Margin Distribution Logistic Machine

Yi Ding Sheng-Jun Huang Chen Zu Daoqiang Zhang

Abstract

Linear classifier is an essential part of machine learning, and improving its robustness has attracted much effort. Logistic regression (LR) is one of the most widely used linear classifier for its simplicity and probabilistic output. To reduce the risk of overfitting, LR was enhanced by introducing a generalized logistic loss (GLL) with a L2-norm regularization, aiming to maximize the minimum margin. However, the strategy of maximizing minimal margin is less robust to noisy data. In this paper, we incorporate GLL with margin distribution to exploit the statistical information from the training data, and propose a margin distribution logistic machine (MDLM) for better generalization performance and robustness. Furthermore, we extend MDLM to a multi-class version and learn different classes simultaneously by utilizing more information shared across these classes. Extensive experimental results validate the effectiveness of MDLM on both binary classification and multi-class classification.

1 Introduction.

Classification is an essential research issue in machine learning research for its closeness to most real world learning tasks. Linear classifier, the simplest but most widely used classification model, is a useful tool in various applications. It can obtain a comparable performance to non-linear classifiers in a rich dimensional space but with much higher efficiency in training and applying [1]. Because traditional linear classifiers, such as perceptron, least mean squared error (LMSE), logistic regression (LR), could be easily affected by noises and outliers, amendments should be made to improve their generalization performance as well as the robustness.

Robustness of a classifier usually comes from the property of the loss function. A robust loss function should be monotone decreasing and insensitive to outliers. In [3], it discussed different loss functions, and showed that the squared loss or exponential loss is more sensitive to outliers than hinge loss adopted in support vector machine (SVM) [2] or logistic loss in logistic regression (LR). Nevertheless, because hinge loss is non-smooth which may introduce complexity to optimization, logistic loss is a good surrogate for classification. Many works have been proposed concentrating on exploring the capability of logistic loss [19, 20, 21, 22, 23, 24]. Also, Vapnik [2] compared logistic regression (LR) with SVM, and demonstrated that the loss function of LR can be approximated well by the hinge loss in SVM. Furthermore, Zhang and Oles [4] proposed that the generalized logistic loss (GLL) function can approximate soft-margin SVM under some assumptions. So there is a great importance of logistic loss for constructing a simple and robust classifier.

The success of SVM demonstrates that the prior hypothesis of maximizing minimum margin can significantly improve the generalization performance. This prior hypothesis has also been applied to logistic regression. With the help of GLL, Zhang et al. [5] replaced hinge loss in SVM with GLL, and proposed a maximum margin LR to approximate SVM. Further, the maximum margin LR is extended for feature selection [6] and sparse learning [7]. Although the strategy of maximizing the minimum margin is usually effective, it is less robust for tasks with noises.

The margin distribution, which considers the contribution from all data points rather than the one closest to the decision boundary, has been shown to be a better choice than the minimal margin. The study in [8] exhibited that every data point has some contribution to the generalization error, and the relative contribution of a point decays exponentially as a function of its distance from the hyperplane. Based on [8], Garg and Roth [9] proposed Margin Distribution Optimization (MDO) algorithm which gives every data point a weight according to their distance to the discriminative plane. In [10], traditional SVM was proved with a natural lower bound to the scatter between classes. However, it discarded the prior data distribution information within classes, which may be vital for classification. A Maximal Average Margin (MAM) algorithm [11] is also proposed which considers the average margin over all data points and it is proved to be effective. Recently, Gao and Zhou [12] made a great theoretical progress in explaining the performance of boosting algorithms, and proved that maximizing the minimum margin may not necessarily induce a good classifier while the mar-

College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. yiding.nuaa@foxmail.com, {huangsj, chenzu, dqzhang}@nuaa.edu.cn.

gin distribution (margin mean and variance) is crucial to the generalization performance. After that, Zhang and Zhou [13] designed a Large Margin Distribution Machine (LDM) based on the margin distribution theory [12], and achieved significant performance improvement in their experiments.

In this paper, to achieve effective linear classification with better robustness, we propose a margin distribution logistic machine (MDLM) to minimize the logistic loss and exploit margin information over the whole data distribution. Furthermore, we extend MDLM to a multi-class version, which adopts structural constraint to learn the shared information across different classes. Our proposed MDLM model is an unconstrained linear classifier and enjoys the advantages of smoothness, convexity and robustness. Also, it can be extended to many other learning tasks easily.

2 Methodology.

In this section, we will first introduce some notations and related works about our proposed margin distribution logistic machine (MDLM), then the detailed model of MDLM is given for both binary classification and multi-class classification. Finally, a proper optimization algorithm for MDLM is given.

2.1 Notations. Assume the dataset contains N data points $\mathbf{X}=[\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \Re^{D\times N}$ with corresponding labels $\mathbf{Y} = [y_1, y_2, \cdots y_N]^T \in \{-1, 1\}^{N\times 1}$, $\mathbf{w}$ denotes the coefficient vector of a linear classifier which defines a hyperplane in feature space. $\mathbf{1}_N$ denotes a column vector with N entries all set as 1. $\odot$ denotes the Hadamard product. The L1-norm is defined as $\|\mathbf{w}\|_1 = |w_1| + \cdots + |w_D|$. The L2-norm is defined as $\|\mathbf{w}\|_2 = \sqrt{w_1^2 + \cdots + w_D^2}$. The L21-norm is defined as $\|\mathbf{W}\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{m} \mathbf{W}_{i,j}^2}$ and $\mathbf{W} = [\mathbf{w_1}, \mathbf{w_2}, \cdots, \mathbf{w_m}]$.

