Multi-Class Kernel Learning: Fast Rate and Algorithms

Anonymous Authors[1]

Abstract

The study on generalization performance of multi-class classification algorithms is one of the fundamental issues in statistical learning theory. In this paper, we study the generalization performance of multi-class classification based on the notion of local Rademacher complexity and obtain a shaper data-dependent generalization error bound with fast convergence rate, substantially improving the state-of-art bounds in the existing data-dependent generalization analysis. The theoretical analysis motivates us to devise two effective multi-class kernel learning algorithms with statistical guarantee and fast convergence rate. Experimental results show that our proposed methods can significantly outperform the existing multi-class classification methods. Theoretical analysis and empirical results demonstrate that our multi-class kernel learning methods based on local Rademacher complexity are sound and effective.

1. Introduction

Multi-class classification is an important problem in various applications, such as natural language processing (Zhang, 2005), information retrieval (Hofmann et al., 2003), computer vision (Deng et al., 2009), web advertising (Beygelzimer et al., 2009), etc. The statistical learning theory of binary classification is by now relatively well developed (Vapnik, 2000; Moh et al., 2012), but there are still numerous statistical challenges to its multi-class extension (Maximov & Reshetova, 2016).

To understand the existing multi-class classification algorithms and guide the development of new ones, people have to investigate the generalization ability of multi-class classification algorithms. A sharper generalization bound usually implies more consistent performance on the training set and the test set. In recent years, some generalization bounds have been proposed to estimate the ability of multi-class classification algorithms based on different measures, such as VC-dimension (Allwein et al., 2000), Natarajan dimension (Daniely & Shalev-Shwartz, 2014), covering Number (Guermeur, 2002; Zhang, 2004; Hill & Doucet, 2007), Rademacher Complexity (Koltchinskii & Panchenko, 2002; Moh et al., 2012; Cortes et al., 2013b), etc. Although there have been several recent advances in the studying of generalization bounds of multi-class classification algorithms, convergence rates of the existing generalization bounds are usually $\mathcal{O}\big(\frac{K^2}{\sqrt{n}}\big)$, where K and n are the number of classes and size of the sample, respectively.

In this paper, we derive a novel data-dependent generalization bound for multi-class classification via the notion of local Rademacher complexity and further devise two effective multi-class kernel learning algorithms based on above theoretical analysis. The rate of this bound is $\mathcal{O}\big(\frac{(\log K)^{2+1/\log K}}{n}\big)$, which substantially improves on the existing data-dependent generalization bounds. Moreover, the proposed multi-class kernel learning algorithms have statistic guarantee and fast convergence rate. Experimental results on benchmark datasets show that our proposed methods can significantly outperform the existing multi-class classification methods. The major contributions of this paper include: 1) A new local Rademacher complexity-based bound of fast convergence rate for multi-class classification is established; 2) Two novel multi-class classification algorithms are proposed with convergence guarantee: a) `Conv-MKL`. Using precomputed kernel matrices regularized by local Rademancher complexity, this method can be implemented by any ℓ_p-norm multi-class MKL solvers; b) `SMSD-MKL`. This method puts local Rademancher complexity in penalized ERM with $\ell_{2,p}$-norm regularizer and multi-class hinge loss function, implemented by stochastic sub-gradient descent by updating dual weights.

2. Related Work

In this section, we introduce related works: multi-class classification bounds, local Rademacher complexity and multi-class kernel learning classification.

2.1. Multi-Class Classification Bounds

Rademacher complexity, VC-dimension, and covering number are three popular tools to derive generalization bounds for multi-class classification:

Rademacher Complexities Bounds. Koltchinskii and Panchenko (2002) and Koltchinskii, Panchenko, and Lozano (2001) first introduced a margin-based bound for multi-class

classification in terms of Rademacher complexity. This bound was slightly improved in (Moh et al., 2012; Cortes et al., 2013b). Maximov and Reshetova (2016) gave a new Rademacher complexity based bound that is linear in the number of classes. Based on the ℓ_p-norm regularization, Lei, Binder, and Klof (2015) introduced a bound with a logarithmic dependence on the number of class size.

Instead of global Rademacher complexity, in this paper, we use local Rademacher complexity to obtain a sharper bound, which substantially improves generalization performance upon existing global Rademancher complexity methods.

VC-dimension Bounds. Allwein, Schapire, and Singer (2000) used the notion of VC-dimension for multi-class learning problems, and derived a VC-dimension based bound. Natarajan dimension was introduced in (Natarajan, 1989) in order to characterize multi-class PAC learnability, which exactly matches the notion of Vapnik-Chervonenkis dimension in the case of binary classification. Daniely and Shalev-Shwartz (2014) derived a risk bound with Natarajan dimension for multi-class classification. VC dimension and Natarajan dimension are the important tools to derive generalization bounds, however these bounds are usually dimension dependent, which makes them hardly applicable to practical large scale problems (such as typical computer vision problems).

Covering Number Bounds. Based on the ℓ_∞-norm covering number bound of linear operators, Guermeur (2002) obtained a generalization bound exhibiting a linear dependence on the class size, which was improved by (Zhang, 2004) to a radical dependence. Hill and Doucet (2007) derived a class-size independent risk guarantee. However, their bound is based on a delicate definition of margin, which is not commonly used in the mainstream multi-class literature.

2.2. Local Rademacher Complexity

One of useful data-dependent complexity measures used in the generalization analysis for traditional binary classification is the notion of (global) Rademacher complexity (Bartlett & Mendelson, 2002). However, it provides global estimates of the complexity of the function class, that is, it does not reflect the fact that the algorithm will likely pick functions that have a small error. In recent years, several authors have applied *local* Rademacher complexity to obtain better generalization error bounds for traditional binary classification (Bartlett et al., 2005; Koltchinskii, 2006). However, numerous statistical challenges remain in multi-class case, and it is still unclear how to use this tool to derive a tighter bound for multi-class. In this paper, we bridge this gap by deriving a sharper generalization bound using local Rademacher complexity.

2.3. Multi-Class Kernel Learning Algorithms

Improvements in multi-class classification has emerged as one of success stories in Multiple Kernel Learning (Zien & Ong, 2007) in which a one-stage multi-class MKL algorithm was presented as a generalization of multi-class loss function (Crammer & Singer, 2002; Tsochantaridis et al., 2004). And Orabona designed stochastic gradient methods, named OBSCURE (Orabona et al., 2010) and UFO-MKL (Orabona & Luo, 2011), which optimize primal versions of equivalent problems.

In this paper, we consider the use of the local Rademacher complexity to devise the novel multi-class classification algorithms, which have statistical guarantee and fast convergence rate.

