Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Max-margin based Bayesian classifier*

Tao-cheng HU‡, Jin-hui YU

(*State Key Lab of CAD & CG, Zhejiang University, Hangzhou 310058, China*)

E-mail: hutaocheng@gmail.com; jhyu@cad.zju.edu.cn

Received Mar. 10, 2016; Revision accepted Sept. 14, 2016; Crosschecked Sept. 19, 2016

Abstract: There is a tradeoff between generalization capability and computational overhead in multi-class learning. We propose a generative probabilistic multi-class classifier, considering both the generalization capability and the learning/prediction rate. We show that the classifier has a max-margin property. Thus, prediction on future unseen data can nearly achieve the same performance as in the training stage. In addition, local variables are eliminated, which greatly simplifies the optimization problem. By convex and probabilistic analysis, an efficient online learning algorithm is developed. The algorithm aggregates rather than averages dualities, which is different from the classical situations. Empirical results indicate that our method has a good generalization capability and coverage rate.

Key words: Multi-class learning, Max-margin learning, Online algorithm
http://dx.doi.org/10.1631/FITEE.1601078 **CLC number:** TP181

1 Introduction

For a set of training instances $(\boldsymbol{x}_t, y_t) \in \mathcal{X} \times \mathcal{Y}$, $t = 1, 2, \cdots, T$, from a sample space $\mathcal{X}$ and label space $\mathcal{Y}$, multi-class classification tries to learn a classifier h from a domain $\mathcal{X}$ to a label space $\mathcal{Y}$, where $|\mathcal{Y}| \geq 2$, and the performance of a prediction is measured by the probability that $h(\boldsymbol{x})$ is not the correct label. It is a basic problem in machine learning, surfacing a variety of domains, including object recognition, speech recognition, document categorization, and many more (Daniely and Shalev-Shwartz, 2014).

Over the years, multi-class classification has been subject to intense study, both theoretical and practical. Numerous methods have been developed to tackle this problem. One of the most popular techniques consists of dividing the multi-class problem into the binary-class problem, which is intended to lever well-studied consequences of the binary classifier, such as the estimation of the generalization bound. Then ensemble methods play important roles in connecting binary classifiers and multi-class classifiers (Galar *et al.*, 2011; Cai *et al.*, 2013; Ramaswamy *et al.*, 2014).

Both one-vs-one and one-vs-all (also known as one-vs-rest) are strategies building a multi-class classifier with binary ones. Different from the one-vs-one strategy, which needs $|\mathcal{Y}|(|\mathcal{Y}| - 1)/2$ binary classifiers and thus suffers from a large overhead in both training and prediction, the one-vs-all strategy involves training a single classifier per class, with samples of that class as positive and others as negative. This strategy requires base classifiers produce a real-valued confidence score for the decision, rather than just a class label. However, due to the lack of a unified measure, the scale of confidence values may differ between binary classifiers, and the binary classifiers see an unbalanced distribution because class distribution is unbalanced, or typically the set of negatives they see is much larger than that of positives (Bishop, 2006).

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 61379069), the Major Program of the National Social Science Foundation of China (No. 12&ZD231), and the National Key Technology R&D Program of China (No. 2014BAK09B04)

ORCID: Tao-cheng HU, http://orcid.org/0000-0002-6722-2420

Maximum entropy discrimination (MED) has been proposed to combine strongly complementary properties of the discriminative estimation with Bayesian statistics and graphical models. Compared with classical discrimination frameworks, MED has a distribution on classifiers instead of a single optimal setting (Jebara, 2004). Although MED has shown superior success in diverse domains (Zhu and Xing, 2009; Zhu *et al.*, 2011; Zhu, 2012), its inference algorithm requires careful design (Zhu *et al.*, 2013).

Seeking to fill the gap of the generalization capability and the reasonable computation cost, we propose a generative probabilistic multi-class classifier, with both the efficient learning algorithm and prediction method.

2 Preliminaries

We use symbol $\mathcal{X}$ to represent the sample space, and $\mathcal{Y}=[k]\triangleq\{1,2,\cdots,k\}$ to correspond to the label space which has k classes. The standard basis $\{\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_k\}$ is introduced to transform a label instance to a vector representation. For convenience, we denote the sample variable by $\boldsymbol{x}$, and sample instances by $\{\boldsymbol{x}_t\}$ with a subscript. Similarly, the label variable is denoted by y, while label instances are denoted by $\{y_t\}$.

In classification, a parametric family of decision functions $\mathcal{H}:\mathcal{X}\times\mathcal{W}\mapsto\mathcal{Y}$ is also called classifiers. The classifier (attached with a specific parameter $\boldsymbol{w}\in\mathcal{W}$) works in the following way: given a sample $\boldsymbol{x}\in\mathcal{X}$ as the input, the classifier produces an output $y\in\mathcal{Y}$ to indicate to which class the sample $\boldsymbol{x}$ belongs. More specifically, in the binary situation where $\mathcal{W}$ is dual to $\mathcal{X}$, a popular linear classifier can be formulated as follows:

$$h(\boldsymbol{x};\boldsymbol{w})=\mathrm{sign}(\langle\boldsymbol{w},\boldsymbol{x}\rangle). \tag{1}$$

