中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

Multi-Class Learning: From Theory to Algorithm

Jian Li[1,2], Yong Liu[1,*],
Rong Yin[1,2], Hua Zhang[1], Lizhong Ding[5], Weiping Wang[1,3,4]

[1]Institute of Information Engineering, Chinese Academy of Sciences
[2]School of Cyber Security, University of Chinese Academy of Sciences
[3]National Engineering Research Center for Information Security
[4]National Engineering Laboratory for Information Security Technology
[5]Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, UAE

32nd Conference on Neural Information Processing Systems (NIPS 2018)

Content

1. Introduction
2. Notations and Preliminaries
3. Sharper Generalization Bounds
4. Multi-Class Multiple Kernel Learning
5. Experiments
6. Conclusion

Content

1. Introduction
2. Notations and Preliminaries
3. Sharper Generalization Bounds
4. Multi-Class Multiple Kernel Learning
5. Experiments
6. Conclusion

Introduction

- Statistical learning of multi-class classification is a crucial problem in machine learning.
- Existing generalization bounds for multi-class classification:

Methods	`Convergence rate`
VC-dimension	$\mathcal{O}\big(\sqrt{V}\log K/\sqrt{n}\big)$
Natarajan dimension	$\mathcal{O}\big(d_{Nat}/n\big)$
Covering Number	$\mathcal{O}\big(1/\sqrt{n}\big)$
Rademacher Complexity	$\mathcal{O}\big(\log^2 K/\sqrt{n}\big)$
Stability	$\mathcal{O}\big(1/\sqrt{n}\big)$
PAC-Bayesian	$\mathcal{O}\big(\sqrt{\hat{L}(h_\gamma)/n}\big)$

- Contributions:
  - A new local Rademacher complexity based bound with fast convergence rate $\mathcal{O}\big((\log K)^{2+1/\log K}/n\big)$ for multi-class classification is establish.
  - Two novel multi-class multiple kernel learning algorithms are proposed with statistical guarantees: a) `Conv-MKL`. b) `SMSD-MKL`.

Content

1 Introduction

2 Notations and Preliminaries

3 Sharper Generalization Bounds

4 Multi-Class Multiple Kernel Learning

5 Experiments

6 Conclusion

Notations and Preliminaries I

- Multi-class classification setting
  Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{1, 2, \ldots, K\}$ the output space. Based on training examples $\mathcal{S}$ drawn i.i.d. from a fixed, but unknown probability distribution on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, we wish to learn a scoring rule h mapping from $\mathcal{Z}$ to $\mathbb{R}$ to predict

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

  For any $h \in \mathcal{H}$, 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 $\rho_h(z) \leq 0$.

Notations and Preliminaries II

- Hypothesis Space
  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:

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

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 $p \geq 1$, the dual exponent q satisfies $1/p + 1/q = 1$. The space of loss function associated with $\mathcal{H}_{p,\kappa}$ is denoted by

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

Notations and Preliminaries III

- Local Rademacher Complexity

Definition (Local Rademacher Complexity)

For any $r > 0$, the local Rademacher complexity of $\mathcal{L}$ is defined as

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

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

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 $\{h | h \in \mathcal{H}_{p,\kappa}, \ell_h \in \mathcal{L}^r\}$.

- Assumptions
  - $\vartheta = \sup_{\mathbf{x} \in \mathcal{X}} \kappa(\mathbf{x}, \mathbf{x}) < \infty$
  - $\ell_h : \mathcal{Z} \to [0, d]$, $d > 0$ is a constant.

Content

1. Introduction
2. Notations and Preliminaries
3. Sharper Generalization Bounds
4. Multi-Class Multiple Kernel Learning
5. Experiments
6. Conclusion

Local Rademacher Complexity

The estimate the local Rademacher complexity of multi-class classification is given as follows.

Theorem

With probability at least $1-\delta$,

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

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.

A Sharper Generalization Bound I

A sharper bound for multi-class classification based on local Rademacher complexity is derived.

Theorem

$\forall h \in \mathcal{H}_{p,\kappa}$ and $\forall k > \max(1, \frac{\sqrt{2}}{2d})$, with probability at least $1-\delta$, we have

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

where

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

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

A Sharper Generalization Bound II

The order of the generalization bound in above Theorem is $\mathcal{O}\big(\xi^2(K)/n\big)$. 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.

Content

1 Introduction

2 Notations and Preliminaries

3 Sharper Generalization Bounds

4 Multi-Class Multiple Kernel Learning

5 Experiments

6 Conclusion

MKL

Consider multiple kernel leanring, $\kappa_{\boldsymbol{\mu}} = \sum_{m=1}^{M} \mu_m \kappa_m$. 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 above Theorem, 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:

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

Cov-MKL I

According to theoretical analysis, we add local Rademacher complexity (the tail sum of the eigenvalues of the kernel) to restrict $\mathcal{H}_{mkl}$:

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

where $\lambda_j(\mathbf{K}_{\boldsymbol{\mu}})$ is the j-th eigenvalues of $\mathbf{K}_{\boldsymbol{\mu}}$ and ζ is free parameter removing the ζ largest eigenvalues to control the tail sum.
One can see that $\mathcal{H}_1$ is not convex, we consider the use of the convex $\mathcal{H}_2$:

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

Cov-MKL II

Using normalized kernels $\tilde{\kappa}_m = \left(\sum_{j>\zeta} \lambda_j(\mathbf{K}_m)\right)^{-1} \kappa_m$ and $\tilde{\kappa}_{\boldsymbol{\mu}} = \sum_{m=1}^{M} \mu_m \tilde{\kappa}_m$, we can simply rewrite $\mathcal{H}_2$ as

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

With precomputed kernel matrices regularized by local Rademacher complexity, the method gets solution by any ℓ_p-norm MC-MKL solvers.

Cov-MKL III

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

SMSD-MKL I

Considering a more challenging case, we perform penalized ERM over the class $\mathcal{H}_1$, aiming to solve a convex optimization problem with an additional term representing local Rademacher complexity :

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

where $\ell(\mathbf{w},\phi_{\boldsymbol{\mu}}(\mathbf{x}_i),y_i) = \left|1-\left(\langle \mathbf{w}_{y_i},\phi_{\boldsymbol{\mu}}(\mathbf{x}_i)\rangle - \max_{y\neq y_i}\langle \mathbf{w}_y,\phi_{\boldsymbol{\mu}}(\mathbf{x}_i)\rangle\right)\right|_+$ 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 widely used stochastic mirror descent framework, we design a stochastic mirror and sub-gradient descent algorithm with updating dual weights, to solve optimization objective.

SMSD-MKL II

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 $\boldsymbol{\theta}_m^{t+1}$.

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: $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \partial C(\mathbf{w}^t)$
  $\nu_m^{t+1} = \|\boldsymbol{\theta}_m^{t+1}\| - t\beta r_m, \forall m = 1, \ldots, M$
  $\mu_m^{t+1} = \frac{\mathrm{sgn}(\nu_m^{t+1})|\nu_m^{t+1}|^{q-1}}{\alpha\|\boldsymbol{\theta}_m^{t+1}\| \||\nu_m^{t+1}|\|_q^{q-2}}, \forall m = 1, \ldots, M$
end for

By training above algorithm, we can get

(1) Decision Function $\quad \mathbf{x} \rightarrow \arg\max_{y \in \mathcal{Y}} h(\mathbf{x}, y) = \arg\max_{y \in \mathcal{Y}} \mathbf{w}_y \phi_{\boldsymbol{\mu}}(\mathbf{x})$

(2) MKL coefficients $\quad \boldsymbol{\mu} \quad$ for $\quad \kappa_{\boldsymbol{\mu}} = \sum_{m=1}^M \mu_m \kappa_m$

Content

1 Introduction

2 Notations and Preliminaries

3 Sharper Generalization Bounds

4 Multi-Class Multiple Kernel Learning

5 Experiments

6 Conclusion

Experiments

We compare our proposed `Conv-MKL` (Algorithm 1) and `SMSD-MKL` (Algorithm 2) with 7 popular multi-class classification methods.

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	91.34±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	78.72±1.92	79.11±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**

Content

1. Introduction
2. Notations and Preliminaries
3. Sharper Generalization Bounds
4. Multi-Class Multiple Kernel Learning
5. Experiments
6. Conclusion

Conclusion

- A new local Rademacher complexity based bound with fast convergence rate for multi-class classification is establish. Convergence rate is improved from sub-linear to linear

$$\mathcal{O}\Big(\frac{K^2}{\sqrt{n}}\Big) \quad \Rightarrow \quad \mathcal{O}\Big(\frac{(\log K)^{2+1/\log K}}{n}\Big).$$

- Two novel multi-class classification algorithms are proposed with statistical guarantees: a) `Conv-MKL`. b) `SMSD-MKL`.