2.2 Generalized Logistic Loss. Suppose $\mathbf{x} \in \Re^{D\times 1}$ denotes a data point, $y \in \{-1, +1\}$ denotes the corresponding binary class label, and the logistic regression model is given by

$$\Pr(y|\mathbf{x}) = \frac{1}{1+\exp(-y\mathbf{w}^T\mathbf{x})} \tag{2.1}$$

where $\Pr(y|\mathbf{x})$ denotes the conditional probability of label y when given a sample $\mathbf{x}$, and $\mathbf{w}^T\mathbf{x} = 0$ defines a discriminative hyperplane in feature space, where data points in the hyperplane have a conditional probability equals to 0.5. It's worth noting that we suggest all data points should be normalized. Otherwise, $\mathbf{x}$ and $\mathbf{w}$ should be augmented like $\mathbf{x} = [x_1, x_2, \cdots x_D, 1]^T \in \Re^{(D+1)\times 1}$, $\mathbf{w} = [w_1, w_2, \cdots w_D, w_0]^T \in \Re^{(D+1)\times 1}$. To optimize a logistic regression model, we try to minimize the negative log-likelihood of logistic regression with regard to $\mathbf{w}$:

$$\begin{aligned} NLL(\mathbf{w}) &= -\ln \prod_{i=1}^{N} \Pr(y_i|\mathbf{x}_i) \\ &= \sum_{i=1}^{N} \ln\left(1+\exp\left(-y_i\mathbf{w}^T\mathbf{x}_i\right)\right) \end{aligned} \tag{2.2}$$

Equation (2.2) denotes the logistic loss. A modified generalized logistic loss (GLL) is further proposed to approximate the SVM loss:

$$GLL\left(\alpha, y\mathbf{w}^T\mathbf{x}\right) = \frac{1}{\alpha}\ln\left(1+\exp\left(-\alpha\left(y\mathbf{w}^T\mathbf{x}-1\right)\right)\right) \tag{2.3}$$

The difference between GLL and logistic loss is that GLL defines a gap (functional distance) between two classes, the objective function defined by GLL is

$$\begin{aligned} \mathbf{w} &= \arg\min_{\mathbf{w}} \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\alpha}\ln\left(1+\exp\left(-\alpha\left(y_i\mathbf{w}^T\mathbf{x}_i-1\right)\right)\right) \\ &= \arg\min_{\mathbf{w}} \frac{1}{N\alpha}\mathbf{1}_N^T\ln\left[\mathbf{1}_N+\exp\left(-\alpha\left(\mathbf{Y}\odot\mathbf{X}^T\mathbf{w}-\mathbf{1}_N\right)\right)\right] \end{aligned} \tag{2.4}$$

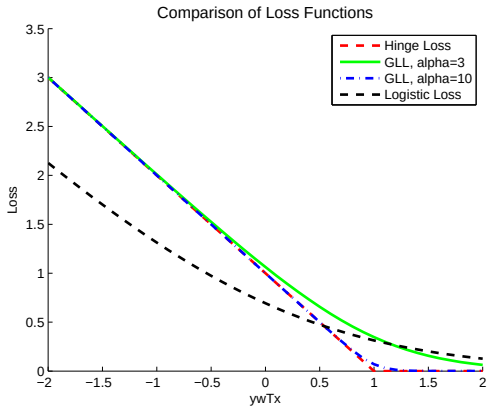


Figure 1: Comparison between different loss functions

The GLL is an adjustable loss function which is controlled by the parameter α. It is proved that it can approximate hinge loss when the parameter α goes larger [5, 7]. As shown in Fig.1, when $\alpha = 10$, the GLL is almost the same to hinge loss. Because hinge loss is a non-smooth loss function and logistic loss does not assign zero penalty to any points which may lead to overfitting, GLL is a better alternative loss function.

2.3 Margin Distribution. As stated in [8, 9], the generalization error of a classifier relies on all samples and the importance of one sample is related to its distance to the hyperplane, thus the margin distribution is proposed. Further in [11], a Maximal Average Margin (MAM) optimality principle demonstrates the superiority of average margin in helping design an effective learning algorithm. Recent progresses [12, 13] point out margin distribution in regard to margin mean and variance is crucial to generalization performance:

THEOREM 2.1. *[12] For any $0 < \delta$, with probability at least $1 - \delta$ over the random choice of sample S with size $m \geq 5$, every voting classifier $\in \mathcal{C}(\mathcal{H})$ satisfies the following bound:*

$$\Pr_D[yf(\mathbf{x}) < 0] \leq \frac{1}{m^{50}} + \inf_{\theta \in (0,1]} \left[\Pr_S[yf(\mathbf{x}) < \theta] + m^{-2/\left(1 - E_S^2[yf(\mathbf{x})] + \theta/9\right)} + \frac{3\sqrt{\mu}}{m^{3/2}} + \frac{7\mu}{3m} + \sqrt{\frac{3\mu}{m}\hat{\mathcal{L}}(\theta)} \right] \tag{2.5}$$

where

$$\mu = 144 \ln m \ln\left(2|\mathcal{H}|\right)/\theta^2 + \ln\left(2|\mathcal{H}|/\delta\right), \tag{2.6}$$

$$\hat{\mathcal{L}}(\theta) = \Pr_S[yf(\mathbf{x}) < \theta] \Pr_S[yf(\mathbf{x}) > 2\theta/3]. \tag{2.7}$$

In (2.7), $E_S[yf(\mathbf{x})]$ is related to margin mean and $\hat{\mathcal{L}}(\theta)$ is related to margin variance.