To obtain the optimal classifier, we are given a training dataset consisting of T pairs $\mathcal{D}=\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_T,y_T)\}$. We wish to find a classifier with the parameter setting $\boldsymbol{w}$ that minimizes some forms of classification errors. Once we have found the best parameter setting $\hat{\boldsymbol{w}}$, we use the classifier to predict the label of a future sample by

$$\hat{y}=h(\boldsymbol{x};\hat{\boldsymbol{w}}). \tag{2}$$

Formally, we use a loss function $\ell(\cdot)$ to measure the classification error. The loss function takes the data point as an input, and the output value is small when label y_t agrees with prediction $h(\boldsymbol{w};\boldsymbol{x})$. The loss function depends on parameter $\boldsymbol{w}$ only through the classification margin. Usually, loss function ℓ is non-decreasing and convex on the margin. A regularization penalty $R(\boldsymbol{w})$ is also introduced in the objective function:

$$\begin{aligned}&\min_{\boldsymbol{w},\gamma_{1:T}}\left[R(\boldsymbol{w})+\frac{1}{T}\sum_{t=1}^{T}\ell(\gamma_t)\right]\\&\text{s.t.}\quad \boldsymbol{e}_{y_t}\cdot h(\boldsymbol{x}_t;\boldsymbol{w})-\gamma_t\geq 0,\quad \forall\{\boldsymbol{x}_t,y_t\}\in\mathcal{D},\end{aligned} \tag{3}$$

which favors certain parameters over others (like prior), where '$\cdot$' denotes the Hadamard product (also known as the Schur product or the entry-wise product), used throughout the paper unless otherwise stated, $\boldsymbol{e}_{y_t}\cdot h(\boldsymbol{x}_t;\boldsymbol{w})$ denotes the margin, and γ_t works as the slack variable in optimization, representing the minimum margin that $\boldsymbol{e}_{y_t}\cdot h(\boldsymbol{x}_t;\boldsymbol{w})$ must satisfy (Jebara, 2004).

3 Probabilistic model for multi-classification

In this section, we start by introducing our Bayesian multi-class classifier model, then give the likelihood-based objective function for optimization, and end up with the analysis of the max-margin property.

3.1 Probabilistic graph

We try to correlate sample variable $\boldsymbol{x}$ and label variable y in a probability model. The core elements of the model are a topic space and its related embedding Emb. The embedding Emb is used to coordinate different dimensions of the input sample. It produces a probability simplex $\boldsymbol{q}\in\mathcal{P}^k$, which aims to correlate the target label variable y with high confidence.

The probabilistic multi-class classifier assumes Algorithm 1 for the sample and label pairs, and the probabilistic graph is shown in Fig. 1.

The embedding Emb is specified by a tensor constructed by the direct product of the k-dimensional real vector space and the dual feature space $\boldsymbol{w}\in\mathbb{R}^d\otimes\mathcal{X}^*$. When $\mathcal{X}\in\mathbb{R}^d$, Emb could be represented by a $k\times d$ matrix. Then the embedding Emb working on a sample variable can be formulated as

$$\mathrm{Emb}(\boldsymbol{x};\boldsymbol{w})=\boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x})\triangleq\boldsymbol{q}, \tag{4}$$

Algorithm 1 Generative correlation multi-class classifier

1: Choose embedding $\boldsymbol{w} \sim \mathcal{N}(\mu = 0, \sigma^2 = 1)$
2: **for** each independent and identically distributed pair $(\boldsymbol{x}, y)$ **do**
3: $\quad \boldsymbol{q} \leftarrow \text{Emb}(\boldsymbol{x}; \boldsymbol{w})$
4: $\quad$ Choose $y \sim \text{Multinomial}(\boldsymbol{q})$
5: **end for**

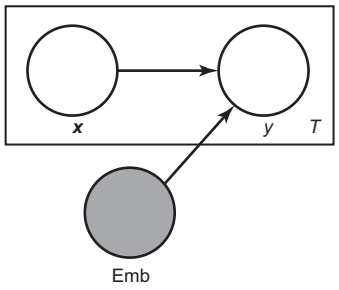

Fig. 1 Probabilistic graph of the generative correlation multi-class classifier (GCMC)

where function $\phi(\cdot)$ denotes the probability assignment, defined as

$$\boldsymbol{\phi}(\boldsymbol{u} \in \mathbb{R}^k) = \frac{\exp(\boldsymbol{u})}{\sum_{j=1}^{k} \exp(\boldsymbol{u}_j)}. \tag{5}$$

We denote the output value by $\boldsymbol{q}$, while those of sample instances are denoted by $\{\boldsymbol{q}_t\}_{t=1}^T$ as before.

3.2 Log-likelihood based objective function

Suppose we are given a set of paired instances $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^T$. The joint distribution of the probabilistic graph could be formulated as

$$\begin{aligned} &p(\{(\boldsymbol{x}_t, y_t)\}_{t=1}^T, \boldsymbol{w}) \\ =&p(\text{Emb} = \boldsymbol{w}) \times \prod_{t=1}^{T} p(\boldsymbol{q}_t|\boldsymbol{w}, \boldsymbol{x}_t) p(y_t|\boldsymbol{q}_t). \end{aligned} \tag{6}$$

Taking the log operator on the joint distribution, we have the following log-likelihood objective function:

$$\begin{aligned} &\max_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}; \{\boldsymbol{x}_t, y_t\}_{t=1}^T) \\ &= -\frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{F}}^2 + \sum_{t=1}^{T} \langle \boldsymbol{e}_{y_t}, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t) \rangle, \end{aligned} \tag{7}$$

where $\|\cdot\|_{\mathcal{F}}$ denotes the Frobenius norm of the matrix.