3. Notations and Preliminaries

We consider multi-class classification problems with $K \geq 2$ classes in this paper. Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{1, 2, \ldots, K\}$ the output space. Assume that we are given a sample $\mathcal{S} = \{z_1 = (\mathbf{x}_1, y_1), \ldots, z_n = (\mathbf{x}_n, y_n)\}$ of size n drawn i.i.d. from a fixed, but unknown probability distribution μ on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. Based on the training examples $\mathcal{S}$, we wish to learn a scoring rule h from a space $\mathcal{H}$ mapping from $\mathcal{Z}$ to $\mathbb{R}$ and use the mapping

$$\mathbf{x} \to \arg\max_{y \in \mathcal{Y}} h(\mathbf{x}, y)$$

to predict. For any hypothesis $h \in \mathcal{H}$, the margin of a labeled example $z = (\mathbf{x}, y)$ is defined as

$$\rho_h(z) := h(\mathbf{x}, y) - \max_{y' \neq y} h(\mathbf{x}, y').$$

The h misclassifies the labeled example $z = (\mathbf{x}, y)$ if $\rho_h(z) \leq 0$ and thus the expected risk incurred from using h for prediction is $L(h) := \mathbb{E}_\mu[1_{\rho_h(z) \leq 0}]$, where $1_{t \leq 0}$ is the 0-1 loss, $1_{t \leq 0} = 1$ if $t \leq 0$, otherwise 0. Since 0-1 loss is hard to handle in learning machines, one ususally considers the proxy loss: such as the the hinge $\ell(t) = (1-t)_+$ and the margin loss $\ell^s(t) = 1_{t \leq 0} + (1 - ts^{-1})1_{0 < t \leq s}$, $s > 0$. In the following, we assume that: 1) $\ell(t)$ bounds the 0-1 loss: $1_{t \leq 0} \leq \ell(t)$; 2) ℓ is decreasing and it has a zero point c_ℓ, i.e., $\ell(c_\ell) = 0$. Note that both hinge loss and margin loss satisfy the above two assumptions.

Any function $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ can be equivalently represented by the vector-valued function $(h_1, \ldots, h_K)$ with $h_j(\mathbf{x}) = h(\mathbf{x}, j), \forall j = 1, \ldots, K$. Let $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a Mercer kernel with ϕ being the associated feature map, i.e., $\kappa(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$. The ℓ_p-norm hypothesis space associated with the kernel κ is denoted by:

$$\mathcal{H}_{p,\kappa} = \Big\{ h_{\mathbf{w},\kappa} = (\langle \mathbf{w}_1, \phi(\mathbf{x}) \rangle, \ldots, \langle \mathbf{w}_K, \phi(\mathbf{x}) \rangle) : \|\mathbf{w}\|_{2,p} \leq 1, 1 \leq p \leq 2 \Big\}, \tag{1}$$

where $\mathbf{w} = (\mathbf{w}_1, \ldots, \mathbf{w}_K)$ and $\|\mathbf{w}\|_{2,p} = \left[\sum_{i=1}^K \|\mathbf{w}_i\|_2^p\right]^{\frac{1}{p}}$ is the $\ell_{2,p}$-norm. For any $p \geq 1$, let q be the dual exponent of p satisfying $\frac{1}{p} + \frac{1}{q} = 1$.

The space of loss function associated with $\mathcal{H}_{p,\kappa}$ is denoted by

$$\mathcal{L} = \{\ell_h := \ell(\rho_h(z)) : h \in \mathcal{H}_{p,\kappa}\}. \tag{2}$$

Let $L(\ell_h)$ and $\hat{L}(\ell_h)$ be expected generalization error and empirical error with respect to ℓ_h:

$$L(\ell_h) := \mathbb{E}_\mu[\ell(\rho_h(z))] \text{ and } \hat{L}(\ell_h) = \frac{1}{n}\sum_{i=1}^n \ell(\rho_h(z_i)).$$

Definition 1 (Rademacher complexity)**.** *Assume $\mathcal{L}$ is a space of loss functions as defined in Equation* (2)*. Then the empirical Rademacher complexity of $\mathcal{L}$ is:*

$$\hat{\mathcal{R}}(\mathcal{L}) := \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\ell_h \in \mathcal{L}} \frac{1}{n}\sum_{i=1}^n \sigma_i \ell_h(z_i)\right],$$

where $\sigma_1, \sigma_2, \ldots, \sigma_n$ is an i.i.d. family of Rademacher variables taking values -1 and 1 with equal probability independent of the sample $\mathcal{S} = (z_1, \ldots, z_n)$. The Rademacher complexity of $\mathcal{L}$ is:

$$\mathcal{R}(\mathcal{L}) = \mathbb{E}_\mu \hat{\mathcal{R}}(\mathcal{L}).$$

Generalization bounds based on the notion of Rademacher complexity for multi-class classification are standard (Koltchinskii & Panchenko, 2002; Koltchinskii et al., 2001; Moh et al., 2012): with probability $1 - \delta$,

$$L(h) \leq \inf_{0<\gamma<1}\left(\hat{L}(h_\gamma) + \mathcal{O}\left(\frac{\mathcal{R}(\mathcal{L})}{\gamma} + \frac{\log 1/\delta}{\sqrt{n}}\right)\right),$$

where $\hat{L}(h_\gamma) = \frac{1}{n}\sum_{i=1}^n \left[1_{\rho_h(z_i)\leq\gamma}\right]$. Since $\mathcal{R}(\mathcal{L})$ is in the order of $\mathcal{O}(\frac{K^2}{\sqrt{n}})$ for various kernel multi-class in practice, so the standard Rademacher complexity bounds converge at rate $\mathcal{O}\left(\frac{K^2}{\sqrt{n}}\right)$, usually.

Although Rademacher complexity is the popular tool to bound generalization, it does not take into consideration the fact that, typically, the hypotheses selected by a learning algorithm have a better performance than in the worst case and belong to a more favorable sub-family of the set of all hypotheses (Cortes et al., 2013a). Therefore, to derive sharper generalization bound, we consider the use of the local Rademacher complexity in this paper. To this end, let $\mathcal{L}^r$ be a star-shaped space of $\mathcal{L}$ with respect to $r > 0$,

$$\mathcal{L}^r = \left\{a\ell_h \middle| a \in [0,1], \ell_h \in \mathcal{L}, L[(a\ell_h)^2] \leq r\right\}, \tag{3}$$

where $L(\ell_h^2) = \mathbb{E}_\mu\left[\ell^2(\rho_h(z))\right]$.

Definition 2 (Local Rademacher Complexity)**.** *For any $r > 0$, the local Rademacher complexity of $\mathcal{L}$ is defined as*

$$\mathcal{R}(\mathcal{L}^r) := \mathcal{R}\left(\left\{a\ell \middle| a \in [0,1], \ell \in \mathcal{L}, L[(a\ell)^2] \leq r\right\}\right).$$

The key idea to obtain sharper generalization error bound is to choose a much smaller class $\mathcal{L}^r \subseteq \mathcal{L}$ with as small a variance as possible, while requiring that the solution is still in $\{h | h \in \mathcal{H}_{p,\kappa}, \ell_h \in \mathcal{L}^r\}$.

In the following, we assume that $\vartheta = \sup_{\mathbf{x}\in\mathcal{X}} \kappa(\mathbf{x}, \mathbf{x}) < \infty$, and $\ell_h : \mathcal{Z} \to [0, d]$, $d > 0$ is a constant.

4. Sharper Generalization Bounds

In this section, we first estimate the local Rademacher complexity, and further derive a sharper generalization bound.

4.1. Local Rademacher Complexity

In this subsection, we will estimate the local Rademacher complexity of multi-class classification.

Theorem 1. *If ℓ is a ζ-smooth loss, i.e., $|\ell'(t) - \ell'(s)| \leq \zeta|t - s|$, then with probability $1 - \delta$, we have*

$$\mathcal{R}(\mathcal{L}^r) \leq \frac{c_{d,\vartheta}\xi(K)\sqrt{\zeta r}\log^{\frac{3}{2}}(n)}{\sqrt{n}} + \frac{4\log(1/\delta)}{n},$$

where

$$\xi(K) = \begin{cases} \sqrt{e}(4\log K)^{1+\frac{1}{2\log K}}, & \text{if } q \geq 2\log K, \\ (2q)^{1+\frac{1}{q}} K^{\frac{1}{q}}, & \text{otherwise}, \end{cases}$$

$c_{d,\vartheta}$ is a constant depends on d and ϑ.