Theorem 2.1 demonstrates that margin distribution with regard to both margin mean and margin variance is important to the generalization performance. The margin distribution defined in [13] are the first-order and second-order moments of margins, known as margin mean ($\bar{\eta}$) and margin variance ($\hat{\eta}$). The goal of margin distribution is to maximize the margin mean and minimize the margin variance simultaneously:

$$\bar{\eta} = \frac{1}{N}\sum_{i=1}^{N} y_i \mathbf{w}^T \mathbf{x} = \frac{1}{N}(\mathbf{XY})^T \mathbf{w} \tag{2.8}$$

$$\hat{\eta} = \frac{1}{N}\sum_{i=1}^{N} \left(y_i \mathbf{w}^T \mathbf{x}_i - \bar{\eta}\right)^2 \tag{2.9}$$

The effect of margin distribution is significant to the learned hyperplane, and it can help learning algorithm consider more statistical information from the training data.

PROPERTY 2.1. *The margin mean constraint defined in (2.8) will enlarge the distance between the center of two classes and the margin variance constraint defined in (2.9) will force the hyperplane lie in the direction with higher data uncertainty, also the hyperplane is constrained being close to the midpoint of two classes' centers.*

Proof. Denote: N_+ and N_- are the number of positive and negative data samples, S_+ and S_- are the set of positive and negative data samples, $\bar{\mathbf{x}}_+$ and $\bar{\mathbf{x}}_-$ represent the center of positive and negative data samples, $\mathbf{S}_W^+$ and $\mathbf{S}_W^-$ are the covariance matrix of positive and negative class, respectively. As margin mean has an equivalent form:

$$\begin{aligned} \bar{\eta} &= \frac{1}{N}\sum_{i=1}^{N} y_i \mathbf{w}^T \mathbf{x}_i \\ &= \frac{1}{N}\left(\sum_{\mathbf{x}_i \in S_+} \mathbf{w}^T \mathbf{x}_i - \sum_{\mathbf{x}_i \in S_-} \mathbf{w}^T \mathbf{x}_i\right) \\ &= \frac{N_+}{N}\mathbf{w}^T \bar{\mathbf{x}}_+ - \frac{N_-}{N}\mathbf{w}^T \bar{\mathbf{x}}_- \end{aligned} \tag{2.10}$$

The margin variance can be rewritten to

$$\begin{aligned} \hat{\eta} &= \frac{1}{N}\sum_{i=1}^{N} \left(y_i \mathbf{w}^T \mathbf{x}_i - \bar{\eta}\right)^2 \\ &= \frac{1}{N}\left(\sum_{\mathbf{x}_i \in S_+} \left(\mathbf{w}^T \mathbf{x}_i - \bar{\eta}\right)^2 + \sum_{\mathbf{x}_i \in S_-} \left(\mathbf{w}^T \mathbf{x}_i + \bar{\eta}\right)^2\right) \end{aligned} \tag{2.11}$$

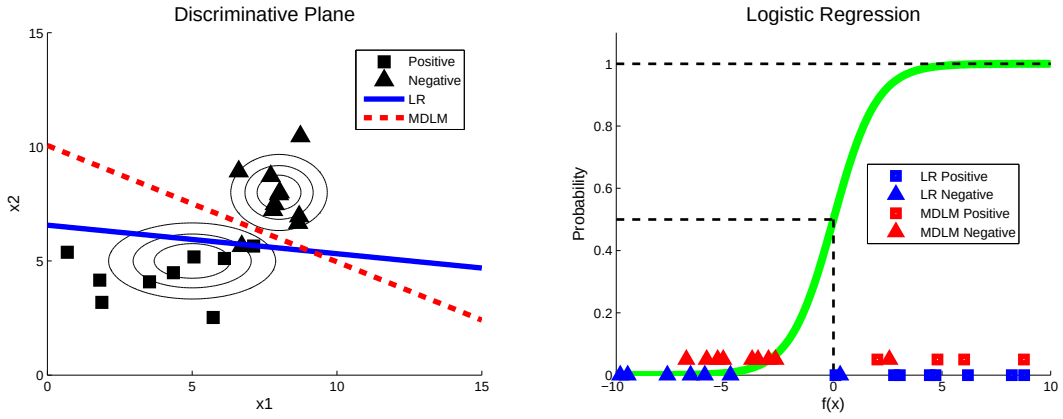
One part of (2.11) is

$$\begin{aligned} &\sum_{\mathbf{x}_i \in S_+} \left(\mathbf{w}^T \mathbf{x}_i - \bar{\eta}\right)^2 \\ &= \sum_{\mathbf{x}_i \in S_+} \left(\mathbf{w}^T \mathbf{x}_i - \left(\frac{N_+}{N}\mathbf{w}^T \bar{\mathbf{x}}_+ - \frac{N_-}{N}\mathbf{w}^T \bar{\mathbf{x}}_-\right)\right)^2 \\ &= N_+ \mathbf{w}^T \mathbf{S}_w^+ \mathbf{w} + \frac{N_+^2 N_-}{N^2}\mathbf{w}^T (\bar{\mathbf{x}}_+ + \bar{\mathbf{x}}_-)^2 \mathbf{w} \end{aligned} \tag{2.12}$$

The margin variance can further be rewritten to

$$\begin{aligned} \hat{\eta} =& \frac{1}{N}\mathbf{w}^T \left(N_+ \mathbf{S}_w^+ + N_- \mathbf{S}_w^-\right)\mathbf{w} \\ &+ \frac{N_+ N_-}{N^2}\mathbf{w}^T (\bar{\mathbf{x}}_+ + \bar{\mathbf{x}}_-)^2 \mathbf{w} \end{aligned} \tag{2.13}$$

For margin mean (2.10), the distance between the centers of two classes is enlarged when maximizing the



(a) Hyperplanes in feature space

(b) Projected points in logistic space

Figure 2: Comparison between LR and MDLM

margin mean. For margin variance, there are two parts in (2.13). The first part denotes the margin variance of each class and it will help the hyperplane lie in the direction with higher data uncertainty. The second part is in proportion to the distance of two class centers' midpoint to the hyperplane, so the hyperplane tends to be close to the midpoint.