3.3 On the max-margin property of the probabilistic multi-class classifier

As demonstrated in the previous section, the margin and the loss function play critical roles in the context of max-margin learning. Now, we try to build these concepts for our probabilistic graph model. Noting that the inner product $\langle y_t, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t) \rangle$ does not exceed zero, (1) we denote its Hadamard product form as the margin, and (2) we can define the loss function $\ell(\gamma) = -\gamma$, which is obviously convex. With the given notations, Eq. (7) can be expressed as

$$\max_{\boldsymbol{w}} \left[\frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{F}}^2 + \sum_{t=1}^{T} \ell(\overbrace{\boldsymbol{e}_{y_t} \cdot \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)}^{\text{margin}}) \right]. \tag{8}$$

Then, we show the following theorem:

Theorem 1 (Max-margin) Our proposed model is a max-margin machine.

Proof It is known that a machine is said to be max-margin if and only if its learning objective function has the following form:

$$\begin{aligned} &\min_{\boldsymbol{w}} \left[\frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{F}}^2 + \lambda \sum_{t} \ell(\gamma_t) \right] \\ &\text{s.t.} \quad \boldsymbol{e}_{y_t} \cdot h(\boldsymbol{w}\boldsymbol{x}_t) \geq \gamma_t, \forall t \in \{1, 2, ..., T\}. \end{aligned} \tag{9}$$

It is sufficient to show that Eq. (9) holds by introducing lower bound variables $\{\gamma_t\}_{t=1}^T$ and replacing the margins $\{\boldsymbol{e}_{y_t} \cdot \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)\}_{t=1}^T$ in Eq. (8).

Before moving a step further, we need to introduce one more definition for the Fenchel conjugate function from convex analysis. Given a real named vector space $\mathcal{S}$ and its dual space $\mathcal{S}^*$, the Fenchel conjugate of a function $f : S \mapsto \mathbb{R}$ is defined as

$$f^*(\boldsymbol{x}^* \in \mathcal{S}^*) = \sup_{\boldsymbol{x} \in S} \left[\langle \boldsymbol{x}^*, \boldsymbol{x} \rangle - f(\boldsymbol{x}) \right], \tag{10}$$

which corresponds to an optimization problem.

The Fenchel conjugate has many useful properties, one of which we will use soon is called bijection: if f is closed and convex, then the Fenchel conjugate of f^* is f itself (a function is closed if for all $\alpha > 0$, the level set $\{\boldsymbol{x} : f(\boldsymbol{x}) \leq \alpha\}$ is a closed set) (Boyd and Vandenberghe, 2004; Shalev-Shwartz, 2007).

We turn back to the proof of the necessary condition. For the optimization problem (9), we say the inequality constraints should be equalities;

otherwise, the objective function is not optimal for the non-decreasing property of ℓ. The optimization problem (9) turns into

$$\min_{\boldsymbol{w}} R(\boldsymbol{w}) + \sum_t \ell(\gamma_t), \tag{11}$$
$$\text{s.t. } \boldsymbol{e}_{y_t} \cdot h(\boldsymbol{x}_t; \boldsymbol{w}) = \gamma_t, \quad \forall \{(\boldsymbol{x}_t, y_t)\} \in \mathcal{D},$$

where we introduce T vectors $\{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T\}$, and each $\boldsymbol{\lambda}_t$ is the Lagrangian multiplier of the equality $\langle y_t, h(\boldsymbol{x}_t; \boldsymbol{w})\rangle = \gamma_t$. We obtain the following Lagrangian function:

$$\begin{aligned}&\mathrm{L}(\boldsymbol{w}, \gamma_1, \gamma_2, \cdots, \gamma_T, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T)\\ =&R(\boldsymbol{w}) + \sum_t \left[\ell(\gamma_t) + \langle \boldsymbol{\lambda}_t, \boldsymbol{e}_{y_t} \cdot h(\boldsymbol{x}_t; \boldsymbol{w}) - \gamma_t\rangle\right].\end{aligned} \tag{12}$$

Applying the Fenchel duality theorem (Boyd and Vandenberghe, 2004) on γ and $\boldsymbol{\lambda}$ in turn, we have

$$\begin{aligned}&\min_{\boldsymbol{w},\gamma_1,\gamma_2,\cdots,\gamma_T} \max_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\cdots,\boldsymbol{\lambda}_T} \mathrm{L}(\boldsymbol{w},\gamma_1,\gamma_2,\cdots,\gamma_T,\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\cdots,\boldsymbol{\lambda}_T)\\ =&\min_{\boldsymbol{w}} \max_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\cdots,\boldsymbol{\lambda}_T} \{R(\boldsymbol{w}) + \sum_t [\langle \boldsymbol{\lambda}_t, \boldsymbol{e}_{y_t} \cdot h(\boldsymbol{x}_t; \boldsymbol{w})\rangle - \ell^*(\boldsymbol{\lambda}_t)]\}\\ =&\min_{\boldsymbol{w}} \{R(\boldsymbol{w}) + \sum_t \ell(\boldsymbol{e}_{y_t} \cdot h(\boldsymbol{x}_t; \boldsymbol{w}))\},\end{aligned} \tag{13}$$

where ℓ^* is the Fenchel conjugate function of ℓ. We execute maximization and minimization operations on $\{\gamma_1, \gamma_2, \cdots, \gamma_T\}$ and $\{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T\}$ sequentially. As a result, an unconstrained optimization problem is again obtained.

The proposition leads to a direct conclusion that our model has a similar generalization bound to the support vector machine (SVM) based multiclass classifiers. Moreover, as local variables $\{\gamma_t\}_{t=1}^T$ have been eliminated, there is only one optimization variable $\boldsymbol{w}$ left in objective function (7), and this makes the optimization problem much simpler than the one with constraints, i.e., optimization problem (9).

4 Inference methods

In this section, we seek to develop efficient algorithms for learning and prediction based on analysis of related optimization problems.

4.1 Online learning algorithm with logarithmic regret

To meet the requirement of the online algorithm with logarithmic regret, we need the following convex analysis: given an objective function of a convex optimization problem consisting of two parts, the regularization term and the data term, an online algorithm with logarithmic regret will be proposed if we can prove the convexity of the data term (Shalev-Shwartz and Kakade, 2009; Srebro *et al.*, 2011).

For convenience of analysis, we list a minimization optimization problem which is equivalent to Eq. (7):

$$\min_{\boldsymbol{w}} \left[\overbrace{\frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{F}}^2}^{\text{regularization term}} \quad -\sum_{t=1}^{T} \overbrace{\langle \boldsymbol{e}_{y_t}, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)\rangle}^{\text{data term}}\right], \tag{14}$$

where $\|w\|_{\mathcal{F}}^2$ acts as the regularization term, and $-\langle \boldsymbol{e}_{y_t}, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)\rangle$ acts as the data term. We need to prove the convexity of $-\langle \boldsymbol{e}_{y_t}, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)\rangle$:

Theorem 2 (Concavity) $\langle \boldsymbol{e}_y, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x})\rangle$ is concave on $\boldsymbol{w}$.

Proof We start by noting that $\log \boldsymbol{\phi}$ can be decomposed into two parts:

$$\log \boldsymbol{\phi}(\boldsymbol{u}) = \boldsymbol{u} - \log \sum_k \exp(\boldsymbol{u}_k). \tag{15}$$

There is a common factor $\log \sum_k \exp(\boldsymbol{u}_k)$ for all dimensions of the vector. Taking the partial derivative on $\log \sum_k \exp(\boldsymbol{u}_k)$ with respect to $\boldsymbol{u}$, we can obtain the following equation for the first-order derivative:

$$\mathrm{d} \log \sum_k \exp(\boldsymbol{u}_k) = \langle \boldsymbol{\phi}, \mathrm{d}\boldsymbol{u}\rangle. \tag{16}$$

The second-order derivative is derived using analogy calculus:

$$\mathrm{d}^2 \log \sum_k \exp(\boldsymbol{u}_k) = \boldsymbol{\phi}_i(1 - \boldsymbol{\phi}_j)\mathrm{d}u_i \mathrm{d}u_j. \tag{17}$$

The Hessian matrix is the sum of two positive rank-one matrices, $\boldsymbol{\phi}(1 - \boldsymbol{\phi})^{\mathrm{T}} + (1 - \boldsymbol{\phi})\boldsymbol{\phi}^{\mathrm{T}}$. Thus, $\log \sum_k \exp(\boldsymbol{u}_k)$ is convex. It is obvious that the other part is linear and that $\boldsymbol{e}_y$ is not less than zero. We obtain the conclusion that $\langle \boldsymbol{e}_y, \log \boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x})\rangle$ is concave.

Given this proposition, we design an online learning algorithm with logarithmic regret (Shalev-Shwartz and Kakade, 2009; Srebro *et al.*, 2011), as shown by Algorithm 2.

As the embedding Emb assigns a probability measure for each sample instance, the embedding acts as the prior of probabilities, which corresponds to the Dirichlet parameter in the exponential family

Algorithm 2 Training the generative correlation multi-class classifier with duality averaging

Input: training data $\mathcal{D}=\{(\boldsymbol{x}_t,y_t)\}_{t=1}^T$
Output: embedding Emb^* with parameter $\boldsymbol{w}^*$

$\boldsymbol{w}_0\leftarrow 0$
for $t=1$ to T **do**
  $\boldsymbol{w}_t\leftarrow \boldsymbol{w}_{t-1}+\frac{1}{t}(\boldsymbol{e}_{y_t}-\boldsymbol{\phi}(\boldsymbol{w}_{t-1}^{\mathrm{T}}\boldsymbol{x}_t))\boldsymbol{x}_t^{\mathrm{T}}$
end for
$\boldsymbol{w}^*\leftarrow \boldsymbol{w}_T$

(Blei *et al.*, 2003). Moreover, there is a translation invariance property of $\boldsymbol{\phi}$:

$$\boldsymbol{\phi}(\boldsymbol{u})=\boldsymbol{\phi}(\boldsymbol{u}+s),\quad \boldsymbol{u}\in\mathbb{R}^k, s\in\mathbb{R}. \tag{18}$$

