC learnablility, whereas the generalization PAC bound studied in this paper is a sufficient condition. Consequently, we need a new decomposition theorem so that the generalization PAC bound can be used to analyze models learned with GD.

Generalization bounds for Lipschitz classifiers and losses [28, 17] use Lipschitz functions as large margin classifiers in general metric spaces and provide generalization bounds for Lipschitz classifiers. Our theoretical guarantee is different from their work in two aspects. First, the input space of the Lipschitz constant is the data space in [28, 17], whereas the input space is the parameter space in our paper. Second, owing to this difference, the generalization bound obtained in our work has a faster convergence in most cases. [41] derives the generalization bound for metric learning algorithms with Lipschitz continuous loss functions. However, when taking the influence of GD into account, Lipschitz continuity is not sufficient to guarantee the generalizability; Lipschitz smoothness is also needed. [48] makes use of a smooth loss function to obtain a fast generalization. However, their work requires the objective function to be strongly convex, which is different from our focus on non-convex problems.

Metric learning with representative instances Reducing the amount of necessary training data as a way of reducing the storage and computational costs of NN has been extensively studied, e.g. in [30, 52, 35]. Among these methods, SNC [26] and ProtoNN [19] are the most relevant to our work, as they also learn the distance metric and representative instances simultaneously. Our method differs from them in the loss function and regularization terms, both of which are designed in our work to provide a theoretical guarantee on the classification performance.

2 Preliminaries

2.1 Notation

This paper focuses on binary classification problems. Let $\boldsymbol{z}^n = \{\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i), i = 1, \ldots, n\} \in \mathcal{Z}^n$ denote the set of n independent and identically distributed (i.i.d.) training instance and label pairs, sampled from an unknown joint distribution $p(\boldsymbol{z}) = p(\boldsymbol{x}, y)$. Let $h(\boldsymbol{x}, \boldsymbol{w})$ be a function with instance $\boldsymbol{x}$ and parameter $\boldsymbol{w} \in \mathcal{W} \subseteq \mathbb{R}^Q$. The output of $h(\boldsymbol{x}, \boldsymbol{w})$ is restricted to be a real value; $\mathrm{sign}\,[h(\boldsymbol{x}, \boldsymbol{w})]$ returns the classification decision, where $\mathrm{sign}[\cdot]$ denotes the sign function.

During the training process of the classifier, given $\boldsymbol{z}^n$, a *classifier* or *hypothesis* $\hat{h}$ can be obtained from an optimization algorithm, such as GD. $R_n(\boldsymbol{z}^n, \hat{h}) := \frac{1}{n}\sum_i r(\boldsymbol{z}_i, \hat{h}) := \frac{1}{n}\sum_i l(\hat{h}(\boldsymbol{x}_i), y_i)$ is called the *training error*, where $r(\cdot,\cdot)$ denotes the risk function and $l(\cdot,\cdot)$ denotes the loss function. Let $\boldsymbol{s} \in \mathcal{S}$ denote a fixed setting of the algorithm, including e.g. the initial values, the number of iterations and the learning rate. With a parametric classifier, $\hat{h}$ can be fully represented by $\hat{\boldsymbol{w}}$. The relationship between $\boldsymbol{w}$ and $\boldsymbol{z}^n$ is represented as $\hat{\boldsymbol{w}} = \boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s})$, where $\boldsymbol{m} : \mathcal{Z}^n \times \mathcal{S} \to \mathcal{W}$; $\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s})$ will sometimes be abbreviated to $\boldsymbol{m}(\boldsymbol{z}^n)$ for notational simplicity. Since $\hat{\boldsymbol{w}}$ is a function of random samples $\boldsymbol{z}^n$, $\hat{\boldsymbol{w}}$ is also a random variable. In the parametric case, the training error will be represented as $R_n(\boldsymbol{z}^n, \hat{\boldsymbol{w}}) := \frac{1}{n}\sum_i r(\boldsymbol{z}_i, \hat{\boldsymbol{w}}) := \frac{1}{n}\sum_i l(h(\boldsymbol{x}_i, \hat{\boldsymbol{w}}), y_i)$.

During the test process, a test pair $\boldsymbol{z}' = (\boldsymbol{x}', y')$ is sampled from the same unknown distribution $p(\boldsymbol{z})$. The predicted value $h(\boldsymbol{x}', \hat{h})$ will be compared with the true label y' to evaluate the performance of the algorithm. $R(\hat{h}) := \mathbb{E}_{\boldsymbol{z}'} r(\boldsymbol{z}', \hat{h}) := \mathbb{E}_{\boldsymbol{z}'} l(h(\boldsymbol{x}', \hat{h}), y')$ is called the *test error*. With a parametric classifier, the following notations will be used $R(\hat{\boldsymbol{w}}) := \mathbb{E}_{\boldsymbol{z}'} r(\boldsymbol{z}', \hat{\boldsymbol{w}}) := \mathbb{E}_{\boldsymbol{z}'} l(h(\boldsymbol{x}', \hat{\boldsymbol{w}}), y')$.

The gap between the training error and the test error, $R(\hat{\boldsymbol{w}}) - R_n(\boldsymbol{z}^n, \hat{\boldsymbol{w}})$, is called the *generalization gap*. A good classifier should have small training error and small generalization gap so as to perform well on test instances.

Let $\|\boldsymbol{a}\|_2$ denote the L_2-norm of a vector $\boldsymbol{a}$ and $\|\boldsymbol{A}\|_F$ denote the Frobenius norm of a matrix $\boldsymbol{A}$. The subscript of norm will be dropped when it is clear from the context. $\boldsymbol{a}_{[q]}$ denotes the qth element of a vector $\boldsymbol{a}$.

2.2 Definitions

Definition 1. [43] Let (Θ, ρ_Θ), $(\mathcal{V}, \rho_\mathcal{V})$ be two metric spaces. A function $h : \Theta \to \mathcal{V}$ is called *Lipschitz continuous* if $\exists L < \infty, \forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \Theta$,

$$\rho_\mathcal{V}(h(\boldsymbol{\theta}_1), h(\boldsymbol{\theta}_2)) \le L\rho_\Theta(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2).$$

The *Lipschitz constant* of h with respect to the input space Θ, denoted by $\mathrm{lip}(h; \mathcal{V} \leftarrow \Theta)$ or $\mathrm{lip}(h \leftarrow \boldsymbol{\theta})$ for short, is the smallest L such that the above inequality holds.

Definition 2. A function $r : \Theta \to \mathbb{R}$ is called *Lipschitz smooth*, if $\exists \eta < \infty, \forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \Theta$,

$$\|\nabla r(\boldsymbol{\theta}_1) - \nabla r(\boldsymbol{\theta}_2)\| \le \eta \|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|.$$

The Lipschitz constant of the derivative of r with respect to Θ, denoted by $\mathrm{lip}(\frac{\partial r}{\partial \boldsymbol{\theta}} \leftarrow \boldsymbol{\theta})$, is the smallest η such that the above inequality holds.

Some properties of Lipschitz functions are frequently used in the paper, such as constructing sophisticated Lipschitz functions from the basic ones and bounding the Lipschitz constant via the gradient of differentiable functions; details are listed in Appendix A.

Definition 3. [43] The *diameter* of a set $\mathcal{V}$ is defined as

$$\mathrm{diam}(\mathcal{V}) = \max_{\boldsymbol{v}_i, \boldsymbol{v}_j \in \mathcal{V}} \|\boldsymbol{v}_i - \boldsymbol{v}_j\|.$$

3 Learnablility via the generalization PAC Bound

In this section, we first introduce the generalization PAC bound and establish its link with the agnostic PAC learnablility. We then propose a decomposition theorem. Finally, we apply the theorem to prove the learnability of the gradient descent algorithm.

3.1 Generalization PAC bound and agnostic PAC learnability

One classical way of determining whether a hypothesis class is agnostic PAC learnable is to verify the uniform convergence condition, which bounds the generalization gap over all hypotheses of the class. However, as some hypotheses are not searched under a fixed setting of the optimization algorithm, [5] proposes to bound the generalization gap for specific algorithms. We adopt this notion and formally define the generalization PAC bound as follows.

Definition 4. A hypothesis class $\mathcal{H}$ has the *generalization PAC bound* if there exists a function $n_{\mathcal{H}}^G : (0,1)^2 \to \mathbb{N}$ such that for every $\epsilon, \delta \in (0,1)$ and for every probability distribution $\mathcal{D}_{\mathcal{Z}}$ over $\mathcal{Z}$, if $\boldsymbol{z}^n$ is a sample of $n \geq n_{\mathcal{H}}^G(\epsilon, \delta)$ i.i.d. examples drawn from $D_{\mathcal{Z}}$, the algorithm returns a hypothesis $\hat{h}$ such that the following inequality is satisfied:

$$\mathbb{P}_{\boldsymbol{z}^n}[R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon] \geq 1 - \delta. \tag{1}$$

First, we note that $\hat{h}$ is regarded as a random variable in this paper. Second, while the generalization PAC bound is a weaker condition than the uniform convergence, as shown in Lemma 1, it is still a sufficient condition for the agnostic PAC learnability, as shown in Theorem 1. Proofs of theorems and lemmas are given in Appendix C.

Lemma 1. The relationship between the generalization PAC bound and the uniform convergence bound is as follows:

$$\mathbb{P}_{\boldsymbol{z}^n}[R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon] \geq \mathbb{P}_{\boldsymbol{z}^n}\Big[\max_{h \in \mathcal{H}} \big(R(h) - R_n(\boldsymbol{z}^n, h)\big) \leq \epsilon\Big]. \tag{2}$$

Theorem 1. Suppose $\mathrm{ERM}_{\mathcal{H}}$ exists for a class $\mathcal{H}$, where $\mathrm{ERM}_{\mathcal{H}}$ denotes the empirical risk minimization learner over the class $\mathcal{H}$. If $\mathcal{H}$ has the generalization PAC bound with a function $n_{\mathcal{H}}^G : (0,1)^2 \to \mathbb{N}$, then $\mathcal{H}$ is agnostic PAC learnable with the sample complexity function $n_{\mathcal{H}}^{AL}(\epsilon, \delta) \leq \max[n_{\mathcal{H}}^G(\epsilon/2, \delta/2), \frac{2C_r^2}{\epsilon^2} \ln \frac{4}{\delta}]$, where the range of the risk function $r(\boldsymbol{z}, h)$ is $[0, C_r]$. Furthermore, in this case, $\mathrm{ERM}_{\mathcal{H}}$ is a successful agnostic PAC learner for $\mathcal{H}$.

3.2 Decomposition theorem for the generalization PAC bound

Directly bounding Eq. 1 is difficult due to the random nature of $\boldsymbol{z}^n$ and $\hat{h}$ in R_n. To disentangle these two quantities, we propose the following decomposition theorem. Its core idea is to use the uniform convergence bound in a much smaller set.

Theorem 2. (Decomposition Theorem) Let $\mathcal{W}$ denote the set of all possible values of $\boldsymbol{w}$ and $\hat{\mathcal{W}} \subseteq \mathcal{W}$; let $\delta_1, \delta_2 \geq 0$. If

$$\mathbb{P}_{\boldsymbol{z}^n}[\hat{\boldsymbol{w}} \in \hat{\mathcal{W}}] \geq 1 - \delta_1 \tag{3}$$

and

$$\mathbb{P}_{\boldsymbol{z}^n}\Big[\max_{\boldsymbol{w} \in \hat{\mathcal{W}}} \big(R(\boldsymbol{w}) - R_n(\boldsymbol{z}^n, \boldsymbol{w})\big) \leq \epsilon\Big] \geq 1 - \delta_2, \tag{4}$$

then the following inequality holds:

$$\mathbb{P}_{\boldsymbol{z}^n}[R(\hat{\boldsymbol{w}}) - R_n(\boldsymbol{z}^n, \hat{\boldsymbol{w}}) \leq \epsilon] \geq 1 - \delta_1 - \delta_2. \tag{5}$$

Theorem 2 decomposes the generalization PAC bound into two terms which are easier to be controlled, namely (i) a smaller parameter space $\hat{\mathcal{W}}$ that includes estimated parameter vectors with high probability; (ii) uniform convergence of $\hat{\mathcal{W}}$. In the following section, the theorem is applied to analyze the generalization ability of the gradient descent algorithm. We show that term (i) can be bounded by applying the concentration inequality to the random variables $\hat{\boldsymbol{w}}$ and term (ii) can be bounded based on the covering number.

3.3 Learnability of the gradient descent algorithm

3.3.1 Settings

The updating equation of the most conventional GD algorithm is as follows:

$$\hat{\boldsymbol{w}}^{(1)} = \boldsymbol{w}^{(0)} - \frac{\alpha^{(1)}}{n}\sum_{i=1}^{n}\frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\hat{\boldsymbol{w}}^{(0)}};$$

$$\vdots$$

$$\begin{aligned}\hat{\boldsymbol{w}}^{(T)} &= \hat{\boldsymbol{w}}^{(T-1)} - \frac{\alpha^{(T)}}{n}\sum_{i=1}^{n}\frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\hat{\boldsymbol{w}}^{(T-1)}} \\ &= \boldsymbol{w}^{(0)} - \sum_{t=1}^{T}\frac{\alpha^{(t)}}{n}\sum_{i=1}^{n}\frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\hat{\boldsymbol{w}}^{(t-1)}},\end{aligned}$$

where $\alpha^{(t)} \geq 0$ denotes the learning rate at iteration t; $\hat{\boldsymbol{w}}^{(t)}$ denotes the estimated parameters of the classifier obtained after t iterations; $\boldsymbol{w}^{(0)}$ denotes the initial parameter of the algorithm. Here the number of iterations T and the learning rate $\alpha^{(t)}$ are treated as the setting parameters of the GD algorithm and determined in advance, i.e. $\boldsymbol{s} = \{T, \alpha^{(t)}, t = 1, \ldots, T\}$. The initial weight $\boldsymbol{w}^{(0)}$ is assumed to be fixed.

3.3.2 Concentration of $\hat{\boldsymbol{w}}^{(T)}$

Recall that $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}) = \hat{\boldsymbol{w}}^{(T)} \in \mathbb{R}^Q$ and $\boldsymbol{m}^{(T)}_{[q]}(\boldsymbol{z}^n; \boldsymbol{s})$ denotes the qth element of $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s})$. To prove that the first term of Theorem 2 holds, we set $\hat{\mathcal{W}}$ as the Euclidean ball centered at $\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s})$ with radius ϵ, denoted by $\mathrm{ball}\left(\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s}), \epsilon\right)$. The condition that $\hat{\boldsymbol{w}} \in \hat{\mathcal{W}}$ with high probability is equivalent to the condition that $\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s}) \in \mathrm{ball}\left(\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s}), \epsilon\right)$ with high probability. With a fixed setting $\boldsymbol{s}$ and any fixed initial parameter vector $\boldsymbol{w}^{(0)}$, given the training samples $\boldsymbol{z}^n$, the value of $\boldsymbol{m}^{(T)}_{[q]}(\boldsymbol{z}^n; \boldsymbol{s})$ is determined. In other words, $\boldsymbol{m}^{(T)}_{[q]}(\boldsymbol{z}^n; \boldsymbol{s})$ is a function from $\mathcal{Z}^n$ to $\mathbb{R}$. By applying the McDiarmid's inequality (Lemma B.1), we obtain the following lemma on the concentration property of $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s})$.

Lemma 2. The following bound holds for any fixed $\boldsymbol{s}$ and $\boldsymbol{w}^{(0)}$:

$$\mathbb{P}_{\boldsymbol{z}^n}\left[\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}) \in \mathrm{ball}\left(\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}), \epsilon\right)\right] \geq 1 - 2Q\exp\left(\frac{-2\epsilon^2 n}{QC^2}\right), \tag{6}$$

where $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}) \in \mathbb{R}^Q$; $C = 2\left(\sum_{t=1}^{T}\eta^{T-t}\alpha^{(t)}\right)\mathrm{lip}\,(r \leftarrow \boldsymbol{w})$; $\eta = \mathrm{lip}(G \leftarrow \boldsymbol{w})$ and $G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) = \boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n) - \frac{\alpha^{(t)}}{n}\sum_{j\in[n]/i}\frac{\partial r(\boldsymbol{z}_j, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)}$; $[n]/i$ denotes the set which contains the integers from 1 to n without i.

The key idea behind the proof is as follows. Randomness of sampling leads to randomness of the learned parameter vector $\hat{\boldsymbol{w}}$. After one iteration of gradient update, the difference between $\hat{\boldsymbol{w}}$ learned on the random samples and that learned on the population is controlled by the Lipschitz constant of r and G. Such differences will accumulate over iterations, thereby affecting the concentration property.

3.3.3 Uniform convergence inside $\hat{\mathcal{W}}$

The following uniform convergence condition is obtained based on the covering number and Dudley's chaining integral [13]. By using the Lipschitz constant, we can bound the covering number of the hypothesis class by the covering number of the parameter space.

Lemma 3. Suppose $\mathrm{lip}(r \leftarrow \boldsymbol{w}) \leq L$ and $\mathrm{diam}(\mathcal{W}) \leq B$, then the following inequality holds:

$$\mathbb{P}_{\boldsymbol{z}^n}\left[\max_{\boldsymbol{w}\in\mathcal{W}}\left(R(\boldsymbol{w}) - R_n(\boldsymbol{z}^n, \boldsymbol{w})\right) \leq CLB\sqrt{\frac{Q}{n}} + \sqrt{\frac{\ln(1/\delta)}{2n}}\right] \geq 1 - \delta, \tag{7}$$

where C is a universal constant.

3.3.4 Application of the decomposition theorem

Theorem 3. Suppose $\text{lip}(h \leftarrow \boldsymbol{w}) \le L_1$ and $\text{lip}(r \leftarrow h) \le L_l$. Then with probability at least $1 - \delta_1 - \delta_2$, the following bound holds:

$$R(\boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s})) - R_n(\boldsymbol{z}^n, \boldsymbol{m}(\boldsymbol{z}^n; \boldsymbol{s})) \le \frac{C_1 C_2 L_1^2 L_l^2 Q \sqrt{\ln(2Q/\delta_1)}}{n} + \sqrt{\frac{\ln(1/\delta_2)}{2n}}, \tag{8}$$

where $\boldsymbol{w} \in \mathbb{R}^Q$; C_1 is a universal constant; $C_2 = \sum_{t=1}^{T} \eta^{T-t} \alpha^{(t)}$, in which T denotes the number of iterations, $\alpha^{(t)}$ denotes the learning rate at time t, $\eta = \text{lip}(G \leftarrow \boldsymbol{w})$, and $G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) = \boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n) - \frac{\alpha^{(t)}}{n} \sum_{j \in [n]/i} \frac{\partial r(\boldsymbol{z}_j, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)}$; $[n]/i$ denotes the set which contains the integers from 1 to n without i.

Theorem 3 suggests that the following factors will affect the generalizability of the learned model. 1) T: A smaller number of iterations leads to better concentration property and thus better generalization performance. Thus, when optimizing via GD, we select the model from the earliest iteration t that yields the minimum training error; the test stage is implemented using the parameters learned at t;
2) Q: A smaller value of Q, i.e. fewer parameters, gives a tighter generalization bound;
3) L_1, L_l: Using a classifier and loss function with smaller Lipschitz constants will improve the generalizability;
4) η: Based on the definition of G and the addition property of Lipschitz functions (Appendix A), if $\text{lip}(\frac{\partial r(\boldsymbol{z}_j, \boldsymbol{w})}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w})$ is bounded by L_s, then η is bounded by $1 + \alpha L_s$. Based on the composition property of Lipschitz functions, we have

$$\text{lip}(\frac{\partial r}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w}) = \text{lip}(\frac{\partial r}{\partial h} \frac{\partial h}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w}) \le \text{lip}(\frac{\partial r}{\partial h} \leftarrow \boldsymbol{w}) \, \text{lip}(\frac{\partial h}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w}).$$

Thus η is bounded if both $\text{lip}(\frac{\partial h}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w})$ and $\text{lip}(\frac{\partial r}{\partial h} \leftarrow h)$ are bounded. In other words, the classifier and loss function should be Lipschitz smooth.

4 Smooth metric and representative instance learning (SMILE)

Theorem 3 shows that Lipschitz smoothness is indispensable for ensuring generalization. To enjoy and illustrate the practical exploitation of this appealing theoretical result, we establish a simple yet theoretically well-founded and new metric learning method called SMILE with a smooth classifier and a smooth loss function. SMILE learns a Mahalanobis distance to enhance the classification performance of NN classifier. Meanwhile, to reduce the storage and computational cost of NN, SMILE learns few representative instances in the training stage and calculate the distances between the test instance and representative instances only in the test stage. In this section, we present the classifier, the loss function, the optimization problem, and some experimental results of SMILE.

4.1 The classifier of SMILE

For any two instances $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, the generalized (squared) Mahalanobis distance is defined as $d^2_{\boldsymbol{M}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T \boldsymbol{M} (\boldsymbol{x}_i - \boldsymbol{x}_j)$, where $\boldsymbol{M}$ is a positive semidefinite (PSD) matrix. Owing to the PSD property, $\boldsymbol{M} = \boldsymbol{L}^T \boldsymbol{L}$ and thus $d^2_{\boldsymbol{M}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = d(\boldsymbol{L}\boldsymbol{x}_i, \boldsymbol{L}\boldsymbol{x}_j) = \|\boldsymbol{L}\boldsymbol{x}_i - \boldsymbol{L}\boldsymbol{x}_j\|_2^2$.

The classifier of SMILE is simply defined as follows:

$$h(\boldsymbol{x}; \boldsymbol{r}^m, \boldsymbol{L}) = \sum_j \exp(-d^2(\boldsymbol{L}\boldsymbol{x}, \boldsymbol{r}_j^+)) - \sum_k \exp(-d^2(\boldsymbol{L}\boldsymbol{x}, \boldsymbol{r}_k^-)), \tag{9}$$

where $\boldsymbol{r}^m$ and $\boldsymbol{L}$ are the parameters of the classifier; $\boldsymbol{r}_j^+$ and $\boldsymbol{r}_k^-$ denote the jth representative instance of the positive class and the kth representative instance of the negative class, respectively; m denotes the total number of learned representative instances. The test instance $\boldsymbol{x}$ is classified to the positive class when $h(\boldsymbol{x}) \ge 0$ and to the negative class when $h(\boldsymbol{x}) < 0$.

As shown in Appendix D, a sufficient condition for h to be Lipschitz smooth is that $\text{diam}(\boldsymbol{L})$, $\text{diam}(\boldsymbol{x})$ and $\text{diam}(\boldsymbol{r})$ are bounded. With a slight abuse of notation, $\text{diam}(\boldsymbol{L})$ denotes the diameter of the set which contains all possible values of $\boldsymbol{L}$; $\text{diam}(\boldsymbol{x})$ and $\text{diam}(\boldsymbol{r})$ are defined similarly. To bound these quantities, we will constrain the Frobenius norm of $\boldsymbol{L}$ and the L_2-norm of $\boldsymbol{x}$ and $\boldsymbol{r}$.

4.2 The loss function of SMILE

Similarly to the Huber loss for regression [23], we propose the following loss function defined by combining a quadratic and a linear function:

$$l(a) = \begin{cases} 1-a & \text{if } a \le 0; \\ \frac{1}{4}(a-2)^2 & \text{if } 0 < a \le 2; \\ 0 & \text{if } a > 2. \end{cases} \tag{10}$$

The derivative of $l(a)$ is as follows:

$$l'(a) = \begin{cases} -1 & \text{if } a \le 0; \\ \frac{a-2}{2} & \text{if } 0 < a \le 2; \\ 0 & \text{if } a > 2. \end{cases}$$

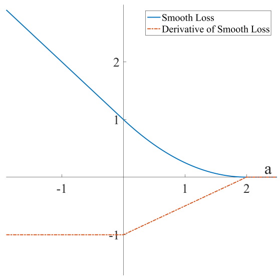

Figure 1: An illustration of the proposed Lipschitz smooth loss function and its derivative.

The loss function and its derivative are illustrated in Figure 1. The Lipschitz constant of $l'(a)$ is $\frac{1}{2}$, meaning that the proposed loss is a Lipschitz smooth function.

4.3 The optimization problem of SMILE

Using the classifier defined in Eq. 9, the loss function defined in Eq. 10, and the convex regularization terms $\sum_j \|\boldsymbol{r}_j^+\|_2^2 + \sum_k \|\boldsymbol{r}_k^-\|_2^2 + \|\boldsymbol{L}\|_F^2$, the following optimization problem is proposed for SMILE:

$$\min_{\Theta} \frac{1}{n} \sum_{i=1}^{n} l(y_i h(\boldsymbol{x}_i; \boldsymbol{r}^m, \boldsymbol{L})) + \lambda \Big(\sum_{j=1}^{m_+} \|\boldsymbol{r}_j^+\|_2^2 + \sum_{k=1}^{m_-} \|\boldsymbol{r}_k^-\|_2^2 + \|\boldsymbol{L}\|_F^2 \Big), \tag{11}$$

where $\Theta = \{\boldsymbol{r}^m, \boldsymbol{L}\}$ denotes the set of parameters to be optimized; $\boldsymbol{r}^m = \{\boldsymbol{r}_j^+, \boldsymbol{r}_k^-; j = 1, \ldots, m_+, k = 1, \ldots, m_-\}$ denotes the set of representative instances with m_+ instances for the positive class and m_- instances for the negative class; and λ is a trade-off parameter balancing the loss term and the regularization term.