According to the property of margin distribution expounded in Property 2.1, by adjusting the weight of margin distribution, the direction and position of hyperplane are going to fit the distribution of training data better. With the help of these statistical information, the model can be robust to noises and outliers. More details of the effect of margin distribution will be discussed in the next section.

2.4 Margin Distribution Logistic Machine. A robust margin distribution logistic machine is defined as:

$$\arg\min_{\mathbf{w}} \frac{1}{N\alpha}\mathbf{1}_N^T \ln\left[\mathbf{1}_N + \exp\left(-\alpha\left(\mathbf{Y}\odot\mathbf{X}^T\mathbf{w} - \mathbf{1}_N\right)\right)\right] + \lambda_1\hat{\eta} - \lambda_2\bar{\eta} \tag{2.14}$$

We give an intuitive explanation of how margin distribution affects logistic regression here. In the objective function (2.14), we minimize the classification error of the GLL. At the same time, we reduce the margin variance and enlarge the margin mean. Margin denotes the functional distance of a point to the hyperplane in feature space, while in logistic space, it corresponds to the distance of a projected point $\mathbf{w}^T\mathbf{x}$ to the zero point where the corresponding probability is 0.5 and the slope is the maximum here. It is clear that points projected into the small interval around 0 in logistic space are easy to be misclassified. When we optimize the margin distribution, like Fig.2(b) shows, there is a larger safety gap between two classes in logistic space. The corresponding hyperplane learned by MDLM in feature space is shown in Fig.2(a), the hyperplane lies in the direction where points have larger uncertainty, at the same time it keeps the ability of accurate classification.

2.5 Margin Distribution Logistic Machine for Multi-class Classification. As most of real world data is composed by various classes, it is necessary to extend MDLM to a multi-class version. Traditionally, a multi-class classifier is constructed by combining One-vs-Other binary classifiers or One-vs-One binary classifiers. However, these two strategies share the same deficiency that these independent binary classifiers are learned individually, so the information shared across different classes is not fully utilized. Alternatively, there is a natural way of constructing a C-way classifier directly and the multi-class classifier can be solved in a single optimization. Weston and Watkins [16] constructed a multi-class SVM as:

$$\begin{aligned} &\min\left\{C\sum_{i=1}^{n}\sum_{k\neq y_i}\xi_i^k + \frac{1}{2}\sum_{k=1}^{C}\sum_{j=1}^{m}\left(w_{k,j}\right)^2\right\} \\ &\text{s.t.}\mathbf{w}_{y_i}^T\mathbf{x}_i \geq \mathbf{w}_k^T\mathbf{x}_i + 2 - \xi_i^k \\ &\xi_i^k \geq 0\,(i = 1, 2, \cdots, n; k \neq y_i) \end{aligned} \tag{2.15}$$

In [5], Zhang et al. further modified the multi-class SVM with GLL, and a Multi-class Maximum margin

Logistic Regression (MMLR) was derived:

$$\min \frac{1}{n\alpha}\sum_{i=1}^{n}\sum_{k\neq y_i}\ln\left(1+\exp\left(-\alpha\left(\left(\mathbf{w}_{y_i}^T-\mathbf{w}_k^T\right)\mathbf{x}-2\right)\right)\right) + \lambda\sum_{k=1}^{C}\sum_{j=1}^{m}(w_{k,j})^2 \tag{2.16}$$

In this paper, an equivalent multi-class model to [16, 5] is proposed. Suppose the data has C different labels, we reconstruct a label matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_C] \in \{-1, 1\}^{N\times C}$ and $\mathbf{y}_i = [-1, \cdots, -1, \underbrace{1, \cdots, 1}_{i-\text{thclass}}, -1, \cdots, -1]^T$, the objective function of multi-class MDLM is

$$\begin{aligned}\arg\min_{\mathbf{W}=[\mathbf{w}_1\cdots,\mathbf{w}_C]} &\sum_{i=1}^{C}\frac{1}{N\alpha}\mathbf{1}_N^T \\ &\ln\left[\mathbf{1}_N+\exp\left(-\alpha\left(\mathbf{y}_i\odot\mathbf{X}^T\mathbf{w}_i-\mathbf{1}_N\right)\right)\right] \\ &+\lambda_1\hat{\eta}_i-\lambda_2\bar{\eta}_i+\beta\|\mathbf{W}\|_{2,1}\end{aligned} \tag{2.17}$$

In (2.17), the multi-class MDLM tries to learn C One-vs-Other binary classifiers simultaneously, and a structural sparsity constraint L21-norm is applied here, which is for capturing the information shared across different classes. Specifically, When C equals to 1 and β equals to 0, the multi-class MDLM will degenerate to binary-class MDLM. It is worth noting that when C equals to 1 and β is not 0, the proposed multi-class MDLM will degenerate to a sparse MDLM (SMDLM) with L1-norm constraint which is demonstrated to be effective in designing a classification model [14, 15] .

3 Optimization.

As MDLM (2.14) and SMDLM are specific case of the multi-class MDLM (2.17), a proper optimization algorithm should be designed to solve (2.17) which is a non-smooth but convex problem. In this paper, we adopt the Accelerate Proximal Gradient(APG) [17, 18] algorithm to optimize (2.17).