Proof. According to the Lemma 3.6 of (Oneto et al., 2013), with probability $1 - \delta$, we have

$$\mathcal{R}(\mathcal{L}^r) \leq \hat{\mathcal{R}}(\mathcal{L}^r) + \sqrt{\frac{2\log(1/\delta)\mathcal{R}(\mathcal{L}^r)}{n}}.$$

Note that $\forall a, b \geq 0, \sqrt{ab} \leq \frac{a}{2} + \frac{b}{2}$. Thus, we have

$$\mathcal{R}(\mathcal{L}^r) \leq \hat{\mathcal{R}}(\mathcal{L}^r) + \frac{\mathcal{R}(\mathcal{L}^r)}{2} + \frac{\log(1/\delta)}{n}.$$

So, we can obtain that

$$\mathcal{R}(\mathcal{L}^r) \leq 2\hat{\mathcal{R}}(\mathcal{L}^r) + \frac{2\log(1/\delta)}{n} \tag{4}$$

From the Lemma 2.2 of (Srebro et al., 2010), we know that if ℓ is a ζ-smooth loss function,

$$\hat{\mathcal{R}}(\mathcal{L}^r) \leq c_d\sqrt{\zeta r}\log^{\frac{3}{2}}(n)\hat{\mathcal{R}}(\mathcal{L}), \tag{5}$$

where c_d is a constant depends on d. Substituting (5) into (4), we have

$$\hat{\mathcal{R}}(\mathcal{L}^r) \leq 2c_d\sqrt{\zeta r}\log^{\frac{3}{2}}(n)\hat{\mathcal{R}}(\mathcal{L}) + \frac{2\log(1/\delta)}{n}. \quad (6)$$

From Theorem 1 in Appendix A, we have

$$\hat{\mathcal{R}}(\mathcal{L}) \leq \frac{1}{\sqrt{n}} \times \begin{cases} \sqrt{e\vartheta}(4\log K)^{1+\frac{1}{2\log K}}, & \text{if } q \geq 2\log K, \\ \sqrt{\vartheta}(2q)^{1+\frac{1}{q}}c^{\frac{1}{q}}, & \text{otherwise.} \end{cases}$$

Substituting the above result into (6), the proof is over. □

Note that the order of the (global) Rademacher complexity over $\mathcal{L}$ is usually $\mathcal{O}\big(\frac{K^2}{\sqrt{n}}\big)$ for various kernel multi-classes. From Theorem 1, one can see that the order of the local Rademacher complexity is

$$\mathcal{R}(\mathcal{L}^r) = \mathcal{O}\left(\frac{\sqrt{r}\xi(K)}{\sqrt{n}} + \frac{1}{n}\right).$$

Note that $\xi(K)$ is logarithmic dependence on K when $q \geq 2\log K$. For $2 \leq q < 2\log K$, $\xi(K) = \mathcal{O}(K^{\frac{2}{q}})$ which is also substantially milder than the quadratic dependence for Rademacher complexity. If we choose a suitable value of r, the order can even reach $\mathcal{O}\big(\frac{(\log K)^{2+1/\log K}}{n}\big)$ (see in the next subsection), which substantially improves the Rademacher complexity bounds.

4.2. A Sharper Generalization Bound using Local Rademacher Complexity

In this subsection, we will derive a sharper bound for multi-class classification based on the notion of local Rademacher complexity.

Theorem 2. *If ℓ is a ζ-smooth loss. Then, $\forall h \in \mathcal{H}_{p,\kappa}$ and $\forall k > \max(1, \frac{\sqrt{2}}{2d})$, with probability $1-\delta$, we have $L(h) \leq$*

$$\leq \max\left\{\frac{k}{k-1}\hat{L}(\ell_h), \hat{L}(\ell_h) + \frac{c_{d,\vartheta,\zeta,k}\xi^2(K)\log^3 n}{n} + \frac{c_\delta}{n}\right\},$$

where

$$\xi(K) = \begin{cases} \sqrt{e}(4\log K)^{1+\frac{1}{2\log K}}, & \text{if } q \geq 2\log K, \\ (2q)^{1+\frac{1}{q}}K^{\frac{1}{q}}, & \text{otherwise,} \end{cases}$$

$c_{d,\vartheta}$ is a constant depending on d, ϑ, ζ, k, and c_δ is a constant depending on δ.

Proof. From Theorem 2 in Appendix B, with probability $1-\delta$, we have

$$L(\ell_h) \leq \max\left\{\frac{k}{k-1}\hat{L}(\ell_h), \hat{L}(\ell_h) + c_d r^* + \frac{c_\delta}{n}\right\}, \quad (7)$$

where r^* is a fixed point of $\mathcal{R}(\mathcal{L}^r)$. From Lemma 5 (see Appendix C), we know that the $\mathcal{R}(\mathcal{L}^r)$ is a sub-root function, so the fixed point r^* of $\mathcal{R}(\mathcal{L}^r)$ is uniquely exists.

According to Theorem 1, we know that

$$\mathcal{R}(\mathcal{L}^r) \leq \frac{c_{d,\vartheta}\xi(K)\sqrt{\zeta r}\log^{\frac{3}{2}}(n)}{\sqrt{n}} + \frac{4\log(1/\delta)}{n}.$$

Thus, if we set $A = \frac{c_{d,\vartheta}\xi(K)\sqrt{\zeta}\log^{\frac{3}{2}}(n)}{\sqrt{n}}$, $B = \frac{4\log(1/\delta)}{n}$, the fixed point r^* is smaller than the solution of $A\sqrt{r} + B = r$, which is

$$\begin{aligned} r^s &= \frac{2B + A^2 + \sqrt{(2B+A^2)^2 - 4B^2}}{2} \\ &\leq 2B + A^2 = \frac{c_{d,\vartheta}^2\xi^2(K)\zeta\log^3(n)}{n} + \frac{4\log(1/\delta)}{n}. \end{aligned}$$

Substituting the above inequality into (7) finishes the proof. □

The order of the generalization bound in Theorem 2 is $\mathcal{O}\left(\frac{\xi^2(K)}{n}\right)$. From the definition of $\xi(K)$, we can obtain that

$$\mathcal{O}\left(\frac{\xi^2(K)}{n}\right) = \begin{cases} \mathcal{O}\left(\frac{(\log K)^{2+1/\log K}}{n}\right), \text{if } q \geq 2\log K, \\ \mathcal{O}\left(\frac{K^{2/q}}{n}\right), \text{if } 2 \leq q < 2\log K. \end{cases}$$

Note that our bounds is linear dependence on the reciprocal of sample size n, while for the existing data-dependent bounds are all radical dependence. Furthermore, our bounds enjoy a mild dependence on the number of classes. The dependence is polynomial with degree $2/q$ for $2 \leq q < 2\log K$ and becomes logarithmic if $q \geq 2\log K$, which is substantially milder than the quadratic dependence established in (Koltchinskii & Panchenko, 2002; Koltchinskii et al., 2001; Moh et al., 2012; Cortes et al., 2013b).

4.3. Comparison with Related Work

In this section, we will compare our bound with three popular bounds of multi-class classification: Rademacher complexity, covering number and VC-dimension Bounds.