In addition, if we define functions $\{g_t\}_{t=1}^T$ on $\boldsymbol{w}$ as

$$g_t(\boldsymbol{w})=-\langle \boldsymbol{e}_{y_t},\log\boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)\rangle, \tag{19}$$

we obtain the following dual form of optimization problem (14):

$$\max_{\boldsymbol{\mu}_1,\boldsymbol{\mu}_2,\cdots,\boldsymbol{\mu}_T} -\left\{\frac{1}{2}\|\sum_t\boldsymbol{\mu}_t\|_{\mathcal{F}}^2+\sum_t g_t^*(\boldsymbol{\mu}_t)\right\}. \tag{20}$$

Given that $\boldsymbol{w}^*$ is the solution of Eq. (14), and that $\{\boldsymbol{\mu}_t^*\}_{t=1}^T$ is the solution of its dual problem (19), we know there is a relationship between $\boldsymbol{w}^*$ and $\{\boldsymbol{\mu}_t\}$, described as

$$\boldsymbol{w}^*=-\sum_{t=1}^T\boldsymbol{\mu}_t^*. \tag{21}$$

Consequently, we propose another online learning algorithm, and empirical study will show that the new algorithm performs better than duality averaging (Algorithm 2). As the algorithm aggregates rather than averages dualities, we term the algorithm 'duality aggregation'. Details are shown in Algorithm 3.

Algorithm 3 Training the generative correlation multi-class classifier with duality aggregation

Input: training data $\mathcal{D}=\{(\boldsymbol{x}_t,y_t)\}_{t=1}^T$.
Output: embedding Emb^* with parameter $\boldsymbol{w}^*$.

$\boldsymbol{w}_0\leftarrow 0$
for $t=1$ to T **do**
  $\boldsymbol{w}_t\leftarrow \boldsymbol{w}_{t-1}+(\boldsymbol{e}_{y_t}-\boldsymbol{\phi}(\boldsymbol{w}_{t-1}^{\mathrm{T}}\boldsymbol{x}_t))\boldsymbol{x}_t^{\mathrm{T}}$
end for
$\boldsymbol{w}^*\leftarrow \boldsymbol{w}_T$

4.2 Prediction method

After learning, the optimal embedding Emb^* is specified by learned $\boldsymbol{w}^*$. Embedding Emb^* assigns a probability measure for a sample input $\boldsymbol{x}_t$ by $\boldsymbol{q}_t=\boldsymbol{\phi}(\boldsymbol{w}\boldsymbol{x}_t)$. Label prediction is based on the max-entropy principle according to the following optimization problem:

$$\max_{y_t\in\mathcal{Y}}\left[\langle \boldsymbol{e}_{y_t},\log\boldsymbol{q}_t\rangle-\langle \boldsymbol{e}_{y_t},\log\boldsymbol{e}_{y_t}\rangle\right]. \tag{22}$$

The objective function consists of two parts, cross entropy $\langle \boldsymbol{e}_{y_t},\log\boldsymbol{q}_t\rangle$ and entropy of $\boldsymbol{e}_{y_t}$: $\langle \boldsymbol{e}_{y_t},\log\boldsymbol{e}_{y_t}\rangle$, both representing the confidence of $\boldsymbol{e}_{y_t}$ given $\boldsymbol{q}_t$. The solution is $y_t^*=\mathbf{1}(\cdot=\arg\max(\boldsymbol{q}_t))$, where $\mathbf{1}(\cdot=\arg\max(\boldsymbol{q}_t))$ is the indicator function, which means the dimension with the maximum value of $\boldsymbol{q}_t$ would be labeled 1, and others would be labeled 0.

5 Experiments

We present empirical results to demonstrate the prediction accuracy and generalization capability of our model. Since relevant datasets are balanced, here we do not use other performance indicators, by which we mean that the numbers of elements in each class are almost equal. The results demonstrate the merits inherited from both online convex optimization and max-margin learning. The data set is divided into training and test samples. We feed the model h with training samples in the format $(\boldsymbol{x},y)$. After training, we feed the model with the sample instance $\boldsymbol{x}_t$ of the test sample set, and the model returns $\hat{y}_t=h(\boldsymbol{x}_t)$. Then we can compare the prediction and the true label with the expression $\mathrm{evidence}(h;D)=\dfrac{\sum_t 1(\hat{y}_t=y_t)}{\|\text{Test Samples}\|}$, which represents the probability of prediction label $\hat{y}_t$ equaling true label y_t.

Considering the dynamic evolutional performance of algorithms, the prediction accuracy of the two proposed algorithms, duality averaging and duality aggregation, versus the iteration number is evaluated, respectively. Note that our algorithm is very fast—it takes only 3 s to train 10 000 samples with a 1024-dimensional representation on an Intel i5-760 (2.8 GHz)+4 GB machine using Python.

To estimate the generalized capability, we need to estimate the true risk of our model. An incremental procedure training with the whole samples

(including not only the training data, but also the test data) is introduced. We take the prediction accuracy of the whole run procedure as a benchmark, considering that the online algorithm with logarithmic regret produces a nearly optimal model.

In all experiments, duality averaging and duality aggregation execute on training samples with 20 rounds, to obtain an incremental improvement in performance.

For comparison, three popular multi-classification strategies, one-vs-one, one-vs-rest, and error-correcting-output-codes (ECOC) with linear SVM estimators, are introduced. All these multi-classification schemes are simple, robust, and efficient, and can attach an online algorithm for training. We are concerned with the issue of how a mechanism integrates ingredients into a multi-class classifier, while kernel-based SVM and deep neural networks can be seen as one of the strategies which are very similar to our framework, but coming with a complicated hypothesis. We also seek a tradeoff between prediction accuracy and computational overhead, while kernel-based SVM and deep neural networks can have better prediction accuracy, but the computational overhead is huge.

Regarding ECOC, we set the code size with an integer rounding $\log(\|\text{label space}\|)$. These methods are contained in the Scikit-Learn package, providing users with Python interfaces. For a fair comparison, we attach these methods with a stochastic gradient descent (SGD) learning algorithm. We evaluate the performances of three indices, learning/prediction time, and prediction accuracy. The performances are evaluated on three datasets, MNIST (LeCun *et al.*, 1998), COIL-20 (Nene *et al.*, 1996a), and COIL-100 (Nene *et al.*, 1996b), which have been extensively assessed in the context of multi-class learning, and are sufficient to cover the issue whose factors affect the classification results.

5.1 On the effect of data representations

The MNIST database, which has handwritten digits ranging from 0 to 9, consists of 60 000 training examples and 10 000 test examples. The size of each image is 28×28 pixels, with 256 gray levels per pixel; i.e., each image is represented by a 784-dimensional vector.

The relationship between prediction accuracy and the iteration number is shown in Fig. 2, and the performances of different multi-classification methods are shown in the top panel of Table 1. We can see that duality averaging achieves $85.53\% \pm 0.62\%$ prediction accuracy after 20 iterations, duality aggregation achieves $92.55\% \pm 0.12\%$, while the true risk is $93.18\% \pm 0.05\%$, and one-vs-one achieves $91.24\% \pm 0.82\%$. So, we conclude that duality aggregation achieves the best generalization capability.

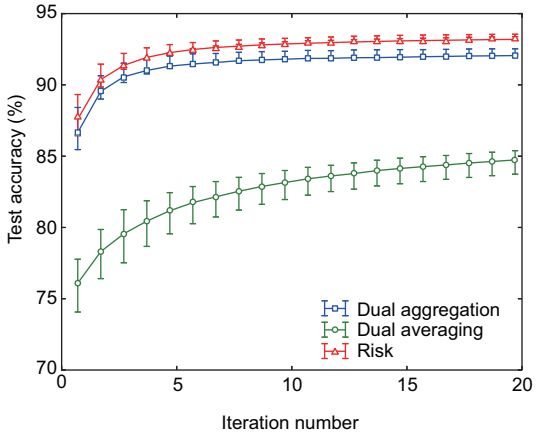

Fig. 2 Evolutions of classification accuracy versus the iteration number on the MNIST dataset

There are studies aimed to show that simple feature learning, such as K-means clustering, can improve the classification accuracy (Rahimi and Recht, 2007; Coates *et al.*, 2011; Agarwal *et al.*, 2014). We extract features on MNIST with principal component analysis (PCA) + radial basis function (RBF) preprocessing (Rahimi and Recht, 2007) to touch on the issue. While the extracted feature has 2048 dimensions after preprocessing, PCA + RBF preprocessing is proposed to map the input data to a randomized feature space, where the inner products of the transformed data are approximately equal to those in the feature space of a user-specified shift-invariant kernel. The related performance is shown in the bottom panel of Table 1, where we can see that due to the PCA + RBF preprocessing, prediction accuracies of all mechanisms are greatly improved. The two controlled experiments clearly show that different representations of features have a huge impact on the classification performance, and this reminds us to choose an appropriate data representation to obtain good performance.

5.2 On the effect of class numbers

The COIL-20 contains 20 objects. The images of each object were taken 5 degrees apart as the object is rotated on a turntable, and each object has 72 images. The size of each image is 32×32 pixels, with 256 gray levels per pixel. Thus, each image is represented by a 1024-dimensional vector. The COIL-100 contains 100 objects. Additional settings are similar to the COIL-20.

As the dataset is not divided into training and test samples, we divide the dataset into training test samples randomly at a ratio 7 : 3. Incremental performance is shown in Fig. 3. Fig. 3a shows that: duality averaging achieves $88.85\% \pm 2.29\%$ prediction accuracy after 20 iterations; duality aggregation achieves $98.00\% \pm 0.66\%$ while the risk is 100% after 5–8 iterations; in COIL-100, duality averaging achieves $55.28\% \pm 1.77\%$ prediction accuracy, and $91.62\% \pm 0.59\%$ while the risk is $98.10\% \pm 0.25\%$. In both COIL-20 and COIL-100, duality aggregation works better than duality averaging. Compared with true risk, duality aggregation has 1.24% more errors in COIL-20, and 6.25% more errors in COIL-100, which may be caused by inadequate learning of some classes since there are 100 classes in COIL-100 and randomized construction of training samples.

We are interested in what will happen when the class number increases. We can see from Table 2 that prediction accuracy decreases as the training time and prediction time increase in all

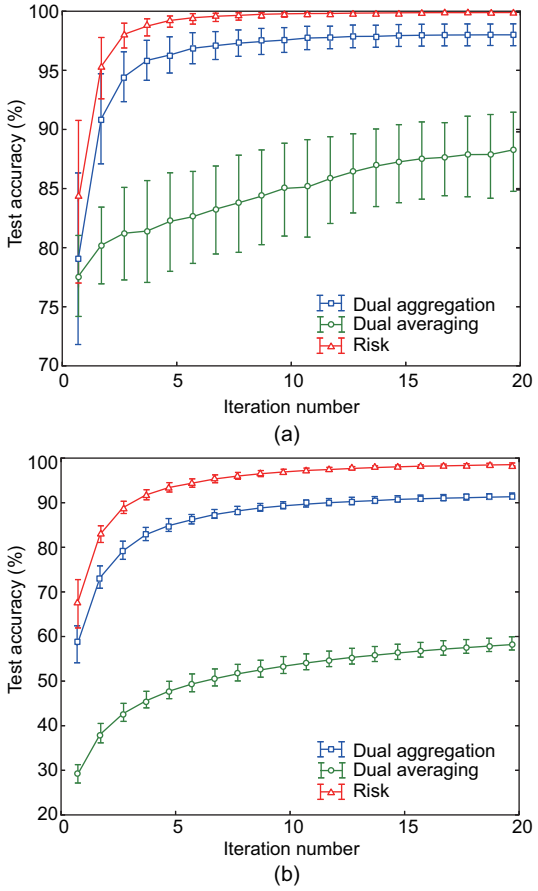

Fig. 3 Evolutions of classification accuracy versus the iteration number on datasets COIL-20 (a) and COIL-100 (b)

Table 1 Performances of various multi-classification schemes on the MNIST dataset with different data representations

Data set	Scheme	Accuracy (%)	Time (s)	
			Training	Prediction
MNIST (raw)	one-vs-one	91.35 ± 0.35	**11.65**	4.09
	one-vs-rest	86.57 ± 1.26	15.30	0.47
	ECOC3	84.57 ± 0.61	48.90	1.30
	ECOC4	85.14 ± 1.00	63.27	1.72
	DAve	85.53 ± 0.62	12.87	0.14
	DAgg	$\mathbf{92.55 \pm 0.12}$	12.87	**0.14**
MNIST (PCA+RBF)	one-vs-one	97.07 ± 0.13	16.40	6.65
	one-vs-rest	95.53 ± 0.08	23.07	0.75
	ECOC3	94.66 ± 0.15	66.93	2.08
	ECOC4	94.89 ± 0.20	97.37	2.08
	DAve	87.58 ± 0.67	16.93	0.15
	DAgg	$\mathbf{97.09 \pm 0.09}$	**12.08**	**0.11**

Bold numbers denote the best performances of the related indicators. ECOCn: ECOC whose code size equals n; DAve: duality averaging; DAgg: duality aggregation

Table 2 Performances of various multi-classification schemes on the COIL dataset with different class numbers

Data set	Scheme	Accuracy (%)	Time (s)	
			Training	Prediction
COIL-20	one-vs-one	98.89 ± 0.50	0.72	0.18
	one-vs-rest	98.08 ± 2.02	0.59	0.01
	ECOC4	97.97 ± 0.97	2.83	0.04
	ECOC5	97.74 ± 0.91	1.16	0.05
	DAve	88.85 ± 2.29	8.54	0.03
	DAgg	$\mathbf{98.00 \pm 0.66}$	**0.68**	**0.01**
COIL-100	one-vs-one	93.33 ± 0.80	41.42	79.89
	one-vs-rest	87.67 ± 1.77	34.87	0.80
	ECOC6	78.02 ± 1.19	352.31	4.92
	ECOC7	77.86 ± 0.97	410.98	5.58
	DAve	55.28 ± 1.77	**14.31**	**0.03**
	DAgg	$\mathbf{91.62 \pm 0.59}$	14.92	**0.03**

Bold numbers denote the best performances of the related indicators. ECOCn: ECOC whose code size equals n; DAve: duality averaging; DAgg: duality aggregation

multi-classification schemes. For vertical comparison, one-vs-one outperforms the other two strategies in prediction accuracy, but its prediction overhead becomes considerable when the label space is large, as its overhead does not vary linearly with the class number. Compared with the introduced schemas, our proposed method has a smooth overhead with the increase of the class number. Duality aggregation achieves the most accurate prediction in the two datasets while the training speed and prediction speed are both quite high. The experiments show that our proposed multi-classification method has a great flexibility under the change of the class number.

6 Conclusions

In this paper, we propose a multi-class learning method, which is aimed to cover the gap between generalization capability and computation overhead. The highlights of our work are concluded as follows:

1. We propose a Bayesian model named the generative correlation multi-class classifier (GCMC) for classification, where a Bayesian network and classification are well integrated as we assign each element with probabilistic semantics.

2. We prove that our model has a max-margin property, obtaining a generalization guarantee like those of other max-margin machines (e.g., SVMs), which means prediction on future unseen data can achieve nearly the same performance as in the training stage.

3. As local variables associated with sample margins are eliminated, there is only one variable $\boldsymbol{w}$ left in the objective function, making the optimization problem much easier.

4. We propose an online learning algorithm termed 'duality aggregation', where the coefficient of the regularization term is derived from the relationship between the data term and the regularized term in the probabilistic graph model. We design experiments to examine which factors affect classification performance, and empirical studies show that our algorithm outperforms many multi-classification frameworks (which also have online algorithm) on many popular datasets.

For further work, we are interested in extending the algorithm with an adaptive learning rate to capture the tradeoff between the regularization term and data term of the objective function in a data-driven manner (Hazan *et al.*, 2007; Duchi *et al.*, 2011). We are also interested in extending the model by, for example, integrating semi-supervised learning (Hu and Yu, 2015; 2016) and feature learning (Coates *et al.*, 2011; Agarwal *et al.*, 2014), to handle data in a large label space.

References

Agarwal, A., Kakade, S.M., Karampatziakis, N., *et al.*, 2014. Least squares revisited: calable approaches for multi-class prediction. Proc. Int. Conf. on Machine Learning, p.541-549.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer, New York, USA.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res.*, **3**(Jan):993-1022.

Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press, Cambridge, UK.

Cai, Q., Yin, Y.F., Man, H., 2013. DSPM: dynamic structure preserving map for action recognition. IEEE Int. Conf. on Multimedia and Expo, p.1-6. http://dx.doi.org/10.1109/ICME.2013.6607606

Coates, A., Lee, H., Ng, A.Y., 2011. An analysis of single-layer networks in unsupervised feature learning. Int. Conf. on Artificial Intelligence and Statistics, p.215-223.

Daniely, A., Shalev-Shwartz, S., 2014. Optimal learners for multiclass problems. Proc. Conf. on Learning Theory, p.287-316.

Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, **12**:2121-2159.

Galar, M., Fernández, A., Barrenechea, E., *et al.*, 2011. An overview of ensemble methods for binary classifiers in multi-class problems: experimental study on one-vs-one and one-vs-all schemes. *Patt. Recogn.*, **44**(8):1761-1776. http://dx.doi.org/10.1016/j.patcog.2011.01.017

Hazan, E., Rakhlin, A., Bartlett, P.L., 2007. Adaptive online gradient descent. *In*: Platt, J.C., Koller, D., Singer, Y., *et al.* (Eds.), Advances in Neural Information Processing Systems 20. MIT Press, Canada, p.65-72.

Hu, T.C., Yu, J.H., 2015. Generalized entropy based semi-supervised learning. IEEE/ACIS Int. Conf. on Computer and Information Science, p.259-263. http://dx.doi.org/10.1109/ICIS.2015.7166603

Hu, T.C., Yu, J.H., 2016. Incremental max-margin learning for semi-supervised multi-class problem. *Stud. Comput. Intell.*, **612**:31-43. http://dx.doi.org/10.1007/978-3-319-23509-7_3

Jebara, T., 2004. Machine learning: discriminative and generative. *In*: Meila, M. (Ed.), the Kluwer International Series in Engineering and Computer Science. Kluwer Academic, Germany.

LeCun, Y., Bottou, L., Bengio, Y., *et al.*, 1998. Gradient-based learning applied to document recognition. *Proc. IEEE*, **86**(11):2278-2324.

Nene, S.A., Nayar, S.K., Murase, H., 1996a. Columbia Object Image Library (COIL-20) Available from http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php [Accessed on Feb. 1, 2016].

Nene, S.A., Nayar, S.K., Murase, H., 1996b. Columbia Object Image Library (COIL-100) Available from http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php [Accessed on Feb. 1, 2016].

Rahimi, A., Recht, B., 2007. Random features for large-scale kernel machines. *In*: Platt, J.C., Koller, D., Singer, Y., *et al.* (Eds.), Advances in Neural Information Processing Systems 20. MIT Press, Canada, p.1177-1184.

Ramaswamy, H.G., Babu, B.S., Agarwal, S., *et al.*, 2014. On the consistency of output code based learning algorithms for multiclass learning problems. Proc. Conf. on Learning Theory, p.885-902.

Shalev-Shwartz, S., 2007. Online learning: theory, algorithms and applications. PhD Thesis, Hebrew University, Jerusalem, Israel.

Shalev-Shwartz, S., Kakade, S.M., 2009. Mind the duality gap: logarithmic regret algorithms for online optimization. *In*: Koller, D., Schuurmans, D., Bengio, Y. (Eds.), Advances in Neural Information Processing Systems 21. MIT Press, Canada, p.1457-1464.

Srebro, N., Sridharan, K., Tewari, A., 2011. On the universality of online mirror descent. *In*: Saul, L.K., Weiss, Y., Bottou, L. (Eds.), Advances in Neural Information Processing Systems 17. MIT Press, Canada, p.2645-2653.

Zhu, J., 2012. Max-margin nonparametric latent feature models for link prediction. Proc. Int. Conf. on Machine Learning, p.719-726.

Zhu, J., Xing, E.P., 2009. Maximum entropy discrimination Markov networks. *J. Mach. Learn. Res.*, **10**(Nov):2531-2569.

Zhu, J., Chen, N., Xing, E.P., 2011. Infinite latent SVM for classification and multi-task learning. *In*: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., *et al.* (Eds.), Advances in Neural Information Processing Systems 24. MIT Press, Canada, p.1620-1628.

Zhu, J., Chen, N., Perkins, H., *et al.*, 2013. Gibbs max-margin topic models with fast sampling algorithms. Proc. Int. Conf. on Machine Learning, p.124-132.