Firstly, (2.17) is separated into a smooth part and a non-smooth part. The non-smooth part is:

$$g(\mathbf{W}) = \beta\|\mathbf{W}\|_{2,1} \tag{3.18}$$

and the smooth part is:

$$\begin{aligned}f(\mathbf{W}) = &\sum_{i=1}^{C}\frac{1}{N\alpha}\mathbf{1}_N^T \\ &\ln\left[\mathbf{1}_N+\exp\left(-\alpha\left(\mathbf{y}_i\odot\mathbf{X}^T\mathbf{w}_i-\mathbf{1}_N\right)\right)\right] \\ &+\lambda_1\hat{\eta}_i-\lambda_2\bar{\eta}_i\end{aligned} \tag{3.19}$$

Algorithm 1 APG for Multi-class Margin Distribution Logistic Machine

Input: Data matrix $\mathbf{X}\in\Re^{D\times N}$, corresponding label $\mathbf{Y}\in\Re^{N\times C}$, parameters of margin distribution λ_1, λ_2, parameter of L21-norm regularization β.
Initialization: $l_0>0$, $\sigma>0$, $\mathbf{W}_0=\mathbf{W}_1=0$, $\mu_0=1$.
For i = 1 **To** Max_Iteration
1. Compute search point $\mathbf{Q}_i$ according to (3.24);
2. $l=l_{i-1}$;
3. **While** $(f(\mathbf{W}_{i+1})+g(\mathbf{W}_{i+1}))>\Omega_l(\mathbf{W}_{i+1},\mathbf{Q}_i)$;
5. $l=\sigma l$;
6. $\mathbf{W}_{i+1}$ is updated by (3.21)
8. **End While**
7. Set $l_i=l$;

End For

Then the original function $f(\mathbf{W})+g(\mathbf{W})$ can be approximately expressed by:

$$\begin{aligned}\Omega_l(\mathbf{W},\mathbf{W}_i) = &f(\mathbf{W}_i)+g(\mathbf{W}) \\ &+\frac{l}{2}\|\mathbf{W}-\mathbf{W}_i\|_F^2+\langle\mathbf{W}-\mathbf{W}_i,\nabla f(\mathbf{W}_i)\rangle\end{aligned} \tag{3.20}$$

$\nabla f(\mathbf{W}_i)$ denotes the gradient of $f(\mathbf{W})$ at point $\mathbf{W}_i$ of the i-th iteration and l is the step size. Finally, the update step of APG algorithm is defined as:

$$\begin{aligned}\mathbf{W}_{i+1} &= \arg\min_{\mathbf{W}}\frac{1}{2}\|\mathbf{W}-\mathbf{V}\|_F^2+\frac{1}{l}g(\mathbf{W}) \\ &= \arg\min_{\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_D}\frac{1}{2}\sum_{j=1}^{D}\left(\|\mathbf{w}_j-\mathbf{v}_j\|_2^2+\frac{\beta}{l}\|\mathbf{w}_j\|_2\right)\end{aligned} \tag{3.21}$$

The analytical solutions of those sub-problems are obtained by:

$$\mathbf{w}_j^* = \begin{cases}\left(1-\frac{\beta}{l\|\mathbf{v}_j\|_2}\right)\mathbf{v}_j, \text{if}\|\mathbf{v}_j\|_2>\frac{\beta}{l}\\ 0, \text{otherwise}\end{cases} \tag{3.22}$$

where $\mathbf{w}_j$ and $\mathbf{v}_j$ denote the j-th column of $\mathbf{W}$ and $\mathbf{V}$ respectively. Note that l can be determined by line search and

$$\mathbf{V} = \mathbf{W}_i-\frac{1}{l}\nabla f(\mathbf{W}_i) \tag{3.23}$$

Instead of performing gradient descent on $\mathbf{w}$, we define the search point as

$$\mathbf{Q}_i = \mathbf{W}_i+\alpha_i(\mathbf{W}_i-\mathbf{W}_{i-1}) \tag{3.24}$$

where $\alpha_i=\frac{(1-\mu_{i-1})\mu_i}{\mu_{i-1}}$ and $\mu_i=\frac{2}{i+3}$.

The optimization algorithm of MDLM is presented in Algorithm 1 briefly. Algorithm 1 can be further modified into a large scale version by applying batch mode Stochastic Gradient Descent technique.

Table 1: Experiments on Synthetic dataset

Training Data Size	LR	h-SVM	s-SVM	MDLM
Positive: N([5 , 5] , [3 0 ; 0 1])		Negative: N([7 , 7] , [1 0 ; 0 1])		
N = 10	0.681±0.090	0.819±0.056	0.843±0.019	**0.844±0.017**
N = 100	0.651±0.069	0.829±0.074	**0.860±0.004**	0.857±0.014
N = 1000	0.765±0.011	Out of time	**0.864±0.003**	0.862±0.009
Positive: N([5 , 5] , [3 0 ; 0 1])		Negative: N([8 , 8] , [1 0 ; 0 1])		
N = 10	0.767±0.077	0.916±0.035	0.929±0.020	**0.930±0.020**
N = 100	0.796±0.029	0.934±0.041	0.947±0.014	**0.947±0.007**
N = 1000	0.867±0.007	Out of time	**0.949±0.002**	0.947±0.007
Positive: N([5 , 5] , [3 0 ; 0 1])		Negative: N([9 , 9] , [1 0 ; 0 1])		
N = 10	0.800±0.077	0.963±0.029	0.967±0.014	**0.969±0.023**
N = 100	0.828±0.038	0.981±0.006	0.982±0.005	**0.984±0.005**
N = 1000	0.900±0.005	Out of time	**0.985±0.001**	0.983±0.008

4 Experiments.

In this section, experiments are performed on both synthetic dataset and real world dataset for the comparison between MDLM and other classifiers in both binary classification and multi-class classification tasks.