The objective function is not convex due to the non-convexity of $h(\boldsymbol{x}; \boldsymbol{r}^m, \boldsymbol{M})$. We apply the gradient descent algorithm to learn the parameters; detailed formulae are given in Appendix D.

4.4 Illustrative results of SMILE

Experimental settings We illustrate the effectiveness of SMILE by comparing it with nine widely adopted metric learning methods: NCA [16], LMNN [44], ITML [9], R2LML [21], SCML [38], RVML [34], GMML [51], DMLMJ [32], and SNC [26]. NCA is implemented by using the drToolbox [40]; LMNN and ITML are implemented by using the metric-learn toolbox [10]; and R2LML, SCML, RVML, GMML, DMLMJ, and SNC are implemented by using the authors' code.

The experiment focuses on binary classification of 12 publicly available datasets from the websites of UCI [12] and Delve [36]. Sample size and feature dimension are listed in Table 1 of Appendix E. All datasets are pre-processed by firstly subtracting the mean and dividing by the standard deviation, and then normalizing the L_2-norm of each instance to 1.

For each dataset, we randomly select 60% of instances to form a training set and the rest are used for testing. This process is repeated 10 times and we report the mean accuracy and the standard deviation. 10-fold cross-validation is used to select the trade-off parameters in the compared algorithms, namely the regularization parameter of LMNN (from $\{0.1, 0.3, \ldots, 0.9\}$), γ in ITML (from $\{0.25, 0.5, 1, 2, 4\}$), λ in RVML (from $\{10^{-5}, 10^{-4}, \ldots, 10\}$), t in GMML (from $\{0.1, 0.3, \ldots, 0.9\}$), and ratio in SNC (from $\{0.01, 0.02, 0.04, 0.08, 0.16\}$). All other parameters are set as default. For the proposed SMILE, the parameters are set as follows: $\boldsymbol{L}$ is initialized as the identity matrix; $\boldsymbol{r}^m$ are initialized as the k-means clustering centers of the positive and negative classes (by using MATLAB *kmeans* function with random initial values); the number of representative instances for each class is set as 2; the trade-off parameter λ is set as 1; and the learning rate α is set as 0.001. The maximum number of iterations is set as 5000 and the final result is based on the parameters at time t, which is the earliest time when the smallest training error is obtained, to conform to early stopping as suggested by Theorem 3.

Table 1: Comparison of classification performances. Mean accuracy and standard deviations are reported with the best ones in bold; '# of best' denotes the number of datasets on which the proposed SMILE obtains the highest accuracy.

Dataset	NCA	LMNN	ITML	R2LML	SCML	RVML	GMML	DMLMJ	SNC	SMILE
Australian	80.0±1.6	78.8±2.6	77.2±1.9	84.7±1.3	82.3±1.4	83.0±1.6	84.4±1.0	83.9±1.3	81.8±8.8	**86.0±0.7**
Cancer	95.4±1.3	96.0±0.7	96.1±1.1	96.7±0.8	96.5±0.5	95.2±1.0	96.5±0.8	96.5±0.5	95.1±1.7	**96.8±0.6**
Climate	91.5±2.1	91.8±1.3	86.7±1.0	91.7±1.7	91.5±1.5	92.2±1.1	91.3±2.5	92.9±1.9	92.0±1.7	**93.5±1.7**
Credit	80.6±2.0	82.2±1.4	77.6±2.0	**86.1±1.5**	83.5±1.2	83.5±1.8	85.9±1.7	84.6±1.4	83.4±3.7	85.6±1.9
German	70.0±2.9	67.9±1.5	67.0±2.1	72.9±1.8	70.9±2.7	71.7±1.8	71.6±1.1	69.3±2.7	70.1±3.3	**75.5±1.1**
Haberman	67.4±3.3	67.9±3.3	68.0±4.1	71.1±3.4	69.2±2.5	66.7±2.3	71.2±3.4	68.5±3.2	72.0±5.2	**72.4±3.3**
Heart	75.6±2.0	76.2±3.8	76.9±3.3	82.0±3.8	79.0±3.2	77.7±4.1	81.2±2.7	80.6±2.8	77.0±5.3	**84.0±2.2**
ILPD	66.8±1.2	67.0±2.1	68.7±2.8	65.9±2.2	68.0±2.9	68.0±2.9	67.1±2.2	68.0±1.6	68.9±2.7	**71.3±1.7**
Liver	59.8±3.4	61.0±4.8	57.2±4.0	**66.8±3.7**	61.7±4.6	64.6±3.9	63.8±5.4	60.9±3.8	63.3±5.2	62.8±5.8
Pima	65.9±3.0	68.5±1.6	68.0±2.0	72.3±1.5	71.1±2.6	69.5±1.7	73.0±1.8	71.1±2.3	**74.0±2.6**	73.2±2.0
Ringnorm	69.3±0.7	65.2±0.7	65.8±0.9	NA	70.9±0.7	72.3±0.6	72.5±0.5	73.9±0.7	71.3±0.6	**77.1±0.5**
Twonorm	96.7±0.4	95.6±0.5	96.4±0.3	NA	97.3±0.4	97.3±0.3	97.5±0.3	97.7±0.2	97.3±0.2	**97.9±0.3**
Average	76.6	76.5	75.5	NA	78.5	78.5	78.8	79.7	79.0	**81.3**
# of best	0	0	0	2	0	0	0	0	1	9

Evaluation on classification performance As shown in Table 1, with only two representative instances learned for each class, the proposed SMILE achieves the best accuracy on 9 out of the 12 datasets; none of the other methods performs the best on more than 2 datasets. The average accuracy of SMILE is also the highest. These results suggest that SMILE, though simple, enjoys competitive performance against existing metric learning algorithms, thanks to its theoretical foundation.

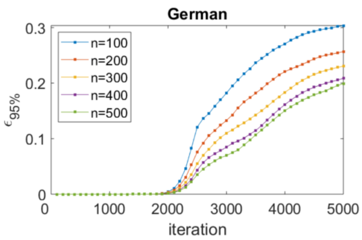

Figure 2: Effect of training iterations and sample size on parameter concentration.

Visualization of the concentration behavior Our theoretical finding suggests that randomness of parameters is caused by random sampling and will accumulate over iterations. We now verify this finding with an empirical study on the dataset German. More specifically, we learn parameters $\boldsymbol{L}, \boldsymbol{r}^m$ from a subset of the data, which serves as $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n)$ in Lemma 2, learn parameters from the entire dataset, which serves as $\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n)$, and quantify their differences via the L_2-norm. The total sample size is 1000 and the subset size is selected as $\{100, 200, \ldots, 500\}$. After randomly sampling the subset for 100 times, we calculate the 95th percentile of the norm differences and denote this value as $\epsilon_{95\%}$. $\epsilon_{95\%}$ can be interpreted as the minimum radius ϵ of ball $\left(\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n), \epsilon\right)$ such that the bound (Eq. 6) holds with 95% probability. From Fig. 2, we first see that learning from fewer training instances leads to a larger value of $\epsilon_{95\%}$, which signifies that sampling randomness contributes to the variance of learned parameters. Second, we see that learning with more iterations increases $\epsilon_{95\%}$, which is also consistent with the theoretical result. Moreover, the rate of increase is exponential in the early stage of training and decreases gradually towards zero, which implies that parameters are optimized to local minima and will no longer be updated.

5 Conclusion

This paper presents a new route to the generalization guarantee on classifiers optimized via GD, considering the influence of sampling randomness to the concentration property of parameters and embracing algorithmic parameters. We propose a new decomposition theorem to obtain the generalization PAC bound, which consequently guarantees the agnostic PAC learnability. We demonstrate the necessity of Lipschitz smooth classifiers and loss functions for generalization and theoretically justify the benefit of early stopping. Our results are derived based only on the Lipschitz property over the parameter space, and hence are applicable to non-convex optimization problems. In addition, we propose a new metric learning method as an illustrative example to demonstrate the practicability of the appealing theoretical results. In the future, we intend to investigate the link between the concentration property and the local convergence behavior, and take it into account to derive tighter bounds.

Broader Impact

This paper is a theoretical analysis relating to gradient descent and metric learning algorithms, and hence does not make a direct impact on ethical and societal issues. The findings can be used to design more effective training strategies or algorithms, and consequently benefit the downstream applications.

References

[1] A. Bellet and A. Habrard. Robustness and generalization for metric learning. *Neurocomputing*, 151:259–267, 2015.

[2] A. Bellet, A. Habrard, and M. Sebban. Similarity learning for provably accurate sparse linear classification. In *International Coference on International Conference on Machine Learning*, pages 1491–1498, 2012.

[3] W. Bian and D. Tao. Constrained empirical risk minimization framework for distance metric learning. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1194–1205, 2012.

[4] J. Bohné, Y. Ying, S. Gentric, and M. Pontil. Large margin local metric learning. In *European Conference on Computer Vision*, pages 679–694. Springer, 2014.

[5] O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526, 2002.

[6] Q. Cao, Z.-C. Guo, and Y. Ying. Generalization bounds for metric and similarity learning. *Machine Learning*, 102(1):115–132, 2016.

[7] S. Chen, L. Luo, J. Yang, C. Gong, J. Li, and H. Huang. Curvilinear distance metric learning. In *Advances in Neural Information Processing Systems*, pages 4225–4234, 2019.

[8] R. Collobert and S. Bengio. Links between perceptrons, mlps and svms. In *International Conference on Machine Learning*, page 23, 2004.

[9] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *International Conference on Machine Learning*, pages 209–216. ACM, 2007.

[10] W. de Vazelhes, C. Carey, Y. Tang, N. Vauquier, and A. Bellet. metric-learn: Metric Learning Algorithms in Python. Technical report, arXiv:1908.04710, 2019.

[11] M. Dong, Y. Wang, X. Yang, and J.-H. Xue. Learning local metrics and influential regions for classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(6): 1522–1529, 2020.

[12] D. Dua and C. Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

[13] R. M. Dudley. *Uniform central limit theorems*, volume 142. Cambridge University Press, 2014.

[14] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[15] L. Gautheron, A. Habrard, E. Morvant, and M. Sebban. Metric learning from imbalanced data with generalization guarantees. *Pattern Recognition Letters*, 133:298–304, 2020.

[16] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems*, pages 513–520, 2005.

[17] L. A. Gottlieb, A. Kontorovich, and R. Krauthgamer. Efficient classification for metric data. *IEEE Transactions on Information Theory*, 60(9):5750–5759, 2014.

[18] Z.-C. Guo and Y. Ying. Guaranteed classification via regularized similarity learning. *Neural Computation*, 26(3):497–522, 2014.

[19] C. Gupta, A. S. Suggala, A. Goyal, H. V. Simhadri, B. Paranjape, A. Kumar, S. Goyal, R. Udupa, M. Varma, and P. Jain. ProtoNN: Compressed and accurate kNN for resource-scarce devices. In *International Conference on Machine Learning*, pages 1331–1340, 2017.

[20] M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. *ICML*, 2016.

[21] Y. Huang, C. Li, M. Georgiopoulos, and G. C. Anagnostopoulos. Reduced-rank local distance metric learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 224–239. Springer, 2013.

[22] Z. Huang, R. Wang, S. Shan, and X. Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 140–149, 2015.

[23] P. J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101, 1964.

[24] Z. Huo, F. Nie, and H. Huang. Robust and effective metric learning using capped trace norm: Metric learning via capped trace norm. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1605–1614, 2016.

[25] R. Jin, S. Wang, and Y. Zhou. Regularized distance metric learning: Theory and algorithm. In *Advances in Neural Information Processing Systems*, pages 862–870, 2009.

[26] M. Kusner, S. Tyree, K. Q. Weinberger, and K. Agrawal. Stochastic neighbor compression. In *International Conference on Machine Learning*, pages 622–630, 2014.

[27] D. Lim, G. Lanckriet, and B. McFee. Robust structural metric learning. In *International Conference on Machine Learning*, pages 615–623, 2013.

[28] U. v. Luxburg and O. Bousquet. Distance-based classification with lipschitz functions. *Journal of Machine Learning Research*, 5(Jun):669–695, 2004.

[29] B. Mason, L. Jain, and R. Nowak. Learning low-dimensional metrics. In *Advances in Neural Information Processing Systems*, pages 4139–4147, 2017.

[30] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh. No fuss distance metric learning using proxies. In *IEEE International Conference on Computer Vision*, pages 360–368, 2017.

[31] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 5949–5958, 2017.

[32] B. Nguyen, C. Morell, and B. De Baets. Supervised distance metric learning through maximization of the jeffrey divergence. *Pattern Recognition*, 64:215–225, 2017.

[33] Y.-K. Noh, M. Sugiyama, K.-E. Kim, F. Park, and D. D. Lee. Generative local metric learning for kernel regression. In *Advances in Neural Information Processing Systems*, pages 2452–2462, 2017.

[34] M. Perrot and A. Habrard. Regressive virtual metric learning. In *Advances in Neural Information Processing Systems*, pages 1810–1818, 2015.

[35] Q. Qian, J. Tang, H. Li, S. Zhu, and R. Jin. Large-scale distance metric learning with uncertainty. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8542–8550, 2018.

[36] C. Rasmussen, R. Neal, G. Hinton, D. Van Camp, M. Revow, Z. Ghahramani, R. Kustra, and R. Tibshirani. Delve data for evaluating learning in valid experiments. URL `http://www.cs.toronto.edu/~delve`.

[37] K. Roth, T. Milbich, S. Sinha, P. Gupta, B. Ommer, and J. P. Cohen. Revisiting training strategies and generalization performance in deep metric learning. In *International Conference on Machine Learning*, 2020.

[38] Y. Shi, A. Bellet, and F. Sha. Sparse compositional metric learning. In *AAAI Conference on Artificial Intelligence*, pages 2078–2084, 2014.

[39] M. Taheri, Z. Moslehi, A. Mirzaei, and M. Safayani. A self-adaptive local metric learning method for classification. *Pattern Recognition*, 96:106994, 2019.

[40] L. van der Maaten. Matlab toolbox for dimensionality reduction. URL `https://lvdmaaten.github.io/drtoolbox/`.

[41] N. Verma and K. Branson. Sample complexity of learning Mahalanobis distance metrics. In *Advances in Neural Information Processing Systems*, pages 2584–2592, 2015.

[42] J. Wang, A. Kalousis, and A. Woznica. Parametric local metric learning for nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1601–1609, 2012.

[43] N. Weaver. *Lipschitz Algebras*. World Scientific, 1999.

[44] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

[45] P. Xie, W. Wu, Y. Zhu, and E. Xing. Orthogonality-promoting distance metric learning: Convex relaxation and theoretical analysis. In *International Conference on Machine Learning*, pages 5403–5412, 2018.

[46] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems*, pages 505–512, 2002.

[47] Y. Yao, L. Rosasco, and A. Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

[48] H.-J. Ye, D.-C. Zhan, and Y. Jiang. Fast generalization rates for distance metric learning. *Machine Learning*, 108(2):267–295, 2019.

[49] H.-J. Ye, D.-C. Zhan, N. Li, and Y. Jiang. Learning multiple local metrics: Global consideration helps. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[50] Y. Ying, K. Huang, and C. Campbell. Sparse metric learning via smooth optimization. In *Advances in Neural Information Processing Systems*, pages 2214–2222, 2009.

[51] P. Zadeh, R. Hosseini, and S. Sra. Geometric mean metric learning. In *International Conference on Machine Learning*, pages 2464–2471, 2016.

[52] K. Zhong, R. Guo, S. Kumar, B. Yan, D. Simcha, and I. Dhillon. Fast classification with binary prototypes. In *Artificial Intelligence and Statistics*, pages 1255–1263, 2017.

[53] P. Zhu, H. Cheng, Q. Hu, Q. Wang, and C. Zhang. Towards generalized and efficient metric learning on riemannian manifold. In *International Joint Conference on Artificial Intelligence*, pages 3235–3241, 2018.

Generalization Bound of Gradient Descent for Non-Convex Metric Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Sec. A lists properties of Lipschitz functions. Sec. B includes established definitions and theorems that will be used in the proof, including definitions of PAC learnability and agnostic PAC learnability, the McDiarmid's Inequality. Sec. C provides proofs of theorems and lemmas. Sec. D shows that the classifier of SMILE is smooth and gives the updating equations of the gradient descent algorithm. Sec. E lists the information about the datasets.

A Properties of Lipschitz functions

The Lipschitz constant of differentiable functions can be obtained from their gradients; this follows from the mean value theorem as shown below.

Theorem A.1. [3] Let $\mathcal{U} \in \mathbb{R}^n$ be open, $h : \mathcal{U} \to \mathbb{R}$ be differentiable and the line segment $[\boldsymbol{u}_1, \boldsymbol{u}_2] \in \mathcal{U}$, where $[\boldsymbol{u}_1, \boldsymbol{u}_2] = \{\boldsymbol{v} \mid \boldsymbol{v} = \boldsymbol{u}_1 + t(\boldsymbol{u}_2 - \boldsymbol{u}_1), t \in [0, 1]\}$ joins $\boldsymbol{u}_1$ to $\boldsymbol{u}_2$. Based on the *Mean Value Theorem*, there exists a $\boldsymbol{u} \in [\boldsymbol{u}_1, \boldsymbol{u}_2]$

$$f(\boldsymbol{u}_2) - f(\boldsymbol{u}_1) = f'(\boldsymbol{u})^T(\boldsymbol{u}_2 - \boldsymbol{u}_1).$$

Corollary A.1. Let $\mathcal{U} \in \mathbb{R}^n$ be open and convex, $h : \mathcal{U} \to \mathbb{R}$ be differentiable inside $\mathcal{U}$, then the following inequality holds:

$$\mathrm{lip}(h \leftarrow \boldsymbol{u}) = \max_{\boldsymbol{u}_1, \boldsymbol{u}_2 \in \mathcal{U}, \boldsymbol{u}_1 \neq \boldsymbol{u}_2} \frac{|h(\boldsymbol{u}_2) - h(\boldsymbol{u}_1)|}{\|\boldsymbol{u}_2 - \boldsymbol{u}_1\|} \leq \max_{u \in \mathcal{U}} \|h'(\boldsymbol{u})\|.$$

Proof. Since $\mathcal{U}$ is convex, $\forall \boldsymbol{u}_1, \boldsymbol{u}_2 \in \mathcal{U}, \boldsymbol{u}_1 \neq \boldsymbol{u}_2$, the line segment $[\boldsymbol{u}_1, \boldsymbol{u}_2] = \{\boldsymbol{v} \mid \boldsymbol{v} = \boldsymbol{u}_1 + t(\boldsymbol{u}_2 - \boldsymbol{u}_1), t \in [0, 1]\} \in \mathcal{U}$.

$$|h(\boldsymbol{u}_2) - h(\boldsymbol{u}_1)| =_{(a)} |h'(\boldsymbol{u})^T(\boldsymbol{u}_2 - \boldsymbol{u}_1)| \leq_{(b)} \|h'(\boldsymbol{u})\|\|\boldsymbol{u}_2 - \boldsymbol{u}_1\| \leq_{(c)} \max_{u \in \mathcal{U}} \|h'(\boldsymbol{u})\|\|\boldsymbol{u}_2 - \boldsymbol{u}_1\|,$$

where equality (a) is due to Theorem A.1; inequality (b) is due to the Cauchy-Schwarz inequality; inequality (c) is due to $\|h'(\boldsymbol{u})\| \leq \max_{u \in \mathcal{U}} \|h'(\boldsymbol{u})\|$. □

Sophisticated Lipschitz functions can be constructed from the basic ones using the following lemma.

Lemma A.1. [4, 10] Let $\mathrm{lip}(h_1 \leftarrow \boldsymbol{u}) \leq L_1$, $\mathrm{lip}(h_2 \leftarrow \boldsymbol{u}) \leq L_2$ and $\mathrm{lip}(h_2 \circ h_1 \leftarrow h_1) \leq L_3$, where $\circ$ denotes the composition of functions. Then
(a) $\mathrm{lip}(ah_1 \leftarrow \boldsymbol{u}) \leq |a|L_1$, where a is a constant;
(b) $\mathrm{lip}(h_1 + h_2 \leftarrow \boldsymbol{u}) \leq L_1 + L_2$, $\mathrm{lip}(h_1 - h_2 \leftarrow \boldsymbol{u}) \leq L_1 + L_2$;
(c) $\mathrm{lip}(\min(h_1, h_2) \leftarrow \boldsymbol{u}) \leq \max\{L_1, L_2\}$, $\mathrm{lip}(\max(h_1, h_2) \leftarrow \boldsymbol{u}) \leq \max\{L_1, L_2\}$, where $\min(h_1, h_2)$ or $\max(h_1, h_2)$ denote the pointwise minimum or maximum of functions h_1 and h_2;
(d) $\mathrm{lip}(h_2 \circ h_1 \leftarrow \boldsymbol{u}) \leq L_1 L_3$.

This lemma illustrates that after the operations of multiplication by constant, addition, subtraction, minimization, maximization and function composition, the functions are still Lipschitz continuous.

B Preliminaries

Definition B.1. [6, 8] A hypothesis class $\mathcal{H}$ is *Probably Approximately Correct (PAC) learnable* if there exist a function $n_{\mathcal{H}}^{L} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}_{\mathcal{X}}$ over $\mathcal{X}$, and for every target function $g \in \mathcal{G}$, if there exists an $h^* \in \mathcal{H}$ which returns the same classification result as g, then when running the learning algorithm on $n \geq n_{\mathcal{H}}^{L}(\epsilon, \delta)$ independent and identically distributed (i.i.d.) instances generated by $\mathcal{D}_{\mathcal{X}}$ and labeled by g, the algorithm returns a hypothesis $\hat{h}$ such that, with probability at least $1-\delta$, $R(\hat{h}) \leq \epsilon$; this can be equivalently written as

$$\mathbb{P}_{\boldsymbol{x}^n}[R(\hat{h}) \leq \epsilon] \geq 1-\delta,$$

or

$$\mathbb{P}_{\boldsymbol{x}^n}\left[\mathbb{E}_{\boldsymbol{x}'}\left[l\big(\hat{h}(\boldsymbol{x}'), g(\boldsymbol{x}')\big)\right] \leq \epsilon\right] \geq 1-\delta,$$

where the probability is taken over $\boldsymbol{x}_n$ and $\hat{h}$ is a random variable related to $\boldsymbol{x}_n$.

Definition B.2. [6, 2] A hypothesis class $\mathcal{H}$ is *agnostic PAC learnable* or has *agnostic PAC learnability* if there exist a function $n_{\mathcal{H}}^{AL} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$ and for every distribution $\mathcal{D}_{\mathcal{Z}}$ over $\mathcal{Z}$, when running the learning algorithm on $n \geq n_{\mathcal{H}}^{AL}(\epsilon, \delta)$ i.i.d. instances generated by $\mathcal{D}_{\mathcal{Z}}$, the algorithm returns a hypothesis $\hat{h}$ such that, with probability at least $1-\delta$, $R(\hat{h}) - \min_{h\in\mathcal{H}} R(h) \leq \epsilon$; this can be equivalently written as

$$\mathbb{P}_{\boldsymbol{z}^n}\left[R(\hat{h}) - \min_{h\in\mathcal{H}} R(h) \leq \epsilon\right] \geq 1-\delta,$$

or

$$\mathbb{P}_{\boldsymbol{z}^n}\left[\mathbb{E}_{\boldsymbol{z}'}\left[l\big(\hat{h}(\boldsymbol{x}'), y\big)\right] - \min_{h\in\mathcal{H}} \mathbb{E}_{\boldsymbol{z}'}\left[l\big(h(\boldsymbol{x}'), y\big)\right] \leq \epsilon\right] \geq 1-\delta.$$

where the probability is taken over $\boldsymbol{z}_n$ and $\hat{h}$ is a random variable related to $\boldsymbol{z}_n$.

Lemma B.1. [5, (McDiarmid's Inequality)] Let $\boldsymbol{z}^n = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{i-1}, \boldsymbol{z}_i, \boldsymbol{z}_{i+1}, \ldots, \boldsymbol{z}_n\}$ be n independent samples. Let $\boldsymbol{z}^{n,i} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{i-1}, \boldsymbol{z}'_i, \boldsymbol{z}_{i+1}, \ldots, \boldsymbol{z}_n\}$, where the replacement example $\boldsymbol{z}'_i$ is assumed to be drawn from the same distribution of $\boldsymbol{z}_i$ and is independent from $\boldsymbol{z}^n$. Furthermore, let $m : \mathcal{Z}^n \to \mathbb{R}$ be a function of $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$ that satisfies $\forall i, \forall \boldsymbol{z}^n, \forall \boldsymbol{z}^{n,i}$

$$|m(\boldsymbol{z}^n) - m(\boldsymbol{z}^{n,i})| \leq c_i, \tag{1}$$

for some constant c_i. Then for all $\epsilon > 0$,

$$\mathbb{P}_{\boldsymbol{z}^n}[m(\boldsymbol{z}^n) - \mathbb{E}_{\boldsymbol{z}^n}[m(\boldsymbol{z}^n)] \geq \epsilon] \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n c_i^2}\right),$$

$$\mathbb{P}_{\boldsymbol{z}^n}[\mathbb{E}_{\boldsymbol{z}^n}[m(\boldsymbol{z}^n)] - m(\boldsymbol{z}^n) \geq \epsilon] \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n c_i^2}\right);$$

that is,

$$\mathbb{P}_{\boldsymbol{z}^n}[|m(\boldsymbol{z}^n) - \mathbb{E}_{\boldsymbol{z}^n}[m(\boldsymbol{z}^n)]| \geq \epsilon] \leq 2\exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n c_i^2}\right). \tag{2}$$

C Proofs of theorems and lemmas

C.1 Proof of Lemma 1

Proof. Let E_1 be the set of events of $R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon$ and E_2 be the set of events of $\max_{h\in\mathcal{H}}\big(R(h) - R_n(\boldsymbol{z}^n, h)\big) \leq \epsilon$. The probabilities of these two events are given as follows:

$$\mathbb{P}_{\boldsymbol{z}^n}(E_1) = \int (p(\boldsymbol{z}^n)\mathbb{1}[E_1])\, d\boldsymbol{z}^n$$

$$\mathbb{P}_{\boldsymbol{z}^n}(E_2) = \int (p(\boldsymbol{z}^n)\mathbb{1}[E_2])\, d\boldsymbol{z}^n.$$

1. At the points $\boldsymbol{z}^n$ where $\mathbb{1}[E_2] = 1$, we have $\mathbb{1}[E_1] = 1$ and thus

$$p(\boldsymbol{z}^n)\mathbb{1}[E_1] = p(\boldsymbol{z}^n)\mathbb{1}[E_2].$$

2. At the points $\boldsymbol{z}^n$ where $\mathbb{1}[E_2] = 0$, we have

$$p(\boldsymbol{z}^n)\mathbb{1}[E_1] \geq 0 = p(\boldsymbol{z}^n)\mathbb{1}[E_2].$$

Therefore, integrating over all possible points $\boldsymbol{z}^n$, we have $\left(\int p(\boldsymbol{z}^n)\mathbb{1}[E_1]\right) \geq \left(\int p(\boldsymbol{z}^n)\mathbb{1}[E_2]\right)$. That is, $\mathbb{P}_{\boldsymbol{z}^n}(E_1) \geq \mathbb{P}_{\boldsymbol{z}^n}(E_2)$. □

C.2 Proof of Theorem 1

After proving Proposition C.1, Theorem 1 is proved.

Proposition C.1. Suppose the range of the risk function $r(\boldsymbol{z}, h)$ is $[0, C_r]$, then

$$\mathbb{P}_{\boldsymbol{z}^n}\Big[\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h) - E_{\boldsymbol{z}^n}\big[\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h)\big] \geq \epsilon\Big) \leq \exp\left(\frac{-2n\epsilon^2}{C_r^2}\right).$$

Proof. Given $\boldsymbol{z}^n$ and a fixed hypothesis class of $\mathcal{H}$, the value of $a(\boldsymbol{z}^n) = \min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h)$ is fixed and the mapping $a : \mathcal{Z}^n \to \mathbb{R}$ is a function. Therefore, the McDiarmid's inequality (Lemma B.1) can be applied as long as the bounded difference condition (Eq. 1) holds. We show that $|\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h) - \min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^{n,i}, h)|$ is bounded as follows:

$$\begin{aligned}
&\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^{n,i}, h)\\
=&\min_{h\in\mathcal{H}}\left(R_n(\boldsymbol{z}^n, h) - \frac{r(z_i, h)}{n} + \frac{r(z_i', h)}{n}\right)\\
\leq&\min_{h\in\mathcal{H}}\left(R_n(\boldsymbol{z}^n, h) - 0 + \frac{C_r}{n}\right)\\
=&\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h) + \frac{C_r}{n}.
\end{aligned}$$

Similarly,

$$\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h) \leq \min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^{n,i}, h) + \frac{C_r}{n}.$$

Therefore

$$\left|\min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h) - \min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^{n,i}, h)\right| \leq \frac{C_r}{n}.$$

The result is obtained by substituting $c_i = \frac{C_r}{n}$ into Lemma B.1. □

Theorem 1 is proved as follows.

Proof. Let $\hat{h} \in \operatorname{argmin}_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h)$, we have

$$R_n(\boldsymbol{z}^n, \hat{h}) = \min_{h\in\mathcal{H}} R_n(\boldsymbol{z}^n, h).$$

Suppose

$$\mathbb{P}_{\boldsymbol{z}^n}[R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon/2] \geq 1 - \delta/2,$$

$$\mathbb{P}_{\boldsymbol{z}^n}\big[R_n(\boldsymbol{z}^n, \hat{h}) - E_{\boldsymbol{z}^n}[R_n(\boldsymbol{z}^n, \hat{h})] \leq \epsilon/2\big] \geq 1 - \delta/2.$$

Let $E_1 = \{\boldsymbol{z}^n | R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon/2\}$ and $E_2 = \{\boldsymbol{z}^n | R_n(\boldsymbol{z}^n, \hat{h}) - E_{\boldsymbol{z}^n}[R_n(\boldsymbol{z}^n, \hat{h})] \leq \epsilon/2\}$. $\forall \boldsymbol{z}^n \in E_1 \cap E_2$, we have

$$\begin{aligned} & R(\hat{h}) \\ (a) \quad & \leq R_n(\boldsymbol{z}^n, \hat{h}) + \frac{\epsilon}{2} \\ (b) \quad & \leq \mathbb{E}_{\boldsymbol{z}^n}[R_n(\boldsymbol{z}^n, \hat{h})] + \epsilon \\ (c) \quad & = \mathbb{E}_{\boldsymbol{z}^n} \min_{h \in \mathcal{H}} \frac{\sum_{i=1}^n r(\boldsymbol{z}_i, h)}{n} + \epsilon \\ (d) \quad & \leq \min_{h \in \mathcal{H}} \mathbb{E}_{\boldsymbol{z}^n} \frac{\sum_{i=1}^n r(\boldsymbol{z}_i, h)}{n} + \epsilon \\ (e) \quad & = \min_{h \in \mathcal{H}} \mathbb{E}_{\boldsymbol{z}} r(\boldsymbol{z}, h) + \epsilon \\ (f) \quad & = \min_{h \in \mathcal{H}} R(h) + \epsilon, \end{aligned}$$

where inequality (a) is due to $R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon/2$; inequality (b) is due to $R_n(\boldsymbol{z}^n, \hat{h}) - E_{\boldsymbol{z}^n}[R_n(\boldsymbol{z}^n, \hat{h})] \leq \epsilon/2$; equality (c) is due to the definitions of $R_n(\boldsymbol{z}^n, h)$ and $\hat{h}$; inequality (d) is due to change the order of $E_{\boldsymbol{z}^n}$ and $\min_{h \in \mathcal{H}}$; equality (e) is due to the identical assumption of $\boldsymbol{z}^n$; equality (f) is due to the definition of $R(h)$.

Therefore

$$\begin{aligned} & \mathbb{P}_{\boldsymbol{z}^n}\left[R(\hat{h}) \leq \min_{h \in \mathcal{H}} R(h) + \epsilon\right] \\ (a) \quad & \geq \mathbb{P}_{\boldsymbol{z}^n}[E_1 \cap E_2] \\ (b) \quad & \geq 1 - \delta/2 - \delta/2, \end{aligned}$$

where inequality (a) is due to the relationship between $E_1 \cap E_2$ and $R(\hat{h}) \leq \min_{h \in \mathcal{H}} R(h) + \epsilon$; inequality (b) is due to the probability of union of sets.

Based on Proposition C.1, in order to guarantee $\mathbb{P}_{\boldsymbol{z}^n}\left[R_n(\boldsymbol{z}^n, \hat{h}) - E_{\boldsymbol{z}^n}[R_n(\boldsymbol{z}^n, \hat{h})] \leq \epsilon/2\right] \geq 1 - \delta/2$, $\frac{2C_r^2}{\epsilon^2} \ln \frac{4}{\delta}$ instances are required. Meanwhile, based on the definition of generalization PAC bound (Definition 4 of the main text), in order to guarantee $\mathbb{P}_{\boldsymbol{z}^n}[R(\hat{h}) - R_n(\boldsymbol{z}^n, \hat{h}) \leq \epsilon/2] \geq 1 - \delta/2$, $m_{\mathcal{H}}^G(\epsilon/2, \delta/2)$ instances are required. Therefore, with more than $\max\left(m_{\mathcal{H}}^G(\epsilon/2, \delta/2), \frac{2C_r^2}{\epsilon^2} \ln \frac{4}{\delta}\right)$ instances, $\mathbb{P}_{\boldsymbol{z}^n}\left[R(\hat{h}) \leq \min_{h \in \mathcal{H}} R(h) + \epsilon\right] \geq 1 - \delta$ is satisfied. Based on the definition of the agnostic PAC learnability (Definition B.2), the hypothesis class is agnostic PAC learnable and the agnostic PAC learner for $\mathcal{H}$ is $\mathrm{ERM}_{\mathcal{H}}$. □

C.3 Proof of Theorem 2

Proof. Let E_1 denote the set of events $R(\hat{\boldsymbol{w}}) - R_n(\boldsymbol{z}^n, \hat{\boldsymbol{w}}) \leq \epsilon$, E_2 denote the set of events $\boldsymbol{w} \in \hat{\mathcal{W}}$, and E_3 denote the set of events $\max_{\boldsymbol{w} \in \hat{\mathcal{W}}}[R(\boldsymbol{w}) - R_n(\boldsymbol{z}^n, \boldsymbol{w})] \leq \epsilon$.

$$\begin{aligned} & \mathbb{P}_{\boldsymbol{z}^n}[\neg E_1] \\ & = \mathbb{P}_{\boldsymbol{z}^n}[\neg E_1, E_2] + \mathbb{P}_{\boldsymbol{z}^n}[\neg E_1, \neg E_2] \\ (a) \quad & \leq \mathbb{P}_{\boldsymbol{z}^n}[\neg E_1, E_2] + \delta_1 \\ (b) \quad & \leq \mathbb{P}_{\boldsymbol{z}^n}[\neg E_3] + \delta_1 \\ & = \delta_2 + \delta_1; \end{aligned}$$

where inequality (a) is due to $\mathbb{P}_{\boldsymbol{z}^n}[\neg E_1, \neg E_2] \leq \mathbb{P}_{\boldsymbol{z}^n}[\neg E_2] = 1 - \mathbb{P}_{\boldsymbol{z}^n}[E_2] \leq \delta_1$; inequality (b) is based on the relationship between $\mathbb{1}[E_2]\mathbb{1}[\neg E_1]$ and $\mathbb{1}[E_3]$. At the points $\boldsymbol{z}^n$ that satisfy $\boldsymbol{m}(\boldsymbol{z}^n) = \hat{\boldsymbol{w}} \in \hat{\mathcal{W}}$, $\mathbb{1}[\neg E_1] = 1 \Rightarrow \mathbb{1}[\neg E_3] = 1$, thus $\mathbb{1}[E_2]\mathbb{1}[\neg E_1] \leq \mathbb{1}[\neg E_3]$ and $\mathbb{P}_{\boldsymbol{z}^n}[\neg E_1, E_2] \leq \mathbb{P}_{\boldsymbol{z}^n}[\neg E_3]$. □

C.4 Proof of Lemma 2

Proof. To show that $\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s})$ is concentrated around its expectation, we make use of the McDiarmid's Inequality (Lemma B.1). First, we note that $\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}) : \mathcal{Z}^n \to \mathbb{R}$ is function mapping from random variables to a real value, and $\boldsymbol{z}^n$ satisfies the independent assumption. Second, we show that $|\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s}) - \boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^{n,i}; \boldsymbol{s})|$ is bounded. $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s})$ and $\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n; \boldsymbol{s})$ are temporarily simplified to $\boldsymbol{m}^{(T)}(\boldsymbol{z}^n)$ and $\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n)$, respectively. $\forall \boldsymbol{s}, \forall q, |\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^{n,i})| \leq \|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(T)}(\boldsymbol{z}^{n,i})\|$, where $\|\cdot\|$ denotes the vector L_2-norm[1]. We will now discuss the bound of $\|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(T)}(\boldsymbol{z}^{n,i})\|$.

(1) Decompose $\boldsymbol{m}^{(t)}(\boldsymbol{z}^n)$. To understand the influence of $\boldsymbol{z}_i$, the updating equation of $\boldsymbol{m}^{(t)}(\boldsymbol{z}^n)$ is divided into two parts:

$$\boldsymbol{m}^{(t)}(\boldsymbol{z}^n) = \left(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n) - \sum_{j \in [n]/i} \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_j, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)} \right) - \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)}.$$

Representing the above updating process via the function G gives:

$$\boldsymbol{m}^{(t)}(\boldsymbol{z}^n) = G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) - \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)}.$$

For both $\boldsymbol{z}^n$ and $\boldsymbol{z}^{n,i}$, $G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) = G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i}))$ because the training instances considered in G are the same. Then

$$\begin{aligned}
&\|\boldsymbol{m}^{(t)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(t)}(\boldsymbol{z}^{n,i})\| \\
=&\Big\| G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) - \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)} - \\
&G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})) + \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}'_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})} \Big\| \\
\leq&\Big\| \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)} - \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}'_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})} \Big\| \text{ (Term 1)+} \\
&\|G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) - G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i}))\| \text{ (Term 2)}.
\end{aligned}$$

Term 1 and Term 2 in the inequality can be bounded by using the Lipschitz constant of a function r with respect to $\boldsymbol{w}$ and the Lipschitz constant of G with respect to $\boldsymbol{w}$, respectively.

(2) Bound Term 1. Recall that the Lipschitz constant is defined as:

$$\text{lip}(r \leftarrow \boldsymbol{w}) = \max_{\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathcal{W}, \boldsymbol{w}_1 \neq \boldsymbol{w}_2, \boldsymbol{z} \in \mathcal{Z}} \frac{|r(\boldsymbol{z}; \boldsymbol{w}_1) - r(\boldsymbol{z}; \boldsymbol{w}_2)|}{\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|}.$$

Term 1 is bounded as follows:

$$\begin{aligned}
&\Big\| \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)} - \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}'_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})} \Big\| \\
\leq&\Big\| \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)} \Big\| + \Big\| \frac{\alpha^{(t)}}{n} \frac{\partial r(\boldsymbol{z}'_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})} \Big\| \\
\leq&\frac{2\alpha^{(t)}}{n} \text{lip}\,(r \leftarrow \boldsymbol{w}).
\end{aligned}$$

(3) Bound Term 2. Let $\eta = \text{lip}\,(G \leftarrow \boldsymbol{w})$.

$$\|G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n)) - G(\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i}))\| \leq \eta \|\boldsymbol{m}^{(t-1)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(t-1)}(\boldsymbol{z}^{n,i})\Big\|$$

(4) Bound $\|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(T)}(\boldsymbol{z}^{n,i})\|$

$t = 1$

$$\begin{aligned}
&\|\boldsymbol{m}^{(1)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(1)}(\boldsymbol{z}^{n,i})\| \\
\leq&\Big\| \frac{\alpha^{(1)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{w}^0} - \frac{\alpha^{(1)}}{n} \frac{\partial r(\boldsymbol{z}'_i, \boldsymbol{w})}{\partial \boldsymbol{w}} |_{\boldsymbol{w}^0} \Big\| + \|G(\boldsymbol{w}^0) - G(\boldsymbol{w}^0)\| \\
\leq&\frac{2\alpha^{(1)}}{n} \text{lip}\,(r \leftarrow \boldsymbol{w});
\end{aligned}$$

[1]In the cases of $\boldsymbol{m}$ being a matrix, the matrix will be reshaped into a vector and the vector L_2-norm can then be used; this is equivalent to using the matrix Frobenius norm directly.

$t = 2$

$$
\begin{aligned}
&\|\boldsymbol{m}^{(2)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(2)}(\boldsymbol{z}^{n,i})\| \\
\leq &\Big\| \frac{\alpha^{(2)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{m^{(1)}(\boldsymbol{z}^n)} - \frac{\alpha^{(2)}}{n} \frac{\partial r(\boldsymbol{z}_i', \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\boldsymbol{m}^{(1)}(\boldsymbol{z}^{n,i})} \Big\| + \\
&\|G(\boldsymbol{m}^{(1)}(\boldsymbol{z}^n)) - G(\boldsymbol{m}^{(1)}(\boldsymbol{z}^{n,i}))\| \\
\leq &\frac{2\alpha^{(2)}}{n} \operatorname{lip}(r \leftarrow \boldsymbol{w}) + \eta \frac{2\alpha^{(1)}}{n} \operatorname{lip}(r \leftarrow \boldsymbol{w}) \\
= &\frac{2(\eta\alpha^{(1)} + \alpha^{(2)}) \operatorname{lip}(r \leftarrow \boldsymbol{w})}{n};
\end{aligned}
$$

$\vdots$

$t = T$

$$
\begin{aligned}
&\|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(T)}(\boldsymbol{z}^{n,i})\| \\
\leq &\Big\| \frac{\alpha^{(T)}}{n} \frac{\partial r(\boldsymbol{z}_i, \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\boldsymbol{m}^{(T-1)}(\boldsymbol{z}^n)} - \frac{\alpha^{(T)}}{n} \frac{\partial r(\boldsymbol{z}_i', \boldsymbol{w})}{\partial \boldsymbol{w}}|_{\boldsymbol{m}^{(T-1)}(\boldsymbol{z}^{n,i})} \Big\| + \\
&\|G(\boldsymbol{m}^{(T-1)}(\boldsymbol{z}^n)) - G(\boldsymbol{m}^{(T-1)}(\boldsymbol{z}^{n,i}))\| \\
\leq &\frac{2\Big(\sum_{t=1}^{T} \eta^{T-t} \alpha^{(t)}\Big) \operatorname{lip}(r \leftarrow \boldsymbol{w})}{n}.
\end{aligned}
$$

(5) Derive the concentration inequality

$$
\begin{aligned}
&|\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^{n,'})| \\
\leq &\|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \boldsymbol{m}^{(T)}(\boldsymbol{z}^{n,i})\| \\
\leq &\frac{2\Big(\sum_{t=1}^{T} \eta^{T-t} \alpha^{(t)}\Big) \operatorname{lip}(r \leftarrow \boldsymbol{w})}{n} \\
= &\frac{C}{n},
\end{aligned}
$$

where $C = 2\Big(\sum_{t=1}^{T} \eta^{T-t} \alpha^{(t)}\Big) \operatorname{lip}(r \leftarrow \boldsymbol{w})$.

Based on Lemma B.1, $\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n)$ can be bounded as

$$
\begin{aligned}
\mathbb{P}_{\boldsymbol{z}^n}\Big[\big|\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n) - \mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n)\big| \leq \frac{\epsilon}{\sqrt{Q}}\Big] &\geq 1 - 2\exp\big(\frac{-2\epsilon^2}{Q\sum_{i=1}^{n} c_i^2}\big) \\
&= 1 - 2\exp\Big(\frac{-2\epsilon^2 n}{QC^2}\Big).
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&\mathbb{P}_{\boldsymbol{z}^n}[\|\boldsymbol{m}^{(T)}(\boldsymbol{z}^n) - \mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n)\| \leq \epsilon] \\
(a) \quad &\geq \mathbb{P}_{\boldsymbol{z}^n}\left[\bigcap_{q=1}^{Q} \big|\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n) - \mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}_{[q]}^{(T)}(\boldsymbol{z}^n)\big| \leq \frac{\epsilon}{\sqrt{Q}}\right] \\
(b) \quad &\geq 1 - 2Q\exp\Big(\frac{-2\epsilon^2 n}{QC^2}\Big),
\end{aligned}
$$

where inequality (a) is due the relationship between the events; inequality (b) is due to a Frechet inequality. □

C.5 Proof of Lemma 3

First, the definitions of Rademacher complexity, uniform convergence and covering number are introduced. Dudley's Integral Theorem that uses covering number to bound Rademacher complexity is also introduced. Then, by using the Lipschitz constant, the covering number of functional space is shown to be bounded by the covering number of parameter space. Finally, based on Dudley's Integral Theorem, Lemma 3 is shown.

C.5.1 Preliminary

Definition C.1. [5] Let $\boldsymbol{\epsilon}^n = \{\epsilon_1, \dots \epsilon_n\}$ be i.i.d. random variables with $P(\epsilon_i = 1) = P(\epsilon_i = -1) = \frac{1}{2}$. $\boldsymbol{z}^n = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_n\}$ are i.i.d. samples. The *empirical Rademacher complexity* is defined as

$$\hat{\mathrm{Rad}}_n(\mathcal{H}) = \mathbb{E}_{\boldsymbol{\epsilon}^n}\Big[\max_{h \in \mathcal{H}} \frac{1}{n} \sum_i \epsilon_i h(\boldsymbol{z}_i) \Big| \boldsymbol{z}^n\Big];$$

and the *Rademacher complexity* is defined as

$$\mathrm{Rad}(\mathcal{H}) = \mathbb{E}_{\boldsymbol{z}^n}\Big[\hat{\mathrm{Rad}}_n(\mathcal{H})\Big].$$

Theorem C.1. [5] With probability at least $1 - \delta$ the following bound holds:

$$R(h) - R_n(\boldsymbol{z}^n, h) \leq 2\hat{\mathrm{Rad}}_n(\phi \circ \mathcal{H}) + 3\sqrt{\frac{\ln \frac{2}{\delta}}{2n}},$$

where $\phi : \mathbb{R} \to \mathbb{R}$ denotes the loss function $l(h(\boldsymbol{x}); y)$; $\circ$ denotes the composition of functions.

Lemma C.1. [5] Let $\phi : \mathbb{R} \to \mathbb{R}$ be an L-Lipschitz function. Then, for any hypothesis set $\mathcal{H}$ of real-valued functions, *Talagrand's Lemma* indicates the following inequality holds:

$$\hat{\mathrm{Rad}}_n(\phi \circ \mathcal{H}) \leq L\hat{\mathrm{Rad}}_n(\mathcal{H}).$$

Corollary C.1. Suppose $\mathrm{lip}(r \leftarrow h) \leq L$, then with probability at least $1 - \delta$ the following bound holds:

$$R(h) - R_n(\boldsymbol{z}^n, h) \leq 2L\hat{\mathrm{Rad}}_n(\mathcal{H}) + 3\sqrt{\frac{\ln \frac{2}{\delta}}{2n}}$$

Proof. Substituting the result of Lemma C.1 into Theorem C.1 gives the result. □

Definition C.2. [9] An ϵ-*cover* of a subset $\mathcal{U}$ of a metric space $(\mathcal{V}, \rho)$ is a set $\hat{\mathcal{U}} \subseteq \mathcal{U}$ such that for each $\boldsymbol{u} \in \mathcal{U}$ there is a $\hat{\boldsymbol{u}} \in \hat{\mathcal{U}}$ such that $\rho(\boldsymbol{u}, \hat{\boldsymbol{u}}) \leq \epsilon$. The ϵ-*cover number* of $\mathcal{U}$ is

$$N(\epsilon, \mathcal{U}, \rho) = \min\{|\hat{\mathcal{U}}| : \hat{\mathcal{U}} \text{ is an } \epsilon\text{-cover of } \mathcal{U}\}.$$

The following theorem illustrates how to bound the covering number.

Theorem C.2. [9] Let $\mathcal{U} \subseteq \mathcal{V} = \mathbb{R}^D$. Then

$$\Big(\frac{1}{\epsilon}\Big)^D \frac{\mathrm{vol}(\mathcal{U})}{\mathrm{vol}(\mathcal{B})} \leq N(\epsilon, \mathcal{U}, \|\cdot\|) \leq \Big(\frac{\mathrm{vol}(\mathcal{U} + \frac{\epsilon}{2}\mathcal{B})}{\mathrm{vol}(\frac{\epsilon}{2}\mathcal{B})}\Big)$$

where $+$ is the Minkovski sum, $\mathcal{B}$ is the unit norm ball and vol indicates the volume of the set.

Remark: Consider $\mathcal{U} \in \mathbb{R}^D$ with diameter $\mathrm{diam}(\mathcal{U})$. Based on the last inequality, we have

$$N(\epsilon, \mathcal{U}, \|\cdot\|) \leq \Big(\frac{\mathrm{vol}(\mathcal{U} + \frac{\epsilon}{2}\mathcal{B})}{\mathrm{vol}(\frac{\epsilon}{2}\mathcal{B})}\Big) \leq \Big(\frac{\mathrm{diam}(\mathcal{U}) + \epsilon}{\epsilon}\Big)^D = \Big(1 + \frac{\mathrm{diam}(\mathcal{U})}{\epsilon}\Big)^D.$$

Definition C.3. Let $\forall h_1, h_2 \in \mathcal{H}$ be two functions mapping $\boldsymbol{z} \in \mathcal{Z}$ into real value, $\rho_{\mathcal{H}|\boldsymbol{z}^n}$ is defined as follows:

$$\rho_{\mathcal{H}|\boldsymbol{z}^n}(h_1, h_2) = \sqrt{\frac{1}{n}\sum_{i=1}^n (h_1(\boldsymbol{z}_i) - h_2(\boldsymbol{z}_i))^2}.$$

Theorem C.3. [7] With metric $\rho_{\mathcal{H}|\boldsymbol{z}^n}$ on $\mathcal{H}$, *Dudley's integral* indicates

$$\hat{\mathrm{Rad}}_n(\mathcal{H}) \leq 12\int_0^\infty \sqrt{\frac{\log N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}|\boldsymbol{z}^n})}{n}} d\epsilon.$$

Dudley's integral bounds the empirical Rademacher complexity by the covering number of the function space (with a metric based on the difference of the function value on n inputs).

C.5.2 Bound of the covering number of functional space

To start with, another definition of metric in function space is given as follows.

Definition C.4. A metric $\rho_{\mathcal{H}_{\boldsymbol{w}}}$ in parametric function space is defined as follows:

$$\rho_{\mathcal{H}_{\boldsymbol{w}}}(h(\cdot;\boldsymbol{w}_1), h(\cdot;\boldsymbol{w}_2)) = \max_{\boldsymbol{x}\in\mathcal{X}} |h(\boldsymbol{x};\boldsymbol{w}_1) - h(\boldsymbol{x};\boldsymbol{w}_2)|. \tag{3}$$

$\mathrm{lip}(h; \mathcal{H}_{\boldsymbol{w}} \leftarrow \mathcal{W})$ will be written as $\mathrm{lip}(h \leftarrow \boldsymbol{w})$ if $\mathcal{W}$ and $\mathcal{H}_{\boldsymbol{w}}$ are clear from the context:

$$\begin{aligned}\mathrm{lip}(h \leftarrow \boldsymbol{w}) &= \max_{\boldsymbol{w}_1,\boldsymbol{w}_2\in\mathcal{W},\boldsymbol{w}_1\neq\boldsymbol{w}_2} \frac{\rho_{\mathcal{H}_{\boldsymbol{w}}}\Big(h(\cdot;\cdot,\boldsymbol{w}_1), h(\cdot;\cdot,\boldsymbol{w}_2)\Big)}{\rho_{\mathcal{W}}(\boldsymbol{w}_1,\boldsymbol{w}_2)} \\ &= \max_{\boldsymbol{w}_1,\boldsymbol{w}_2\in\mathcal{W},\boldsymbol{w}_1\neq\boldsymbol{w}_2,\boldsymbol{x}} \frac{|h(\boldsymbol{x};\boldsymbol{w}_1) - h(\boldsymbol{x};\boldsymbol{w}_2)|}{\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|}.\end{aligned}$$

Proposition C.2. For all spaces of parametric functions $\mathcal{H}_{\boldsymbol{w}}, \forall\epsilon, \forall\mathcal{H}$,

$$N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}|\boldsymbol{z}^n}) \le N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}_{\boldsymbol{w}}}), \tag{4}$$

where $\boldsymbol{w}$ denotes all parameters of the function, $\rho_{\mathcal{H}|\boldsymbol{z}^n}$ is defined in Definition C.3 and $\rho_{\mathcal{H}_{\boldsymbol{w}}}$ is defined in Definition C.4.

Proof. Let $\{\hat{h}_1, \dots, \hat{h}_N\}$ be an ϵ-covering set in $\mathcal{H}_{\boldsymbol{w}}$ with metric $\rho_{\mathcal{H}_{\boldsymbol{w}}}$, then based on the definition of covering set,

$$\forall h \in \mathcal{H}, \min_j \rho_{\mathcal{H}_{\boldsymbol{w}}}(h, \hat{h}_j) \le \epsilon.$$

Based on the definitions of $\rho_{\mathcal{H}|\boldsymbol{z}^n}$ and $\rho_{\mathcal{H}_{\boldsymbol{w}}}$, we have

$$\begin{aligned}\rho_{\mathcal{H}|\boldsymbol{z}^n}(h, \hat{h}_j) &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(h(\boldsymbol{z}_i) - \hat{h}_j(\boldsymbol{z}_i))^2} \le \sqrt{\frac{1}{n}\sum_{i=1}^{n}\big(\max_{\boldsymbol{z}}|h(\boldsymbol{z}) - \hat{h}_j(\boldsymbol{z})|\big)^2} \\ &= \sqrt{\frac{1}{n}\times n \times \big(\rho_{\mathcal{H}_{\boldsymbol{w}}}(h, \hat{h}_j)\big)^2} = \rho_{\mathcal{H}_{\boldsymbol{w}}}(h, \hat{h}_j) \le \epsilon.\end{aligned}$$

Therefore, $\{\hat{h}_1, \dots, \hat{h}_N\}$ is also an ϵ-covering set of $\mathcal{H}_{\boldsymbol{w}}$ with metric $\rho_{\mathcal{H}|\boldsymbol{z}^n}$ and

$$N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}|\boldsymbol{z}^n}) \le |\{\hat{h}_1, \dots, \hat{h}_N\}| = N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}_{\boldsymbol{w}}}).$$

□

Corollary C.2. The empirical Rademacher complexity can be bounded by the covering number with metric $\rho_{\mathcal{H}_{\boldsymbol{w}}}$ as follows:

$$\hat{\mathrm{Rad}}_n(\mathcal{H}) \le 12\int_0^\infty \sqrt{\frac{\log N(\epsilon, \mathcal{H}, \rho_{\mathcal{H}_{\boldsymbol{w}}})}{n}}d\epsilon.$$

Proof. Substituting the result of Proposition C.2 into Theorem C.3 gives the result. □

Proposition C.3. Let $h(\boldsymbol{z};\boldsymbol{w})$ be a parameterized function and $\boldsymbol{w} \in \mathcal{W} \in \mathbb{R}^Q$. Suppose $\mathrm{lip}(h \leftarrow \boldsymbol{w}) \le L$. Then,

$$N(\epsilon, \mathcal{H}_{\boldsymbol{w}}, \rho_{\mathcal{H}_{\boldsymbol{w}}}) \le N(\epsilon/L, \mathcal{W}, \rho_{\mathcal{W}}) \le \Big(1 + \frac{\mathrm{diam}(\mathcal{W})L}{\epsilon}\Big)^Q.$$

Proof. The second inequality follows from Theorem C.2. We now show the first inequality. Let $\{\hat{\boldsymbol{w}}_1, \dots, \hat{\boldsymbol{w}}_N\}$ be an (ϵ/L)-covering set in $\mathcal{W}$. Based on the definition of covering set,

$$\forall \boldsymbol{w} \in \mathcal{W}, \min_i \rho_{\mathcal{W}}(\boldsymbol{w}, \hat{\boldsymbol{w}}_i) \le \epsilon/L.$$

Based on the definition of Lipschitz constant,

$$\forall h(\cdot;\boldsymbol{w}) \in \mathcal{H}_{\boldsymbol{w}}, \min_i \rho_{\mathcal{H}_{\boldsymbol{w}}}\Big(h(\cdot;\boldsymbol{w}), h(\cdot;\hat{\boldsymbol{w}}_i)\Big) \le L \min_i \rho_{\mathcal{W}}(\boldsymbol{w}, \hat{\boldsymbol{w}}_i) \le \epsilon.$$

Therefore, $\{h(\cdot;\hat{\boldsymbol{w}}_1), \dots, h(\cdot;\hat{\boldsymbol{w}}_N)\}$ is a ϵ-covering set of $\mathcal{H}$ and

$$N(\epsilon, \mathcal{H}(\boldsymbol{w}), \rho_{\mathcal{H}_{\boldsymbol{w}}}) \overset{(c)}{\le} |\{h(\cdot;\hat{\boldsymbol{w}}_1), \dots, h(\cdot;\hat{\boldsymbol{w}}_N)\}| \overset{(d)}{\le} |\{\hat{\boldsymbol{w}}_1, \dots, \hat{\boldsymbol{w}}_N\}| = N(\epsilon/L, \mathcal{W}, \rho_{\mathcal{W}}),$$

where inequality (c) is based on the definition of covering number; inequality (d) is due to the fact that h is a function. □

C.5.3 Proof of Lemma 3

Proof. Based on the result of Corollary C.2,

$$\begin{aligned}
&\hat{\mathrm{Rad}}_n(\mathcal{H}) \\
&\leq 12\int_0^\infty \sqrt{\frac{\log N(\epsilon,\mathcal{H},\rho_{\mathcal{H}_{\boldsymbol{w}}})}{n}}d\epsilon \overset{(a)}{=} 12\int_0^{LB} \sqrt{\frac{\log N(\epsilon,\mathcal{H},\rho_{\mathcal{H}_{\boldsymbol{w}}})}{n}}d\epsilon \\
&\overset{(b)}{\leq} \frac{12}{\sqrt{n}}\int_0^{LB}\sqrt{\log\left(1+\frac{LB}{\epsilon}\right)^Q}d\epsilon \overset{(c)}{=} \frac{12LB}{\sqrt{n}}\int_0^1\sqrt{Q\log\left(1+\frac{1}{\epsilon'}\right)}d\epsilon' \\
&\overset{(d)}{\leq} 12LB\sqrt{\frac{Q}{n}}\int_0^1\sqrt{\log(2/\epsilon')}d\epsilon' \overset{(e)}{=} 24LB\sqrt{\frac{Q}{n}}\int_0^{1/2}\sqrt{\log(1/\epsilon)}d\epsilon.
\end{aligned}$$

Equality (a) holds as the value of h is bounded by LB; if $\epsilon > LB$, then $\log N(\epsilon,\mathcal{H},\rho_{\mathcal{H}_{\boldsymbol{w}}}) = 0$; inequality (b) is based on Proposition C.3; equality (c) follows from variable substitution $\epsilon' = \frac{\epsilon}{LB}$; inequality (d) is due to $\epsilon' \in [0,1]$; equality (e) follows from another variable substitution $\epsilon = \frac{\epsilon'}{2}$.

Then we calculate the integral

$$\begin{aligned}
&\int_0^{1/2}\sqrt{\log(1/\epsilon)}d\epsilon \\
&\overset{(a)}{=} \int_\infty^{\sqrt{log2}} yd(e^{-y^2}) \overset{(b)}{=} e^{-y^2}y|_\infty^{\sqrt{\log 2}} - \int_\infty^{\sqrt{\log 2}} e^{-y^2}dy \\
&= e^{-y^2}y|_\infty^{\sqrt{\log 2}} + \int_{\sqrt{\log 2}}^\infty e^{-y^2}dy \leq e^{-y^2}y|_\infty^{\sqrt{\log 2}} + \int_0^\infty e^{-y^2}dy \\
&= \frac{\sqrt{\log 2}}{2} + \frac{\sqrt{\pi}}{2},
\end{aligned}$$

where equality (a) is based on variable substitution $y = \sqrt{\log(1/\epsilon)}$, i.e. $\epsilon = e^{-y^2}$ and equality (b) is based on integration by parts.

Therefore,

$$\begin{aligned}
\hat{\mathrm{Rad}}_n(\mathcal{H}) &\leq 24LB\sqrt{\frac{Q}{n}}\int_0^{1/2}\sqrt{\log(1/\epsilon)}d\epsilon \\
&\leq 24(\frac{\sqrt{\log 2}}{2} + \frac{\sqrt{\pi}}{2})LB\sqrt{\frac{Q}{n}} \\
&= CLB\sqrt{\frac{Q}{n}},
\end{aligned}$$

where $C = 12(\sqrt{\log 2} + \sqrt{\pi})$.

Finally, substituting the above bound of empirical Rademacher complexity into Corollary C.1 gives Lemma 3. □

C.6 Proof of Theorem 3

Proof. Let $\mathrm{ball}(E,\epsilon) := \mathrm{ball}\left(\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n),\epsilon\right)$ denote the ball with the center at $\mathbb{E}_{\boldsymbol{z}^n}\boldsymbol{m}^{(T)}(\boldsymbol{z}^n)$ and radius of ϵ. Let $L = \mathrm{lip}(r \leftarrow \boldsymbol{w})$. Based on Lemma 2, we have

$$\mathbb{P}_{\boldsymbol{z}^n}[\boldsymbol{m}(\boldsymbol{z}^n) \in \mathrm{ball}(E,\epsilon)] \geq 1 - \delta_1, \tag{5}$$

where $\delta_1 = 2Q\exp(\frac{-2\epsilon^2 n}{Q(2C_2)^2L^2})$, that is $\epsilon = C_2L\sqrt{\frac{2Q}{n}\ln\frac{2Q}{\delta_1}}$.

Based on the result of Lemma 3,

$$\mathbb{P}_{\boldsymbol{z}^n}\left[\max_{\boldsymbol{w}\in\mathrm{ball}(E,\epsilon)}\big(R(\boldsymbol{w}) - R_n(\boldsymbol{z}^n,\boldsymbol{w})\big) \leq CL(2\epsilon)\sqrt{\frac{Q}{n}} + \sqrt{\frac{\ln 1/\delta_2}{2n}}\right] \geq 1 - \delta_2.$$

Substituting $\epsilon = C_2 L\sqrt{\frac{2Q}{n}\ln\frac{2Q}{\delta_1}} \le C_2 L_1 L_l \sqrt{\frac{2Q}{n}\ln\frac{2Q}{\delta_1}}$ into the above formula, we have

$$\mathbb{P}_{\boldsymbol{z}^n}\Big[\max_{\boldsymbol{w}\in \text{ball}(E,\epsilon)} \big(R(\boldsymbol{w}) - R_n(\boldsymbol{z}^n, \boldsymbol{w})\big) \le \epsilon'\Big] \ge 1-\delta_2, \tag{6}$$

where

$$\epsilon' = \frac{2CC_2L_1^2L_l^2Q\sqrt{2\ln(2Q/\delta_1)}}{n} + \sqrt{\frac{\ln(1/\delta_2)}{2n}}.$$

Based on Theorem 2, the final result is obtained by combining Eqs. 5,6 and setting $C_1 = 2\sqrt{2}C$:

$$\mathbb{P}_{\boldsymbol{z}^n}[R(\boldsymbol{m}(\boldsymbol{z}^n)) - R_n(\boldsymbol{z}^n, \boldsymbol{m}(\boldsymbol{z}^n)) \le \epsilon] \ge 1-\delta_1-\delta_2.$$

□

D Lipschitz smoothness and updating equations of SMILE

For a classifier h with convex constraints on parameters, the parameter $\boldsymbol{w}$ will be restricted to be inside a convex set, as explained in Sec. D.1. Then based on Corollary A.1, a sufficient condition for bounded $\text{lip}(\frac{\partial h}{\partial \boldsymbol{w}} \leftarrow \boldsymbol{w})$ is to have finite values of the first and second partial derivatives.

D.1 Equivalence between constrained optimization and the use of regularization terms

Let us review two optimization problems.

Problem 1:

$$\min_{\boldsymbol{w}} R_n(\boldsymbol{z}_n, h_{\boldsymbol{w}}) \quad \text{s.t. } \mathcal{P}(\boldsymbol{w}) \le C;$$

Problem 2:

$$\min_{\boldsymbol{w}} \quad R_n(\boldsymbol{z}_n, h_{\boldsymbol{w}}) + \lambda\mathcal{P}(\boldsymbol{w}).$$

The Lagrange function of Problem 1 is

$$\mathcal{L}(\boldsymbol{w}, u) = R_n(\boldsymbol{z}_n, h_{\boldsymbol{w}}) + u(\mathcal{P}(\boldsymbol{w}) - C), \quad u \ge 0,$$

where u is the Lagrangian multiplier.

For Problem 1, the (KKT) necessary conditions imply

$$\begin{aligned} &\text{Condition 1} \quad \frac{\partial R_n(\boldsymbol{z}_n, h_{\boldsymbol{w}})}{\partial \boldsymbol{w}} + u\frac{\partial \mathcal{P}(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0; \\ &\text{Condition 2} \quad u(\mathcal{P}(\boldsymbol{w}) - C) = 0. \end{aligned}$$

For Problem 2, the necessary condition implies

$$\frac{\partial R_n(\boldsymbol{z}_n, h_{\boldsymbol{w}})}{\partial \boldsymbol{w}} + \lambda\frac{\partial \mathcal{P}(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0.$$

Suppose $\boldsymbol{w}_1^*$ and μ^* satisfy the necessary condition of Problem 1. Setting $\lambda = \mu^*$, we can see that $\boldsymbol{w}_1^*$ satisfies for the necessary condition of Problem 2. Suppose $\boldsymbol{w}_2^*$ satisfies the necessary condition of Problem 2. Setting $\mu = \lambda$ and $C = \mathcal{P}(\boldsymbol{w}_2^*)$, we can see that Condition 1 and Condition 2 of Problem 1 are satisfied, so $\boldsymbol{w}_2^*$ satisfies the necessary condition of Problem 1 as well. Based on the above results, the necessary conditions of Problem 1 and Problem 2 are equivalent.

Meanwhile, when the regularization term in Problem 2 is a convex function, the equivalent Problem 1 constrains $\boldsymbol{w}$ inside the set of $\{\boldsymbol{w}|\mathcal{P}(\boldsymbol{w}) \le C\}$, which is a convex set [1].

D.2 First partial derivative of SMILE classifier

The first partial derivatives of the classifier (Eq. 9) are as follows:

$$\begin{aligned} \frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{r}_j^+} &= -\exp(-\|\boldsymbol{Lx} - \boldsymbol{r}_j^+\|^2)(2\boldsymbol{r}_j^+ - 2\boldsymbol{Lx}) \\ \frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{r}_j^-} &= \exp(-\|\boldsymbol{Lx} - \boldsymbol{r}_k^-\|^2)(2\boldsymbol{r}_j^- - 2\boldsymbol{Lx}) \\ \frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{L}_{[a,b]}} &= -\sum_j 2(\boldsymbol{Lx} - \boldsymbol{r}_j^+)_{[a]}\boldsymbol{x}_{[b]}\exp(-\|\boldsymbol{Lx} - \boldsymbol{r}_j^+\|^2) \\ &\quad + \sum_k 2(\boldsymbol{Lx} - \boldsymbol{r}_k^-)_{[a]}\boldsymbol{x}_{[b]}\exp(-\|\boldsymbol{Lx} - \boldsymbol{r}_k^-\|^2), \end{aligned}$$

where $\boldsymbol{L}_{[a,b]}$ denotes the ath row and bth column element of matrix $\boldsymbol{L}$ and $\boldsymbol{x}_{[a]}$ denotes the ath element of the vector $\boldsymbol{x}$; $(\boldsymbol{Lx}-\boldsymbol{r})_{[a]} = \sum_i \boldsymbol{L}_{[ai]}\boldsymbol{x}_{[i]} - \boldsymbol{r}_{[a]}$.

$\frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{r}_j^+}$ and $\frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{r}_j^-}$ are bounded by $2\,\mathrm{diam}(\boldsymbol{r}_a^-) + 2\,\mathrm{diam}(\boldsymbol{L})\,\mathrm{diam}(\boldsymbol{x})$; $\frac{\partial h(\boldsymbol{x};\mathcal{W})}{\partial \boldsymbol{L}}$ is bounded by $4m(\mathrm{diam}(\boldsymbol{r}) + \mathrm{diam}(\boldsymbol{L})\,\mathrm{diam}(\boldsymbol{x}))\,\mathrm{diam}(\boldsymbol{x})$, where m denotes the number of representative instances. All first partial derivatives have finite values as long as $\mathrm{diam}(\boldsymbol{L})$, $\mathrm{diam}(\boldsymbol{x})$ and $\mathrm{diam}(\boldsymbol{r})$ are bounded.

D.3 Second partial derivative of SMILE classifier

The second partial derivatives are as follows:

$$\begin{aligned}
\frac{\partial^2 h(\boldsymbol{x};\mathcal{W})}{\partial {\boldsymbol{r}_i^+}^2} =& 4\exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_j^+\|^2)(\boldsymbol{r}_j^+ - \boldsymbol{Lx})(\boldsymbol{r}_j^+ - \boldsymbol{Lx})^T - 2\exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_j^+\|^2)\boldsymbol{I};\\
\frac{\partial^2 h(\boldsymbol{x};\mathcal{W})}{\partial {\boldsymbol{r}_j^-}^2} =& -4\exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_k^-\|^2)(\boldsymbol{r}_j^- - \boldsymbol{Lx})(\boldsymbol{r}_j^- - \boldsymbol{Lx})^T + 2\exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_k^-\|^2)\boldsymbol{I};\\
\frac{\partial^2 h(\boldsymbol{x};\mathcal{W})}{\partial {\boldsymbol{L}_{[a,b]}}^2} =& \sum_j 4(\boldsymbol{Lx}-\boldsymbol{r}_j^+)_{[a]}^2 \boldsymbol{x}_{[b]}^2 \exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_j^+\|^2) - 2\sum_j \boldsymbol{x}_{[b]}^2 \exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_j^+\|^2)\\
&- \sum_k 4(\boldsymbol{Lx}-\boldsymbol{r}_k^-)_{[a]}^2 \boldsymbol{x}_{[b]}^2 \exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_k^-\|^2) + 2\sum_k \boldsymbol{x}_{[b]}^2 \exp(-\|\boldsymbol{Lx}-\boldsymbol{r}_k^-\|^2),
\end{aligned}$$

where $\boldsymbol{I}$ is the identity matrix. All second partial derivatives have finite values as long as $\mathrm{diam}(\boldsymbol{L})$, $\mathrm{diam}(\boldsymbol{x})$ and $\mathrm{diam}(\boldsymbol{r})$ are bounded.

D.4 Updating equations of SMILE

The updating equations of SMILE are as follows:

$$\begin{aligned}
\boldsymbol{r}_j^{+,t+1} =& \boldsymbol{r}_j^{+,t} - 2\lambda\alpha\boldsymbol{r}_j^{+,t} + \frac{\alpha}{n}\sum_{i=1}^n y_i l'(y_i h(\boldsymbol{x}_i;\mathcal{W}))\exp(-\|\boldsymbol{Lx}_i - \boldsymbol{r}_j^+\|^2)(2\boldsymbol{r}_j^+ - 2\boldsymbol{Lx}_i)|_{\mathcal{W}^t};\\
\boldsymbol{r}_k^{-,t+1} =& \boldsymbol{r}_k^{-,t} - 2\lambda\alpha\boldsymbol{r}_k^{-,t} - \frac{\alpha}{n}\sum_{i=1}^n y_i l'(y_i h(\boldsymbol{x}_i;\mathcal{W}))\exp(-\|\boldsymbol{Lx}_i - \boldsymbol{r}_k^-\|^2)(2\boldsymbol{r}_k^- - 2\boldsymbol{Lx}_i)|_{\mathcal{W}^t};\\
\boldsymbol{L}^{t+1} =& \boldsymbol{L}^t - 2\lambda\alpha\boldsymbol{L}^t + \frac{\alpha}{n}\sum_{i=1}^n y_i l'(y_i h(\boldsymbol{x}_i;\mathcal{W}))\sum_j \exp(-\|\boldsymbol{Lx}_i - \boldsymbol{r}_j^+\|^2)2(\boldsymbol{Lx}_i - \boldsymbol{r}_j^+)\boldsymbol{x}_i^T|_{\mathcal{W}^t}\\
&- \frac{\alpha}{n}\sum_{i=1}^n y_i l'(y_i h(\boldsymbol{x}_i;\mathcal{W}))\sum_k \exp(-\|\boldsymbol{Lx}_i - \boldsymbol{r}_k^-\|^2)2(\boldsymbol{Lx}_i - \boldsymbol{r}_k^-)\boldsymbol{x}_i^T|_{\mathcal{W}^t}.
\end{aligned}$$

E Data description

Table 1 lists information on sample size and feature dimension, as well as the source of studied datasets.

Table 1: Data description

Dataset	Source	# instances	# features
Australian	UCI	690	14
Cancer	UCI	699	9
Climate	UCI	540	18
Credit	UCI	653	15
German	UCI	1000	24
Haberman	UCI	306	3
Heart	UCI	270	13
ILPD	UCI	583	10
Liver	UCI	345	6
Pima	UCI	768	8
Ringnorm	Delve	7400	20
Twonorm	Delve	7400	20

References

[1] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[2] David Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, pages 78–150, 1992.

[3] John H Hubbard and Barbara Burke Hubbard. *Vector calculus, linear algebra, and differential forms: a unified approach*. Matrix Editions, 2015.

[4] H Quang Minh and Thomas Hofmann. Learning over compact metric spaces. In *International Conference on Computational Learning Theory*, pages 239–254. Springer, 2004.

[5] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT press, 2012.

[6] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[7] N Srebro and K Sridharan. Note on refined Dudley integral covering number bound, 2010.

[8] Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[9] M. J. Wainwright. *High-dimensional Statistics: A Non-asymptotic Viewpoint*. Cambridge University Press, 2019.

[10] Nik Weaver. *Lipschitz Algebras*. World Scientific, 1999.