Rademacher Complexity Bounds Currently Rademacher complexity are the most powerful tools to get generalization bounds for multi-class classification. The important property of Rademacher complexity based bounds is that the bounds are applicable in arbitrary Banach spaces and do not depend on the dimension of the feature space directly.

Koltchinskii and Panchenko (2002) and Koltchinskii, Panchenko, and Lozano (2001) introduce a margin-based

bound for multi-class classification in terms of Rademacher complexities:

$$L(h) \leq \inf_{0<\gamma<1} \hat{L}(h_\gamma) + \mathcal{O}\left(\frac{K^2}{\gamma\sqrt{n}} + \frac{\log 1/\delta}{\sqrt{n}}\right).$$

The order is $\mathcal{O}\big(\frac{K^2}{\sqrt{n}}\big)$, which is slightly improved (by a constant factor prior to the Rademacher complexity term) by (Moh et al., 2012; Cortes et al., 2013b). Maximov and Reshetova (2016) give a new Rademacher complexity bound:

$$L(h) \leq \inf_{0<\gamma<1} \hat{L}(h_\gamma) + \mathcal{O}\left(\frac{K}{\gamma\sqrt{n}} + \frac{\log 1/\delta}{\sqrt{n}}\right),$$

which has the form of $\mathcal{O}\big(\frac{K}{\sqrt{n}}\big)$. Based on the ℓ_p-norm regularization, Lei, Binder, and Klof (2015) derive a new bound:

$$L(h) \leq \hat{L}(\ell_h) + \mathcal{O}\left(\frac{\log^2 K}{\sqrt{n}}\right).$$

The existing bounds based on Rademacher complexity are all radical dependence on the reciprocal of sample size.

In this paper, we derive a sharper bound based on the local Rademacher complexity with order $\mathcal{O}\big(\frac{(\log K)^{2+\frac{1}{\log K}}}{n}\big)$, which is substantially sharper than the existing bounds of Rademacher complexity.

VC-dimension Bounds VC-dimension is a important tool to derive the generalization bound for binary classification. Allwein, Schapire, and Singer (2000) show how to use it for multi-class learning problems, and derive a VC-dimension based bounds:

$$L(h) \leq \hat{L}(h_\gamma) + \mathcal{O}\left(\frac{\sqrt{V}\log K}{\sqrt{n}}\right),$$

where V is the VC-dimension. Natarajan dimension is introduced in (Natarajan, 1989) in order to characterize multi-class PAC learnability. Daniely and Shalev-Shwartz (2014) derive a generalization bound with Natarajan dimension:

$$L(h) \leq \hat{L}(h_\gamma) + \mathcal{O}\left(\frac{d_{Nat}}{n}\right),$$

where d_{Nat} is the Natarajan dimension. Note that VC dimension bounds as well as Natarajan dimension bounds are usually dimension dependent, which makes them hardly applicable for practical large scale problems (such as typical computer vision problems).

Covering Number Bounds Based on the ℓ_∞-norm covering number bound of linear operators, Guermeur (2002) obtain a generalization of form $\mathcal{O}\big(\frac{K}{\sqrt{n}}\big)$, which is improved by (Zhang, 2004) to a radical dependence:

$$L(h) \leq \hat{L}(\ell_h) + \mathcal{O}\left(\sqrt{\frac{K}{n}}\right).$$

Hill and Doucet (2007) derive a class-size independent risk guarantee of form $\mathcal{O}\big(\sqrt{1/n}\big)$. However, their bound is based on a delicate definition of margin, which is not commonly used in the mainstream multi-class literature.

The above theoretical analysis indicates that it is a good choice to use the local Rademacher complexity to analyze the generalization ability of multi-class classification.

5. Multi-Class Multiple Kernel Learning

Motivated by the above analysis of generalization bound and convergence rate for multi-class classification, we will exploit properties of the local Rademacher complexity to devise two algorithms for multi-class multiple kernel learning (MC-MKL).

In this paper, we consider the use of multiple kernels,

$$\kappa_{\boldsymbol{\mu}} = \sum_{m=1}^{M} \mu_m \kappa_m.$$

A common approach to multi-class classification is the use of joint feature maps $\phi(\mathbf{x}) : \mathcal{X} \to \mathcal{H}$ (Tsochantaridis et al., 2004). For multiple kernel learning, we have M feature mappings ϕ_m, $m = 1, \ldots, M$ and

$$\kappa_m(\mathbf{x}, \mathbf{x}') = \langle \phi_m(\mathbf{x}), \phi_m(\mathbf{x}') \rangle$$

where $m = 1, \ldots, M$. Let $\phi_{\boldsymbol{\mu}}(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_M(\mathbf{x})]$. Using Theorem 2, to obtain a shaper generalization bound, we confine $q \geq 2\log K$, thus $1 < p \leq \frac{2\log K}{2\log K - 1}$. The ℓ_p hypothesis space of multiple kernels can be written as:

$$\mathcal{H}_{mkl} = \Big\{ h_{\mathbf{w},\kappa_{\boldsymbol{\mu}}} = \left(\langle \mathbf{w}_1, \phi_{\boldsymbol{\mu}}(\mathbf{x}) \rangle, \ldots, \langle \mathbf{w}_K, \phi_{\boldsymbol{\mu}}(\mathbf{x}) \rangle\right), \\ \|\mathbf{w}\|_{2,p} \leq 1, 1 < p \leq \frac{2\log K}{2\log K - 1} \Big\}.$$

5.1. Conv-MKL

The global Rademacher complexity of $\mathcal{H}_{mkl}$ can be bounded by the trace of kernel matrix $\mathbf{K}_{\boldsymbol{\mu}} = \sum_{m=1}^{M} \mathbf{K}_m$. Existing works on (Lanckriet et al., 2002; Bach et al., 2004; Sonnenburg et al., 2006) use the following constraint to $\mathcal{H}_{mkl}$: $\mathrm{Tr}(\mathbf{K}_{\boldsymbol{\mu}}) \leq 1$. According the above theoretical analysis, local Rademancher complexity (the tail sum of the eigenvalues of the kernel) lead to tighter generalization bounds than the global Rademancher complexity (the trace). Thus, we add local Rademancher complexity to restrict $\mathcal{H}_{mkl}$:

$$\mathcal{H}_1 = \Big\{ h_{\mathbf{w},\kappa_{\boldsymbol{\mu}}} \in \mathcal{H}_{mkl} : \sum_{j>\zeta} \lambda_j(\mathbf{K}_{\boldsymbol{\mu}}) \leq 1 \Big\}, \qquad (8)$$

where $\lambda_j(\mathbf{K}_{\boldsymbol{\mu}})$ is the j-th eigenvalues of $\mathbf{K}_{\boldsymbol{\mu}}$, ζ is free parameter removing the ζ largest eigenvalues to control the tail sum. Note that the tail sum is the difference between

the trace and the ζ largest eigenvalues: $\sum_{j>\zeta}\lambda_j(\mathbf{K}_{\boldsymbol{\mu}}) = \mathrm{Tr}(\mathbf{K}_{\boldsymbol{\mu}}) - \sum_{j=1}^{\zeta}\lambda_j(\mathbf{K}_{\boldsymbol{\mu}})$, thus the tail sum can be calculated in $O(n^2\zeta)$ for each kernel.

One can see that $\mathcal{H}_1$ is not convex, and we know that:

$$\sum_{m=1}^{M}\mu_m\sum_{j>\zeta}\lambda_j(\mathbf{K}_m) = \sum_{m=1}^{M}\mu_m/\|\boldsymbol{\mu}\|_1\sum_{j>\zeta}\lambda_j(\|\boldsymbol{\mu}\|_1\mathbf{K}_m) \leq \sum_{j>\zeta}\lambda_j\big(\mathbf{K}_{\boldsymbol{\mu}}\big).$$

Thus, we consider the use of the convex $\mathcal{H}_2$ replace of $\mathcal{H}_1$:

$$\mathcal{H}_2 = \Big\{h_{\mathbf{w},\kappa_{\boldsymbol{\mu}}} \in \mathcal{H}_{mkl} : \sum_{m=1}^{M}\mu_m\sum_{j>\zeta}\lambda_j(\mathbf{K}_m) \leq 1\Big\}. \tag{9}$$

Note that by a renormalization of the kernels $\kappa_1,\ldots,\kappa_M$, according to $\tilde{\kappa}_m := \Big(\sum_{j>\zeta}\lambda_j(\mathbf{K}_m)\Big)^{-1}\kappa_m$ and $\tilde{\kappa}_{\boldsymbol{\mu}} = \sum_{m=1}^{M}\mu_m\tilde{\kappa}_m$, we can simply rewrite $\mathcal{H}_2$ as

$$\mathcal{H}_2 = \Big\{h_{\mathbf{w},\tilde{\kappa}_{\boldsymbol{\mu}}} = \Big(\langle\mathbf{w}_1,\tilde{\phi}_{\boldsymbol{\mu}}(\mathbf{x})\rangle,\ldots,\langle\mathbf{w}_K,\tilde{\phi}_{\boldsymbol{\mu}}(\mathbf{x})\rangle\Big), \|\mathbf{w}\|_{2,p}\leq 1, 1<p\leq\frac{2\log K}{2\log K-1}, \boldsymbol{\mu}\succeq 0, \|\boldsymbol{\mu}\|_1\leq 1\Big\}.$$

which is the commonly studied hypothesis class in multi-class multiple kernel learning. A simpler algorithm in which precomputed kernel matrices are regularized by local Rademancher complexity respectively seen in Algorithm 1:

Algorithm 1 Conv-MKL

Input: precomputed kernel matrices $\mathbf{K}_1,\ldots,\mathbf{K}_M$ and ζ
for $i = 1$ **to** M **do**
  Compute tail sum: $r_m = \sum_{j>\zeta}\lambda_j\left(\mathbf{K}_m\right)$
  Normalize precomputed kernel matrix: $\widetilde{\mathbf{K}}_m = \mathbf{K}_m/r_m$
end for
Use $\widetilde{\mathbf{K}}_m$, $m = 1,\ldots,M$, as the basic kernels in any ℓ_p-norm MKL solver

5.2. SMSD-MKL

In this subsection, we consider a more challenging case. We add local Rademancher complexity in optimization formulation:

$$\min_{\mathbf{w},\boldsymbol{\mu}} \overbrace{\frac{1}{n}\sum_{i=1}^{n}\ell(\mathbf{w},\phi_{\boldsymbol{\mu}}(\mathbf{x}_i),y_i)}^{C(\mathbf{w})} + \underbrace{\frac{\alpha}{2}\|\mathbf{w}\|_{2,p}^2 + \beta\sum_{m=1}^{M}\mu_m r_m}_{\Omega(\mathbf{w})}, \tag{10}$$

Algorithm 2 SMSD-MKL

Input: $\alpha,\beta,\boldsymbol{r},T$
Initialize: $\mathbf{w}^1 = \mathbf{0},\boldsymbol{\theta}^1 = \mathbf{0},\boldsymbol{\mu}^1 = \mathbf{1},q = 2\log K$
for $t = 1$ **to** T **do**
  Sample at random $(\mathbf{x}^t,y^t)$
  Compute the dual weight: $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \partial C(\mathbf{w}^t)$
  $\nu_m^{t+1} = \|\boldsymbol{\theta}_m^{t+1}\| - t\beta r_m, \forall m = 1,\ldots,M$
  $\mu_m^{t+1} = \frac{\mathrm{sgn}(\nu_m^{t+1})|\nu_m^{t+1}|^{q-1}}{\alpha\|\boldsymbol{\theta}_m^{t+1}\|\|\nu_m^{t+1}|_q^{q-2}}, \forall m = 1,\ldots,M$
  $\mathbf{w}_m^{t+1} = \mu_m^{t+1}\boldsymbol{\theta}_m^{t+1}, \forall m = 1,\ldots,M$
end for

where

$$\ell(\mathbf{w},\phi_{\boldsymbol{\mu}}(\mathbf{x}_i),y_i) = \max_{i\neq j}|1-(\langle\mathbf{w}_i,\phi_{\boldsymbol{\mu}}(\mathbf{x}_i)\rangle - \langle\mathbf{w}_j,\phi_{\boldsymbol{\mu}}(\mathbf{x}_j)\rangle)|_+,$$

and $r_m = \sum_{j>\zeta}\lambda_j(\mathbf{K}_m)$ is the tail sum of the m-th kernel matrix, $m = 1,\ldots,M$.

Based on the Stochastic Mirror Descent framework for minimization in (Shalev-Shwartz & Tewari, 2011; Orabona & Luo, 2011), we design a Stochastic Mirror and Sub-gradient Descent Algorithm, called `SMSD-MKL`, to minimize (10), seen in Algorithm 2.

As in the mirror descent algorithm, the algorithm maintain two weight vectors: primal $\mathbf{w}$ and dual $\boldsymbol{\theta}$. Meanwhile, the optimization formulation can be divided into two parts: $C(\mathbf{w})$ to update the dual $\boldsymbol{\theta}$ vector and $\Omega(\mathbf{w})$ to update the primal vector $\mathbf{w}$ by the gradient of the Fenchel dual of Ω. Actually, the algorithm puts the kernel weight $\boldsymbol{\mu}$ aside when updating $\boldsymbol{\theta}$, but $\boldsymbol{\mu}$ is updated together with $\mathbf{w}$ according to a tricky link function given in Theorem 3.

Initially, the algorithm starts with $\mathbf{w}^1 = \mathbf{0}$, $\boldsymbol{\theta}^1 = \mathbf{0}$ and $\boldsymbol{\mu}^1 = \mathbf{1}$. Especially, the algorithm initializes $q = 2\log K$ to make the order of generalization reach $\mathcal{O}\big(\frac{(\log K)^{2+1/\log K}}{n}\big)$, according to Theorem 2. For each iteration, the algorithm randomly samples a training example from train set.

For $C(\mathbf{w})$, the algorithm updates the dual vector with the gradient of $C(\mathbf{w})$. Since hinge loss used in $C(\mathbf{w})$ is not differentiable, the algorithm uses sub-gradient of $z^t = \partial\ell(\mathbf{w}^t,\phi_{\boldsymbol{\mu}}(\mathbf{x}^t),y^t)$ instead, where $\partial\ell(\mathbf{w}^t,\phi_{\boldsymbol{\mu}}(\mathbf{x}^t),y^t)$ is the sub-gradient w.r.t $\mathbf{w}^t$.

For $\Omega(\mathbf{w})$, as in the UFO-MKL (Orabona & Luo, 2011), the algorithm uses $\mathbf{w} = \nabla\Omega^*(\boldsymbol{\theta})$ to update the primal vector $\mathbf{w}$, of which the calculation has been given in Theorem 3.

Actually, the algorithm updates real numbers $\|\boldsymbol{\theta}_m^{t+1}\|$, ν_m^{t+1} and μ_m^{t+1} in scalar products instead of high-dimensional variables $\mathbf{w}^{t+1}$ and $\boldsymbol{\theta}_m^{t+1}$. The $\|\boldsymbol{\theta}_m^{t+1}\|$ can be calculated in an efficient incremental way by scalar values as following:

$$\|\boldsymbol{\theta}_m^{t+1}\| = \|\boldsymbol{\theta}_m^t - z^t\|_2^2 = \|\boldsymbol{\theta}_m^t\|_2^2 - 2\boldsymbol{\theta}_m^t\cdot z^t + \|z^t\|_2^2$$

where $z^t = \partial\ell(\mathbf{w}^t, \phi_{\boldsymbol{\mu}}(\mathbf{x}^t), y^t)$.

Theorem 3. *Let* $\boldsymbol{\nu} = \Big[\|\boldsymbol{\theta}_1\| - \beta r_1, \ldots, \|\boldsymbol{\theta}_M\| - \beta r_M\Big]$, *then the component* m*-th of* $\nabla\Omega^*(\boldsymbol{\theta})$ *is*

$$\frac{\operatorname{sgn}(\nu_m)\boldsymbol{\theta}_m}{\alpha\|\boldsymbol{\theta}_m\|}\frac{|\nu_m|^{q-1}}{\|\boldsymbol{\nu}\|_q^{q-2}},$$

where $\operatorname{sgn}(x)$ *is defined as* $\operatorname{sgn}(x) = 1$, *if* $x > 0$, $\operatorname{sgn}(x) = -1$, *if* $x < 0$, $\operatorname{sgn}(x) \in [-1, +1]$, *if* $x = 0$.

Proof. According to standard Legendre-Fenchel duality, we can get

$$\begin{aligned}\nabla\Omega^*(\boldsymbol{\theta}) &= \arg\max_{\mathbf{w}} \mathbf{w}\cdot\boldsymbol{\theta} - \Omega(\mathbf{w}) \\ &= \arg\max_{\mathbf{w}} \mathbf{w}\cdot\boldsymbol{\theta} - \frac{\alpha}{2}\|\mathbf{w}\|_{2,p}^2 - \beta\boldsymbol{\mu}\cdot\boldsymbol{r}.\end{aligned}$$

To reach the above argmax, the derivative of argmax should be zero, so $\mathbf{w}$ must be proportional to $\boldsymbol{\theta}$. As in UFO-MKL (Orabona & Luo, 2011), we explicitly give a tricky link function $\mathbf{w}_m = \mu_m\boldsymbol{\theta}_m$ for different kernels. By this explicit link function, the algorithm can update both $\mathbf{w}$ and $\boldsymbol{\mu}$ by $\nabla\Omega^*(\boldsymbol{\theta})$. Then, for the convenience of computation, we focus on $c_m = \mu_m\|\boldsymbol{\theta}_m\|$, rewriting the argmax:

$$\arg\min_{\boldsymbol{c}} (\beta\mathbf{r} - \mathbf{a})\cdot\boldsymbol{c} + \frac{\alpha}{2}\|\boldsymbol{c}\|_p^2 \tag{11}$$

where $\mathbf{a} = \Big[\|\boldsymbol{\theta}_1\|, \ldots, \|\boldsymbol{\theta}_M\|\Big]$.

The optimality condition of the above minimization problem (Rockafellar, 1970) states that $\boldsymbol{c}^*$ is an optimal solution of (11). And we set the derivative of above argmin being zero

$$\beta\mathbf{r} - \mathbf{a} + \alpha\boldsymbol{c}^* = 0 \tag{12}$$

And then we can get $\boldsymbol{c}^* = \frac{1}{\alpha}\Big(\mathbf{a} - \beta\mathbf{r}\Big)$. Following similar arguments of Lemma 6 (see in Appendix) and Lemma 7 (see in Appendix), we find that it has a closed-form solution

$$c_m = f^{-1}(c_m^*) = \nabla_m\Big(\frac{1}{2}\|c_m^*\|_q^2\Big) = \frac{\operatorname{sgn}(c_m^*)|c_m^*|^{q-1}}{\alpha\|\boldsymbol{c}^*\|_q^{q-2}}.$$

Let $\boldsymbol{\nu} = \Big[\|\boldsymbol{\theta}_1\| - \beta\mathbf{r}_1, \ldots, \|\boldsymbol{\theta}_M\| - \beta\mathbf{r}_M\Big]$, and use $\mu_m = c_m/\|\boldsymbol{\theta}_m\|$ and $\mathbf{w}_m = \mu_m\boldsymbol{\theta}_m$, We can get

$$\mu_m = \frac{\operatorname{sgn}(\nu_m)|\nu_m|^{q-1}}{\alpha\|\boldsymbol{\theta}_m\|\|\boldsymbol{\nu}\|_q^{q-2}}, \quad \mathbf{w}_m = \frac{\operatorname{sgn}(\nu_m)\boldsymbol{\theta}_m|\nu_m|^{q-1}}{\alpha\|\boldsymbol{\theta}_m\|\|\boldsymbol{\nu}\|_q^{q-2}}.$$

Similar argmax has been analysis in Section 7.2 of (Xiao, 2010). □

5.3. Convergence Rate of Algorithms

5.3.1. CONV-MKL

Convergence rate of the proposed `Conv-MKL` is decided by which ℓ_p MC-MKL algorithm it uses. In experiments, we following implement `Conv-MKL` based on UFO-MKL (Orabona & Luo, 2011). Thus, convergence rate of `Conv-MKL` is same with UFO-MKL in Section 4.1 of (Orabona & Luo, 2011).

5.3.2. SMSD-MKL

Denote by $z^t = \partial\ell(\mathbf{w}, \phi_{\boldsymbol{\mu}}(\mathbf{x}^t), y^t)$, we now state the convergence therem for any loss function that satisfies the following hypothesis

$$\|z_m\| \le L\|\phi(\mathbf{w}_m, y')\|_2, \forall t = 1, \ldots, M, y' \in \mathcal{Y}. \tag{13}$$

The hinge loss for multi-class $\ell(\mathbf{w}, \phi_{\boldsymbol{\mu}}(\mathbf{x}_i), y_i) = \max_{i\neq j}|1 - (\langle\mathbf{w}_i, \phi_{\boldsymbol{\mu}}(\mathbf{x}_i)\rangle - \langle\mathbf{w}_j, \phi_{\boldsymbol{\mu}}(\mathbf{x}_j)\rangle)|_+$ satisfies the hypothesis with $L = \sqrt{2}$.

Theorem 4. *Denote by*

$$f(\mathbf{w}) = \Omega(\mathbf{w}) + \frac{1}{n}\sum_{i=1}^n \ell(\mathbf{w}, \phi_{\boldsymbol{\mu}}(\mathbf{x}_i), y_i)$$

and by $\mathbf{w}^*$ *the solution that minimize* (10). *Suppose that* $\|\phi_m(\mathbf{w}^t, \cdot)\|_2 \le 1$, *and the loss function* ℓ *satisfies* (13). *Let* $\delta \in (0, 1)$, *then with probability at least* $1 - \delta$ *over the choices of the random samples we have that after* T *iterations of the SMSD-MKL algorithm*

$$f(\mathbf{w}^{T+1}) - f(\mathbf{w}^*) \le \frac{eL^2(1 + \log T)\log M}{\alpha\delta T},$$

where e *is the Euler's number.*

Proof. Using (13) and $\|\phi_m(\mathbf{w}^t, y^t)\|_2 \le 1$, we have

$$\begin{aligned}&\|\partial(\mathbf{w}^t, \phi(\mathbf{x}^t, \cdot), y^t)\|_{2,q} \\ &\le LM^{1/q}\max_{j=1,\ldots,M}\|\phi_m(\mathbf{x}^t, \cdot)\|_2 \le LF^{1/q}\end{aligned}$$

The function $\Omega(\mathbf{w}) = \frac{\alpha}{2}\|\mathbf{w}\|_{2,p}^2 + \beta\mu\cdot\mathbf{r}$ is α-strongly convex w.r.t. the norm $\|\cdot\|_{2,q}$. Hence, using according to Theorem 1 in (Shalev-Shwartz et al., 2007), using $\eta = 1$ and $g = \Omega$, and using Markov inequality as in (Shalev-Shwartz et al., 2007) we prove the stated result. □

6. Experiments

In this section, we will compare our proposed `Conv-MKL` (Algorithm 1) and `SMSD-MKL` (Algorithm 2) with 7 popular multi-class classification methods: One-against-One (Knerr et al., 1990), One-against-the-Rest (Bottou et al.,

Table 1. Comparison of average test accuracies of our Conv−MKL and SMSD−MKL with the others. We bold the numbers of the best method, and underline the numbers of the other methods which are not significantly worse than the best one.

	Conv−MKL	SMSD−MKL	LMC	One vs. One	One vs. Rest	GMNP	ℓ_1 MC-MKL	ℓ_2 MC-MKL	UFO-MKL
plant	77.14±2.25	**78.01±2.17**	70.12±2.96	75.83±2.69	75.17±2.68	75.42±3.64	77.60±2.63	75.49±2.48	76.77±2.42
psortPos	74.41±3.35	**76.23±3.39**	63.85±3.94	73.33±4.21	71.70±4.89	73.55±4.22	71.87±4.87	70.70±4.89	74.56±4.04
psortNeg	74.07±2.16	**74.66±1.90**	57.85±2.49	73.74±2.87	71.94±2.50	74.27±2.51	72.83±2.20	72.42±2.65	73.80±2.26
nonpl	**79.15±1.51**	78.69±1.58	75.16±1.48	77.78±1.52	77.49±1.53	78.35±1.46	77.89±1.79	77.95±1.64	78.07±1.56
sector	92.83±2.62	**93.39±0.70**	93.16±0.66	90.61±0.69	93.34±0.61	\	\	92.15±2.57	92.60±0.47
segment	96.79±0.91	**97.62±0.83**	95.07±1.11	97.08±0.61	97.02±0.80	96.87±0.80	96.98±0.64	97.58±0.68	97.20±0.82
vehicle	79.35±2.27	77.28±2.78	75.61±3.56	82.72±1.92	**85.11±1.94**	81.57±2.24	74.96±2.93	76.27±3.15	76.92±2.83
vowel	98.82±1.19	**98.83±5.57**	62.32±4.97	98.12±1.76	98.22±1.83	97.04±1.85	98.27±1.22	97.86±1.75	98.22±1.62
wine	**99.63±0.96**	**99.63±0.96**	97.87±2.80	97.24±3.05	98.14±3.04	97.69±2.43	98.61±1.75	98.52±1.89	99.44±1.13
dna	96.08±0.83	**96.30±0.79**	92.02±1.50	95.89±0.56	95.61±0.73	94.60±0.94	96.27±0.68	95.06±0.92	95.84±0.61
glass	**75.19±5.05**	73.72±5.80	63.95±6.04	71.98±5.75	70.00±5.75	71.24±8.14	69.07±8.08	74.03±6.41	72.46±6.12
iris	96.67±2.94	**97.00±2.63**	88.00±7.82	95.93±3.25	95.87±3.20	95.40±7.34	95.40±6.46	94.00±7.82	95.93±2.88
svmguide2	82.69±5.65	**85.17±3.83**	81.10±4.15	84.79±3.45	84.27±3.03	81.77±3.45	83.16±3.63	83.84±4.21	82.91±3.09
satimage	91.64±0.88	91.78±0.82	84.95±1.15	90.67±0.91	89.29±0.96	89.97±0.81	91.86±0.62	90.43±1.27	**91.92±0.83**

1994), ℓ_1-norm linear multi-class SVM (LMC) (Crammer & Singer, 2002), Generalized Minimal Norm Problem solver (GMNP) (Franc, 2005), the Multiclass MKL (MC-MKL) for ℓ_1-norm and ℓ_2-norm (Zien & Ong, 2007) and mixed-norm MKL solved by stochastic gradient descent (UFO-MKL) (Orabona & Luo, 2011). Actually, we complete comparison tests via implements in LIBSVM [1] (One-against-One and One-against-the-Rest), the DOGMA libary [2] (LMC, GMNP, ℓ_1-nomr and ℓ_2-norm MC-MKL) and the SHOGUN-6.1.3 [3] (UFO-MKL). What's more, we implement our proposed Conv−MKL and SMSD−MKL algorithms based on UFO-MKL.

We experiment on 14 publicly avaliable datasets: four of them evaluated in (Zien & Ong, 2007) (plant, nonpl, psortPos, and psortNeg) [4] and others from LIBSVM Data. For each data set, we use the Gaussian kernel $K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{2\tau}\right)$ as our basic kernels, where $\tau \in 2^i, i = -10, -9, \ldots, 9, 10$. For single Kernel methods (One vs. One, One vs. Rest, GMNP), we choose the kernel which have the highest performance among basic kernels by 10-folds cross-validation. Meanwhile, we use all basic kernels in MKL methods (Conv−MKL, SMSD−MKL, ℓ_1 MC-MKL, ℓ_2 MC-MKL, UFO-MKL). For our algorithms Conv−MKL and SMSD−MKL, we fix the parameter $\zeta = 2$ to curve local Rademancher complexity. The regularization parameterized $\alpha \in 2^i, i = -2, \ldots, 12$ in all algorithms and $\beta \in 10^i, i = -4, \ldots, 1$ in SMSD−MKL are determined by 10-folds cross-validation on training data. Other parameters in compared algorithms follow the same experimental setting in their papers. For each data set, we run all methods 50 times with randomly selected 80% for training and 20% for testing, allowing an estimate of the statistical significance of differences in performance between methods. All statement of statistical significance in the remainder refer to a 95% level of significance under t-test.

The average test accuracies are reported in Table 1. The results show: 1) Our Conv−MKL and SMSD−MKL methods give best results on nearly all data sets except *vehicle* and *satimage*; 2) SMSD−MKL is significantly better than Conv−MKL because it wins on 9 data sets and defeats on 3 data sets; 3) Compared with typical MKL methods, our methods get better results over almost all data sets except that only UFO-MKL works slightly better than ours on *satige*; 4) The MKL methods usually works better than the compared single kernel methods (One vs. One, One vs. Rest and GMNP), but *vehicle* leads fairly good results in single kernel methods but worse results in MKL methods; 5) The kernel based methods have better performance than linear classification machine (LMC) on all data sets.

The above results show that the use of local Rademacher complexity can significantly improve the performance of multi-class multiple kernel learning algorithms, which conforms to our theoretical analysis.

Note: Source code are attached in Supplementary Material.

7. Conclusion

In this paper, we studied the generalization performance of multi-class classification, and derived a sharper data dependent generalization error bound using local Rademacher complexity, which is much sharper than existing data-dependent generalization bounds of multi-class classification. Then, we designed two algorithms with statistical guarantees and fast convergence rates: Conv−MKL and SMSD−MKL. Empirical results show our methods outperform the state-of-the-art multi-class classification methods. Based on local Rademacher complexity, our analysis can be used as a solid basis for the design of new multi-class kernel learning algorithms.

[1] Available at https://www.csie.ntu.edu.tw/ cjlin/libsvmtools/

[2] Available at http://dogma. sourceforge. net

[3] Available at http://www.shogun-toolbox.org/

[4] Available at http://www.raetschlab.org/suppl/protsubloc

References

Allwein, Erin L, Schapire, Robert E, and Singer, Yoram. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of machine learning research*, 1:113–141, 2000.

Bach, Francis R., Lanckriet, Gert R. G., and Jordan, Michael I. Multiple kernel learning, conic duality, and the SMO algorithm. In *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, 2004.

Bartlett, Peter L. and Mendelson, Shahar. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Bartlett, Peter L., Bousquet, Olivier, and Mendelson, Shahar. Local Rademacher complexities. *The Annals of Statistics*, 33(4):1497–1537, 2005.

Beygelzimer, Alina, Langford, John, Lifshits, Yuri, Sorkin, Gregory, and Strehl, Alex. Conditional probability tree estimation analysis and algorithms. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 51–58, 2009.

Bottou, Léon, Cortes, Corinna, Denker, John S, Drucker, Harris, Guyon, Isabelle, Jackel, Lawrence D, LeCun, Yann, Muller, Urs A, Sackinger, Edward, Simard, Patrice, et al. Comparison of classifier methods: a case study in handwritten digit recognition. In *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, pp. 77–82. IEEE, 1994.

Cortes, Corinna, Kloft, Marius, and Mohri, Mehryar. Learning kernels using local Rademacher complexity. In *Advances in Neural Information Processing Systems 25 (NIPS)*, pp. 2760–2768, 2013a.

Cortes, Corinna, Mohri, Mehryar, and Rostamizadeh, Afshin. Multi-class classification with maximum margin multiple kernel. In *Proceedings of the 30th International Conference on Machine Learning (ICML 2013)*, pp. 46–54, 2013b.

Crammer, Koby and Singer, Yoram. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2002.

Daniely, Amit and Shalev-Shwartz, Shai. Optimal learners for multiclass problems. In *Proceedings of The 27th Conference on Learning Theory (COLT 2014)*, pp. 287–316, 2014.

Deng, Jia, Dong, Wei, Socher, Richard, Li, Li-Jia, Li, Kai, and Fei-Fei, Li. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 248–255, 2009.

Franc, Vojtech. Optimization algorithms for kernel methods. *Prague: A PhD dissertation. Czech Technical University*, 2005.

Guermeur, Yann. Combining discriminant models with new multi-class SVMs. *Pattern Analysis & Applications*, 5(2): 168–179, 2002.

Hill, Simon I. and Doucet, Arnaud. A framework for kernel-based multi-category classification. *Journal of Artificial Intelligence Research*, 30:525–564, 2007.

Hofmann, Thomas, Cai, Lijuan, and Ciaramita, Massimiliano. Learning with taxonomies: Classifying documents and words. In *NIPS workshop on syntax, semantics, and statistics*, 2003.

Knerr, Stefan, Personnaz, Léon, and Dreyfus, Gérard. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In *Neurocomputing*, pp. 41–50. Springer, 1990.

Koltchinskii, Vladimir. Local Rademacher complexities and oracle inequalities in risk minimization. *The Annals of Statistics*, 34(6):2593–2656, 2006.

Koltchinskii, Vladimir and Panchenko, Dmitry. Empirical margin distributions and bounding the generalization error of combined classifiers. *The Annals of Statistics*, 30: 1–50, 2002.

Koltchinskii, Vladimir, Panchenko, Dmitriy, and Lozano, Fernando. Some new bounds on the generalization error of combined classifiers. In *Advances in Neural Information Processing Systems (NIPS 2001)*, pp. 245–251, 2001.

Lanckriet, Gert R. G., Cristianini, Nello, Bartlett, Peter L., Ghaoui, Laurent El, and Jordan, Michael I. Learning the kernel matrix with semi-definite programming. In *Machine Learning, Proceedings of the Nineteenth International Conference (ICML 2002), University of New South Wales, Sydney, Australia, July 8-12, 2002*, pp. 323–330, 2002.

Lei, Yunwen, Binder, Urun Doganand Alexander, and Kloft, Marius. Multi-class SVMs: From tighter data-dependent generalization bounds to novel algorithms. In *Advances in Neural Information Processing Systems 27*, pp. 2035–2043, 2015.

Maximov, Yu and Reshetova, Daria. Tight risk bounds for multi-class margin classifiers. *Pattern Recognition and Image Analysis*, 26(4):673–680, 2016.

Moh, Mehryar, Rostamizadeh, Afshin, and Talwalkar, Ameet. *Foundations of machine learning*. MIT press, 2012.

Natarajan, B. K. On learning sets and functions. *Machine Learning*, 4(1):67–97, 1989.

Oneto, Luca, Ghio, Alessandro, Anguita, Davide, and Ridella, Sandro. An improved analysis of the rademacher data-dependent bound using its self bounding property. *Neural Networks*, 44:107–111, 2013.

Orabona, Francesco and Luo, Jie. Ultra-fast optimization algorithm for sparse multi kernel learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML 2011)*, pp. 249–256, 2011.

Orabona, Francesco, Luo, Jie, and Caputo, Barbara. Online-batch strongly convex multi kernel learning. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, pp. 787–794, 2010.

Rockafellar, Ralph Tyrell. *Convex analysis*. Princeton university press, 1970.

Shalev-Shwartz, Shai and Tewari, Ambuj. Stochastic methods for l_1-regularized loss minimization. *Journal of Machine Learning Research*, 12:1865–1892, 2011.

Shalev-Shwartz, Shai, Singer, Yoram, and Srebro, Nathan. Pegasos: Primal estimated sub-gradient solver for SVM. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, pp. 807–814, 2007.

Sonnenburg, Sören, Rätsch, Gunnar, Schäfer, Christin, and Schölkopf, Bernhard. Large scale multiple kernel learning. *Journal of Machine Learning Research*, 7:1531–1565, 2006.

Srebro, Nathan, Sridharan, Karthik, and Tewari, Ambuj. Smoothness, low noise and fast rates. In *Advances in Neural Information Processing Systems 22 (NIPS)*, pp. 2199–2207, 2010.

Tsochantaridis, Ioannis, Hofmann, Thomas, Joachims, Thorsten, and Altun, Yasemin. Support vector machine learning for interdependent and structured output spaces. In *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004)*, pp. 104, 2004.

Vapnik, Vladimir. *The nature of statistical learning theory*. Springer Verlag, 2000.

Xiao, Lin. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.

Zhang, Tong. Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, 5:1225–1251, 2004.

Zhang, Tong. Class-size independent generalization analsysis of some discriminative multi-category classification. In *Advances in Neural Information Processing Systems (NIPS 2005)*, pp. 1625–1632, 2005.

Zien, Alexander and Ong, Cheng Soon. Multiclass multiple kernel learning. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007)*, pp. 1191–1198, 2007.