4.1 Synthetic Dataset. We build a synthetic dataset to give a glance at the property of our proposed MDLM model. We sample 10/100/1000 data points from two 2-D Gaussian distributions for training a classifier and sample another 10000 data points from these two distributions for validating the performance of the learned classifier, the experiment is conducted for 100 times. In this experiment, the comparison is conducted between LR, hard-margin SVM (h-SVM)[1], soft-margin SVM (s-SVM) and MDLM. The experimental results show LR is prone to overfitting, and SVM and MDLM perform better than traditional LR. For the same reason, they are also better than hard-margin SVM. When training data is limited, MDLM obtains the best performance because MDLM can utilize more statistical information from data. When the data size goes larger, soft-margin SVM and MDLM have a similar performance on synthetic dataset, but hard-margin SVM is hard to converge to a proper solution.

4.2 Binary Classification. We collect 9 binary classification datasets[2], and the detailed information of these datasets is presented in Table 2. In this experiment, we compare our proposed MDLM with baseline classifier SVM with RBF kernel and some state-of-art classifiers, i.e., L2-norm regularized Logistic Regression (L2LR), Maximum margin Logistic Regression (MLR) [5] and Large margin Distribution Machine (LDM) [13]. All experiments are conducted with 30 times 2-folds cross-validation to split the training data and testing data. The parameters for margin distribution in MDLM and LDM are tuned from 2^{-3} to 2^3. The experimental result is shown in Table 3. From Table 3, in most cases, MDLM has a significant better performance to other classifiers in accuracy and AUC (area under ROC curve) score. This demonstrates the superiority of MDLM to other classifiers in binary classification tasks.

Table 4: Performance on Noisy Data

Dataset	**0%**	**10%**	**20%**	**30%**
diabetes	2.55%	3.65%	4.04%	4.70%
wpbc	0.78%	3.93%	4.10%	5.15%
ionosphere	1.96%	3.23%	3.42%	4.55%
hypothyroid	-1.04%	3.09%	5.95%	10.0%
australian	1.52%	1.53%	2.30%	2.99%
heart	0.24%	0.87%	1.04%	2.34%
promoter	1.02%	2.14%	3.45%	4.07%
vote	2.00%	1.69%	2.83%	3.31%
house	1.68%	2.13%	2.92%	3.88%

To further validate the robustness of the proposed MDLM method, we examine its performance under different noise conditions. Specifically, we compare MDLM with MLR, which also uses GLL as the loss function but tries to maximize the minimum margin. For each dataset, we increase the noise[3] rate from 0% to 30% with a step size of 10%, and report the improvement of MDLM over MLR for each case. From the results in Table 4, we can observe a clear trend that the advantage of MDLM over MLR becomes more

[1] In this paper, SVM is realized by LIBSVM [25].
[2] http://archive.ics.uci.edu/ml/
[3] Gaussian white noise with a certain Signal-Noise ratio

Table 2: Notations of Experimental Datasets

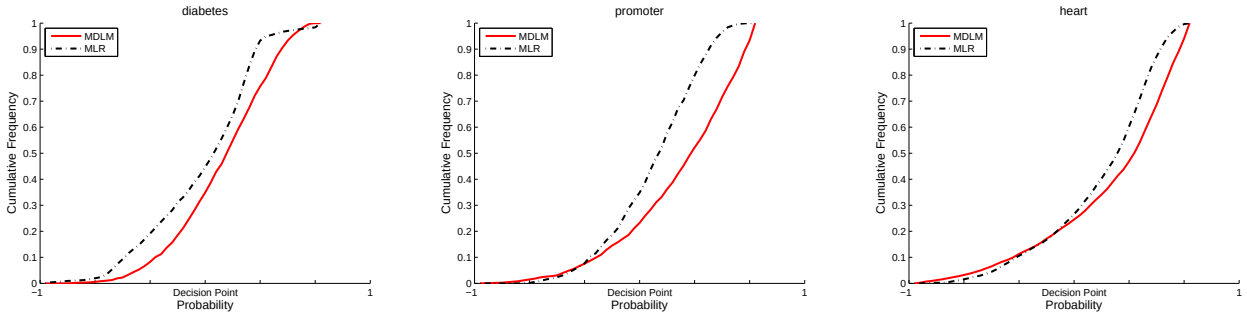
Binary-class Dataset			***Multi-class Dataset***			
Dataset	**Dimension**	**Number**	**Dataset**	**Dimension**	**Number**	**Class**
diabetes	8	269/500	*soybean*	35	47	4
wpbc	33	151/47	*lymph*	38	148	4
ionosphere	34	225/126	*dermatology*	34	361	6
hypothyroid	60	232/136	*ORL*	1024	400	40
australian	14	307/383	*USPS_resam*	256	9298	10
heart	9	120/150				
promoter	57	53/53				
vote	16	108/124				
house	16	124/108				

Table 3: Experimental Results of Binary Classification

Dataset		**SVM**	**L2LR**	**MLR**	**LDM**	**MDLM**
diabetes	*ACC*	0.680±0.023	0.672±0.031	0.657±0.019	0.651±0.058	**0.683±0.022**
	AUC	0.632±0.028	0.649±0.022	0.655±0.021	0.639±0.057	**0.661±0.019**
wpbc	*ACC*	0.751±0.032	0.582±0.055	0.761±0.033	0.742±0.035	**0.767±0.047**
	AUC	0.601±0.033	0.679±0.068	0.501±0.047	0.646±0.074	0.687±0.068
ionosphere	*ACC*	0.845±0.024	0.861±0.029	0.868±0.024	**0.897±0.031**	0.885±0.016
	AUC	0.881±0.015	0.935±0.026	0.940±0.018	**0.961±0.018**	0.942±0.014
hypothyroid	*ACC*	0.646±0.016	0.770±0.028	**0.772±0.026**	0.667±0.033	0.764±0.026
	AUC	0.702±0.057	**0.854±0.031**	0.845±0.019	0.669±0.053	0.842±0.021
australian	*ACC*	0.845±0.023	0.858±0.011	0.853±0.014	0.797±0.023	**0.866±0.013**
	AUC	0.903±0.012	0.916±0.011	0.921±0.012	0.826±0.017	**0.929±0.011**
heart	*ACC*	0.781±0.022	0.809±0.030	0.822±0.031	0.787±0.023	**0.824±0.028**
	AUC	0.859±0.017	0.885±0.019	0.901±0.021	0.861±0.022	**0.903±0.013**
promoter	*ACC*	0.617±0.041	0.751±0.036	0.788±0.052	0.780±0.036	**0.796±0.025**
	AUC	0.863±0.032	0.861±0.021	0.868±0.036	0.850±0.042	**0.872±0.037**
vote	*ACC*	0.897±0.025	0.953±0.014	0.951±0.016	0.947±0.012	**0.970±0.012**
	AUC	0.976±0.005	0.983±0.005	0.985±0.005	0.978±0.006	**0.987±0.010**
house	*ACC*	0.906±0.032	0.945±0.024	0.953±0.016	0.950±0.021	**0.969±0.011**
	AUC	0.979±0.009	0.988±0.007	0.987±0.004	0.980±0.012	**0.990±0.006**

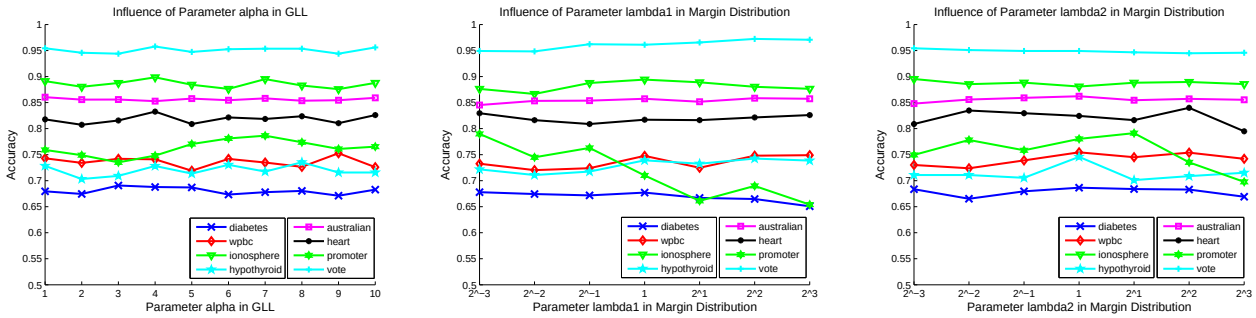
Table 5: Experimental Results of Multi-class Classification

Dataset		**SVM**	**L2LR**	**MMLR**	**MDLM**
soybean	*Macro F1*	0.993±0.015	0.997±0.007	0.997±0.010	**1±0.000**
	Micro F1	0.989±0.024	0.995±0.010	0.987±0.011	**1±0.000**
lymph	*Macro F1*	0.941±0.010	0.922±0.008	0.923±0.011	**0.948±0.009**
	Micro F1	0.883±0.021	0.903±0.010	0.902±0.013	**0.908±0.010**
dermatology	*Macro F1*	0.966±0.005	0.976±0.007	0.981±0.009	**0.986±0.009**
	Micro F1	0.967±0.008	0.978±0.008	0.983±0.009	**0.987±0.007**
ORL	*Macro F1*	0.967±0.014	0.927±0.005	0.940±0.008	**0.975±0.004**
	Micro F1	0.962±0.016	0.933±0.006	0.939±0.007	**0.972±0.006**
USPS_resam	*Macro F1*	**0.963±0.009**	0.935±0.001	0.939±0.001	0.955±0.002
	Micro F1	**0.962±0.012**	0.934±0.002	0.939±0.001	0.957±0.002



(a) diabetes (b) promoter (c) heart

Figure 3: Effects of Margin Distribution to Probability Output of Logistic Regression



(a) Influence of α (b) Influence of λ_1 (c) Influence of λ_2

Figure 4: Influence of Parameters in MDLM

significant when the noise rate increases.

4.3 Multi-class Classification. We also examine the performance of MDLM in multi-class classification and the detailed information of these multi-class datasets is presented in Table 2. We adopt 5 multi-class datasets and 4 different classifiers including SVM with RBF kernel, L2-norm regularized Logistic Regression (L2LR), Multi-class Maximum margin Logistic Regression (MMLR) [5] to evaluate the effectiveness proposed MDLM. From Table 5, for most datasets, with the help of structural information and margin distribution, the multi-class MDLM can not only utilize more information shared across different classes, but also more distribution information of training data, thus MDLM achieves better performance in Macro-F1 and Micro-F1 scores.

4.4 Effects of Margin Distribution. We further examine the effects of margin distribution to MDLM. This experiment is conducted with MDLM and MLR on three representative datasets. We plot the cumulative frequency curve of probability output from the learned logistic regression in Fig.3. In Fig.3, the decision point decides when one sample is correctly predicted, its corresponding probability is positive, if not, versus. As we can see from Fig.3, the cumulative frequency curve of MDLM lies in the right of MLR in all three datasets, which means the probability output of MDLM is more convincing than MLR.

4.5 Parameter Influence. There are three parameters in MDLM including α in GLL, λ_1 and λ_2 in margin distribution, which can be determined by cross-validation. We examine the influence of these three parameters on 8 representative datasets by changing one parameter with other two parameters fixed. The experimental result is shown in Fig.4. Specifically, Fig.4(a) shows the influence of α in GLL when λ_1, λ_2 are fixed with the suggested value by cross-validation. It can be seen from Fig.4(a), when α changes from 1 to 10, for most datasets, the performance of MDLM changes slightly. Fig.4(b) and Fig.4(c) show the influence of λ_1 and λ_2, respectively. As we can see from Fig.4(b)(c), when the parameter is adjusted from 2^{-3} to 2^3, the performance of MDLM is stable at most time. It means MDLM is not much sensitive to parameters set in a reasonable interval. These features make MDLM a prac-

tical and powerful classifier for classification tasks.

5 Conclusion.

Linear classifier is one of the most commonly used technology in machine learning for its simplicity and effectiveness. However, improving the robustness of the linear classifier is still a challenge. In this paper, we incorporate the flexible generalized logistic loss with margin distribution to propose the Margin Distribution Logistic Machine (MDLM). By considering the complementary statistical information of data distribution, the MDLM is robust to outliers and noises. From experimental results on synthetic datasets and real world datasets, the binary classification performance of MDLM is demonstrated to be superior to traditional classifier and other state-of-art models. Furthermore, the proposed MDLM is extended to multi-class classification by introducing a structural sparsity constraint, which help capture more information shared across different classes, the effectiveness of proposed multi-class MDLM is demonstrated by experiments. With the robustness, smoothness and convexity of the proposed MDLM, it is worth being extended to many other learning tasks in future works.

References

[1] G.-X. YUAN, C.-H. HO and C.-J. LIN, Recent advances of large-scale linear classification, Proceedings of the IEEE, 100 (2012), pp. 2584-2603.

[2] V. VAPNIK, The nature of statistical learning theory, Springer Science & Business Media, 2013.

[3] T. J. HASTIE, R. J. TIBSHIRANI and J. H. FRIEDMAN, The elements of statistical learning: data mining, inference, and prediction, Springer, 2011.

[4] T. ZHANG and F. J. OLES, Text categorization based on regularized linear classification methods, Information retrieval, 4 (2001), pp. 5-31.

[5] J. ZHANG, R. JIN, Y. YANG and A. G. HAUPTMANN, Modified logistic regression: An approximation to SVM and its applications in large-scale text categorization, ICML, 2003, pp. 888-895.

[6] C. PARK, J.-Y. KOO, P. T. KIM and J. W. LEE, Stepwise feature selection using generalized logistic loss, Computational Statistics & Data Analysis, 52 (2008), pp. 3709-3718.

[7] S. PATRA, K. SHANKER and D. KUNDU, Sparse maximum margin logistic regression for credit scoring, 2008 Eighth IEEE International Conference on Data Mining, IEEE, 2008, pp. 977-982.

[8] A. GARG, S. HAR-PELED and D. ROTH, On generalization bounds, projection profile, and margin distribution, ICML, 2002, pp. 171-178.

[9] A. GARG and D. ROTH, Margin distribution and learning algorithms, Proceedings of the Fifteenth International Conference on Machine Learning (ICML), 2003, pp. 210-217.

[10] H. XUE, S. CHEN and Q. YANG, Structural support vector machine, International Symposium on Neural Networks, Springer, 2008, pp. 501-511.

[11] K. PELCKMANS, J. SUYKENS and B. D. MOOR, A risk minimization principle for a class of parzen estimators, Advances in Neural Information Processing Systems, 2008, pp. 1137-1144.

[12] W. GAO and Z.-H. ZHOU, On the doubt about margin explanation of boosting, Artificial Intelligence, 203 (2013), pp. 1-18.

[13] T. ZHANG and Z.-H. ZHOU, Large margin distribution machine, Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2014, pp. 313-322.

[14] P. S. BRADLEY and O. L. MANGASARIAN, Feature selection via concave minimization and support vector machines, ICML, 1998, pp. 82-90.

[15] J. ZHU, S. ROSSET, T. HASTIE and R. TIBSHIRANI, 1-norm support vector machines, Advances in neural information processing systems, 16 (2004), pp. 49-56.

[16] J. WESTON and C. WATKINS, Support vector machines for multi-class pattern recognition, ESANN, 1999, pp. 219-224.

[17] X. CHEN, W. PAN, J. T. KWOK and J. G. CARBONELL, Accelerated gradient method for multi-task sparse learning problem, 2009 Ninth IEEE International Conference on Data Mining, IEEE, 2009, pp. 746-751.

[18] J. LIU and J. YE, Efficient l1/lq norm regularization, arXiv preprint arXiv:1009.4766 (2010).

[19] K. KOH, S.-J. KIM and S. BOYD, An interior-point method for large-scale l1-regularized logistic regression, Journal of Machine learning research, 8 (2007), pp. 1519-1555.

[20] G.-X. YUAN, C.-H. HO and C.-J. LIN, An improved glmnet for l1-regularized logistic regression, Journal of Machine Learning Research, 13 (2012), pp. 1999-2030.

[21] A. Y. NG, Feature selection, L1 vs. L2 regularization, and rotational invariance, Proceedings of the twenty-first international conference on Machine learning, ACM, 2004, pp. 78.

[22] S. RYALI, K. SUPEKAR, D. A. ABRAMS and V. MENON, Sparse logistic regression for whole-brain classification of fMRI data, NeuroImage, 51 (2010), pp. 752-764.

[23] M. Y. PARK and T. HASTIE, Penalized logistic regression for detecting gene interactions, Biostatistics, 9 (2008), pp. 30-50.

[24] J. LIU, J. CHEN and J. YE, Large-scale sparse logistic regression, Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2009, pp. 547-556.

[25] C.-C. CHANG and C.-J. LIN, LIBSVM: a library for support vector machines, ACM Transactions on Intelligent Systems and Technology (TIST), 2 (2011), pp. 27. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm