Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence

Robust Distance Metric Learning in the Presence of Label Noise

Dong Wang, Xiaoyang Tan
Department of Computer Science and Technology
Nanjing University of Aeronautics and Astronautics
#29 Yudao Street, Nanjing 210016, P.R.China
{dongwang, x.tan}@nuaa.edu.cn

Abstract

Many distance learning algorithms have been developed in recent years. However, few of them consider the problem when the class labels of training data are noisy, and this may lead to serious performance deterioration. In this paper, we present a robust distance learning method in the presence of label noise, by extending a previous non-parametric discriminative distance learning algorithm, i.e., Neighbourhood Components Analysis (NCA). Particularly, we analyze the effect of label noise on the derivative of likelihood with respect to the transformation matrix, and propose to model the conditional probability of the true label of each point so as to reduce that effect. The model is then optimized within the EM framework, with additional regularization used to avoid overfitting. Our experiments on several UCI datasets and a real dataset with unknown noise patterns show that the proposed RNCA is more tolerant to class label noise compared to the original NCA method.

Introduction

The goal of distance learning is to learn a distance function tailored to some task at hand, and has been found to be useful in the KNN or other classification algorithms relying on distances or similarities (Kulis 2012). Many algorithms (Blitzer, Weinberger, and Saul 2005), (Davis et al. 2007), (Park et al. 2011) have been developed to achieve this goal recently. One of the most popular ones is to treat the problem as learning a linear transformation (e.g., a Mahalanobis matrix) using some form of supervision.

One typical example is the Neighbourhood Components Analysis (NCA) algorithm among others, which is a discriminative non-parametric method with the goal to learn a Mahalanobis distance measure to be used in the KNN algorithm. The key idea of NCA is to minimize the probability of error under stochastic neighborhood assignments using gradient descent. Computationally this is equal to drive the linear transform of interest in a direction that reduces the probabilistic intra-class scatter matrix most.

However, one major problem of this type of algorithm is that, to calculate the intra-class scatter matrix, one has to know the perfect class labels, which are almost impossible in many cases. In fact, when the needed class label information comes from the web (e.g., through crowdsourcing (Howe 2006) or harvesting them with weak labels by web searching (Bergamo and Torresani 2010)) or when the number of data to be labeled is huge, they tend to be noisy and inaccurate and using them blindly is dangerous. In particular, many distance learning algorithms (including NCA) are based on the idea of minimizing the pairwise distances between examples within the same category. But if the observed labels are noisy (i.e., labeled incorrectly), they can be misled to pull examples from different classes together.

In this paper, we present a robust distance learning method in the presence of label noise based on the NCA method (hence called RNCA - robust NCA). The reasons we choose NCA as our basic model are mainly due to the fact that NCA is a popular and typical linear transformation learning-based method and that it is developed under a well-formulated probabilistic framework. Particularly, we analyze the effect of label noise on the derivative of likelihood with respect to the transformation matrix, and propose to model the conditional probability of the true label of each point for a more robust estimation of intra-class scatter matrix. The model is then optimized within the EM framework. In addition, considering that the model tends to be complex under the situation of label noise, we regularize its objective to avoid overfitting. Our experiments on several UCI datasets and a real dataset with unknown noise patterns show that the proposed RNCA is more tolerant to class label noise compared to the original NCA method.

In what follows, we review the related work in section 2 and the NCA is introduced in Section 3. Section 4 details the method and experiments are given in Section 5. The paper concludes in section 6.

Related Work

The problem of label noise is previously studied under the umbrella of agnostic learning (Kearns, Schapire, and Sellie 1994), in which the relationship between the label and the data is largely relaxed. It is closely related to overfitting - any strategy (e.g., regularization) that prevents a learning algorithm from overfitting the data has the capability to reduce the influence of label noise to some extend. In this sense, many supervised methods in practice are robust to the label noise. Despite this, another important factor influencing generalization is the loss function - some are known to be

extremely sensitive to the label noise, such as the exponential loss adopted by Adaboost and even for the hingle loss adopted by SVM, an incorrectly labelled data point could lead to arbitrary large loss, despite of the SVM's noise relaxation mechanism[1]. Metric learning is a kind of supervised learning and hence suffers from the same problem as well.

Recently many works have been devoted to deal with the problem of label noise recently (Frénay and Verleysen 2013) and they can be roughly divided into three categories. The first type - perhaps the most intuitive type among them - is to pre-precess the data such that the data points whose labels are likely to be noise will be removed before feeding to classifier training (Van Hulse, Khoshgoftaar, and Napolitano 2010) (Fefilatyev et al. 2012).

The second type of methods tries to estimate the probability of their labels being noised and warns the classifier for this. The key issue here, therefore, is how to identify those suspicious points confidently. For this, in (Lawrence and Schölkopf 2001) a probabilistic model of a kernel Fisher Discriminant is presented in which an EM algorithm is proposed to update the probability of the data point being incorrectly labeled. This EM-type algorithm has inspired many later methods (including this work) in which the true but unknown label of each data point is treated as latent variable and its posterior given the current label is estimated in a probabilistic framework (Pal, Mann, and Minerich 2007) (Bootkrajang and Kabán 2012). Alternatively, a multiple instance learning-based method is proposed in (Leung, Song, and Zhang 2011) to cope with label noise, but it essentially has to estimate the most correctly labeled positive samples in a bag. Some heuristic strategy can also be adopted. For example, (Cantador and Dorronsoro 2005) takes boosting to detect the incorrect labels based on the observation that those data are likely to have a big weight.

The third type of methods uses various robust optimization methods or robust estimation methods to bound the influence of each data point, such that the model behaves stable despite of the existence of few outliers with incorrect labels. (Wu and Liu 2007) and (Biggio, Nelson, and Laskov 2011) respectively propose to use truncated hinge loss or kernel matrix correction to improve the robustness of SVM against label noise. Recently, (Natarajan et al. 2013) proposes a method which modifies any given surrogate loss function so that it becomes a label noise robust one, and (Scott, Blanchard, and Handy 2013) studies the problem of under what conditions that consistent classification with label noise is possible.

Despite the above development, surprisingly few works investigate this problem in the context of distance metric learning. In (Yang, Jin, and Jain 2010) a kernel logistic regression method is proposed to deal with the problem when the side information used for distance learning is noisy. Our method is different to (Yang, Jin, and Jain 2010) in that we do not rely on pairwise side information for distance learning, and we adopted a non-parametric way instead of a parametric model to deal with the label noise problem. Detailed account will be given later after an introduction to the NCA.

[1]This mechanism of relaxation in SVM helps to improve its robustness against perturbation of data points, which is also an important problem and is not addressed in this paper.

A Brief Introduction to NCA

Neighbourhood components analysis (NCA)(Goldberger et al. 2004), (Blitzer, Weinberger, and Saul 2005) is a method of distance metric learning that maximizes the performance of KNN classification. For a training set containing n data from k classes, that is: $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n), x_i \in R^d, y_i \in \{1, 2, ..., K\}\}$, the goal of NCA is to learn a symmetric PSD (positive semi-definite) distance metric. Assuming the target learning metric is M, then it can be write as $M = A^T A$ such that $d(x_i, x_j) = (x_i - x_j)^T M (x_i - x_j) = (Ax_i - Ax_j)^T (Ax_i - Ax_j)$. NCA projects the data into a new subspace where each point is more likely to select those points from the same class as its neighbors.

The NCA algorithm begins by constructing a complete graph with each data point as its node, characterizing the manifold the data lie on. Let the weight of each edge between any two nodes denoted by p_{ij}, which can be calculated as the probability that data point x_i selects x_j as its neighbour:

$$p_{ij} = \frac{exp(-\|Ax_i - Ax_j\|^2)}{\sum_{k \neq i} exp(-\|Ax_i - Ax_k\|^2)}, \;\; p_{ii} = 0 \qquad (1)$$

It can be checked that $p_{ij} \geq 0$ and $\sum_j p_{ij} = 1$, and hence p_{ij} is a valid probability measure. Then the conditional probability that a data point x_i belongs to class k is,

$$p(y_i = k|x_i) = \sum_{j \in C_i^k} p_{ij} = \sum_j p_{ij} 1(y_i = k) 1(y_j = k) \qquad (2)$$

where $1(\cdot)$ is an indicator function with output 1 if the input condition is satisfied and 0 otherwise, and $C_i^k = \{j | y_j = y_i = k\}$ is the set in which all data belong to class k. This can also be understood as the probability for point x_i select its same class points as its neighbours. For convenience, let us denote $p_i^k \equiv p(y_i = k|x_i)$.

The object of NCA is then to learn a linear transformation A such that the log likelihood that all data select the points within its same category as neighbours is maximized, i.e.,

$$f(A) = \sum_i \sum_k \log(p_i^k) \qquad (3)$$

Differentiating f with respect the transformation matrix A yields the gradient which can be used for learning (denote $x_{ij} = x_i - x_j$):

$$\begin{aligned} \frac{\partial f}{\partial A} = 2A \sum_i (& \sum_j p_{ij} x_{ij} x_{ij}^T \\ & - \sum_j p_{ij} x_{ij} x_{ij}^T \sum_k \frac{1(y_i = k) 1(y_j = k)}{p_i^k}) \end{aligned} \qquad (4)$$

To gain further understanding of the NCA algorithm, let us denote the two terms in the gradient (E.q.(4)) as C_E and C_I, respectively,

$$C_E = \sum_i \sum_j p_{ij} x_{ij} x_{ij}^T \tag{5}$$

$$C_I = \sum_i \sum_j p_{ij} x_{ij} x_{ij}^T \sum_k \frac{1(y_i = k) 1(y_j = k)}{p_i^k} \tag{6}$$

We see that,

$$\frac{\partial f}{\partial A} = 2A(C_E - C_I) \tag{7}$$

Intuitively, the C_E term denotes the total scatter matrix of the data points lying on the manifold induced by A and C_I is the corresponding intra-class scatter matrix. E.q.(7) reveals that, up to a constant matrix, in each step the NCA algorithm tries to seek a better linear transformation such that after projection the total covariance becomes 'larger' while the intra-class covariance becomes 'smaller'. In other words, the NCA aims to learn a distance transformation with the following two goals, i.e., to keep the total energy of whole data set while clustering data points within the same class more tighter.

The Proposed Method

The analysis above shows that the key to the success of NCA relies on the accurate estimation of intra-class scatter matrix C_I on the manifold. However, when the class labels are inaccurate or noisy, the estimation of C_I tends to be inaccurate (the C_E will be not influenced by this). To address issue, we introduce a latent variable y, to represent the true label, and the observed (noisy) label will be denoted as $\hat{y}$ from then on. We model the true conditional probability as the following:

$$p(y_i = k|x_i, \theta) = \sum_j p_{ij} \cdot p(y_j = k|\hat{y}_j) \tag{8}$$

It can be easily checked that this is a valid probability measure. E.q. (8) says that the probability of instance x_i belonging to class k depends on its neighbors' probability being class k given their observed labels, no matter what the observed label of x_i is. Note that this is still a non-parametric model with the parameters $\theta = \{A, \gamma_{jk}\}$ (denote $\gamma_{jk} = p(y_j = k|\hat{y}_j)$).

Optimizing the Parameters

The log-likelihood objective function of our model takes the form of:

$$L(\theta) = \sum_{i=1}^{N} \log\ p(\hat{y}_i|x_i, \theta) \tag{9}$$

But considering that the model tends to be complex under the situation of label noise, we regularize this log-likelihood function to prevent overfitting, i.e.,

$$L(\theta) = \sum_{i=1}^{N} \log\ p(\hat{y}_i|x_i, \theta) - \lambda r(A) \tag{10}$$

where the $r(A)$ is the regularizer of some form. Several options can be considered here, such as the trace norm and the *LogDet divergence* (Davis et al. 2007). In this work, we take the simple Frobenius norm, i.e., $r(A) = ||A||_F^2$.

To determine each $p(\hat{y}_i|x_i, \theta)$ we must marginalise the associated latent variable y, which can done through an EM algorithm (Dempster, Laird, and Rubin 1977). By introducing an unknown posterior Q on the true class label y_i, we have the following lower bound of the likelihood function, which will be used as our objective function for parameter estimation,

$$\begin{aligned} L(\theta) &\geq \sum_{i=1}^{N} \sum_k Q(y_i = k|x, \hat{y}_i) \log\ \frac{p(\hat{y}_i, y_i|x_i, \theta)}{Q(y_i = k|x, \hat{y}_i)} - \lambda r(A) \\ &\equiv L_c(\theta) \end{aligned} \tag{11}$$

where λ is a parameter for model selection and has to be tuned using cross validation in practice. This bound becomes an equality if $Q(y_i|x_i, \hat{y}_i) = p(y_i|x_i, \hat{y}_i, \theta)$. By assuming a uniform distribution on $p(y = k)$, and using the notations defined above, Q can be estimated as follows,

$$Q(y_i = k|x_i, \hat{y}_i = j, \theta) = \frac{p_i^k \gamma_{ik}}{\sum_k p_i^k \gamma_{ik}} \tag{12}$$

where p_i^k is the discriminative model defined in E.q.(8).

The EM algorithm can be understood as a self-taught algorithm: first we use the current model (with its parameter vector denoted as θ^t) to estimate the true label of each data point (E-step), then we optimize the model again based on the estimated true labels (M-step). For the M-step, we first rearrange our objective function according to the parameters of interest (below we temporally drop the regularizer term for clearness), discarding those items irrelevant, as follows,

$$L_c(\theta) = \sum_{i=1}^{N} \sum_{k=1}^{K} Q(y_i = k|x_i, \hat{y}_i) \log\ p(\hat{y}_i, y_i = k|x_i, A) \tag{13}$$

and,

$$p(\hat{y}_i, y_i = k|x_i, A) = p(y_i = k|x_i, A) p(\hat{y}_i|y_i = k) \tag{14}$$

Then the objective function involving A would be,

$$\begin{aligned} L_c(A) &= \sum_{i=1}^{N} \sum_{k=1}^{K} Q(y_i = k|x_i, \hat{y}_i) \log\ p(y_i = k|x_i, A) \\ &= \sum_{i=1}^{N} \sum_{k=1}^{K} Q(y_i = k|x_i, \hat{y}_i) \log\ \sum_j p_{ij} \cdot p(y_j = k|\hat{y}_j) \end{aligned} \tag{15}$$

Denoting α_{jk} for $p(y_j = k|x_j, \hat{y}_j, A)$, and recalling that $p_i^k = p(y_i = k|x_i)$, and $\gamma_{jk} = p(y_j = k|\hat{y}_j)$, we differentiate E.q.(15) by A as follows,

$$\begin{aligned} \frac{\partial L_c}{\partial A} &= \sum_{i=1}^{N} \sum_k \frac{\alpha_{ik}}{p_i^k} \frac{\partial p_i^k}{\partial A} \\ &= 2A \sum_{i=1}^{N} (\sum_{l=1}^{N} p_{il} x_{il} x_{il}^T - \sum_{j=1}^{N} p_{ij} x_{ij} x_{ij}^T \sum_k \frac{\alpha_{ik} \cdot \gamma_{jk}}{p_i^k}) \end{aligned} \tag{16}$$

Note that the gradient of regularizer term $-\lambda A$ should be added on to E.q.(16) before feeding this into a conjugate gradients optimizer.

Comparing E.q.(16) with E.q.(4), we see that the former is a natural relaxation to that of NCA. Actually, the coefficient of the intra-class scatter term can be shown to be,

$$\sum_k \frac{\alpha_{ik} \cdot \gamma_{jk}}{p_i^k} \propto \sum_k p(y_i = k|\hat{y}_i) \cdot p(y_j = k|\hat{y}_j) \quad (17)$$

This can be understood as the strength of the belief that point i and point j have the same true labels[2].

The objective involving parameter γ_{ik} would be,

$$\begin{aligned} L_c(\gamma_{ik}) = &\sum_{i=1}^{N}\sum_{k=1}^{K} Q(y_i = k|x_i, \hat{y_i}) \log\ \gamma_{ik} \\ &+ \beta(\sum_{k=1}^{K} \gamma_{ik} - 1) \end{aligned} \quad (18)$$

By taking the partial derivative and set it to zero, we have,

$$\gamma_{ik} = \frac{1}{v_i} \sum_{i=1}^{N} Q(y_i = k|x_i, \hat{y_i}) \cdot 1\{\hat{y_i} = k\} \quad (19)$$

where $v_i = \sum_{i=1}^{N} 1\{\hat{y_i} = k\}$. In words, γ_{ik} is the empirical expectation of $Q(y_i = k|x_i, \hat{y_i})$ for those points i whose observed labels are k.

Implementation

The prediction on the true label y in the E-step (see E.q.(12)) is inherently uncertain, even when the noise level is relatively low - while in this case most of the observed labels $\hat{y}$ are reliable. This inspires us to take a conservative strategy for low-level label noise, by first making an empirical estimation of the probability that the label of x_i is k based on the observations:

$$p_{ik}^o = \begin{cases} 1 & \text{if } k = \hat{y_i} \\ 0 & \text{otherwise} \end{cases} \quad (20)$$

then updating the estimation of α_{ik} as follows,

$$\alpha_{ik}^{new} = \eta p_{ik}^o + (1 - \eta)\alpha_{ik} \quad (21)$$

where η is a mixing parameter with value in $[0, 1]$. When $\eta = 1$, our model reduces to a soft version of NCA. How to set this value properly will be discussed in the experimental section, although in general we can use the cross validation technique for this.

We summarize the proposed method in Algorithm. 1.

Experiments and Analysis

The Behavior of the Proposed Method

First, three experiments are conducted on the toy data to investigate the behavior of the proposed method.

[2]In implementation, we have found that one can first find the k^* so that $k^* = arg\max_k Q(y_i = k|x_i, \hat{y_i})$, and then estimate the coefficient as $\frac{\alpha_{ik^*} \cdot \gamma_{jk^*}}{p_i^{k^*}}$. This robust estimator usually leads to better performance when the noise level is relatively high.

Algorithm 1 Robust Neighbourhood Components Analysis.

Input:

Training set: $\{(x_i, \hat{y}_i)|\ i = 1, 2, ..., N\}$;
Test set: $\{(x_i)|\ i = 1, 2, ..., N\}$;
Parameters: the regularization parameter λ, mixing parameter η,the maximal EM iteration steps T;

Output:

The prediction y of the test data x

———————- Training Stage

1: Initialisation: set α_{ik} and γ_{ik} according to the number of observations for $\hat{y}_i = k$, otherwise use uniform distribution for $\hat{y}_i \neq k$.
2: Run NCA to initialize the linear transform A.
3: while $t \leq T$
4: M-step: optimize the model parameters A and γ according to E.q.(16) and E.q.(19), respectively.
5: E-step: re-estimate p_i^k according to E.q.(8) and then α_{ik} using E.q.(21).

———————- Test Stage

6: Use KNN to make the prediction for the test data using the learnt distance measure (parametered by A);

Visualizing the influence of label noise on distance learning We first constructed a small toy dataset by sampling 400 points from two 2D unit normal distributions centered at (0,0) and (4,0) respectively, with 200 points from each distribution. Then we added three types of label noise artificially on this dataset: 1) symmetric random noise, i.e., a proportion of points are randomly selected and their corresponding class labels are flipped; 2) asymmetric label noise, i.e., flipping the labels of randomly chosen points only in one class of data; and 3) random label noise occurred in some particular regions (e.g., the boundary between two classes). The noise level added is about 15% in each case.

Fig. 1 illustrates the respective projection directions learnt by NCA and our method along with the one using true labels. It can be seen that in all the three cases, the proposed distance learning method consistently shows better tolerance against label noise compared to the NCA method.

The effect of mixing parameter η We study this on the Balance dataset of UCI database. Fig. 2 shows the system's accuracy as a function of η value. It can be seen that adding the empirical term (E.q.(20)) is beneficial to the performance when the noise level is less than 20%. In practice, the mixture weight (η) should be set according to the current label noise level - a relatively large value ($0.6 \sim 1.0$) is advised if most labels are known to be reliable, otherwise we should set η to be a very small value (0.001 or simply 0).

Visualizing the learning process: To visualize how our estimation of true label ($p(y|\hat{y}, x, A)$) evolves during the training process, we sampled 100 points from two unit 3D Gaussian distributions and add the random label noise at a noise level of 20% and run our RNCA algorithm. Figure.3 gives the results. It clearly shows that, as expected, with the iteration of the EM algorithm, the 'weights' of the samples with true labels increase gradually while the weights of those

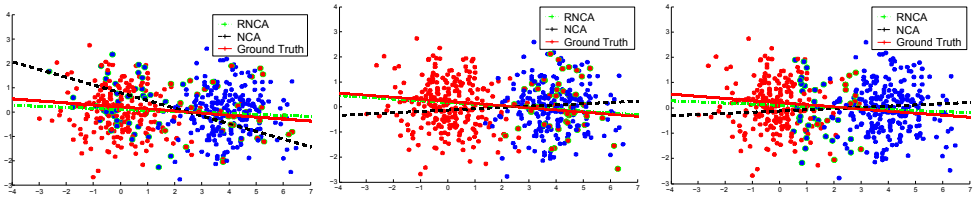

Figure 1: Comparison of the projection directions learnt by our method and the NCA on the toy data, where the data point with label noise is highlighted with a green circle. The three types of label noise are (from left to right): (a) symmetric random label noise; (b) asymmetric random label noise; (c) label noise occurs on the boundary between two classes. (noise level: 15%).

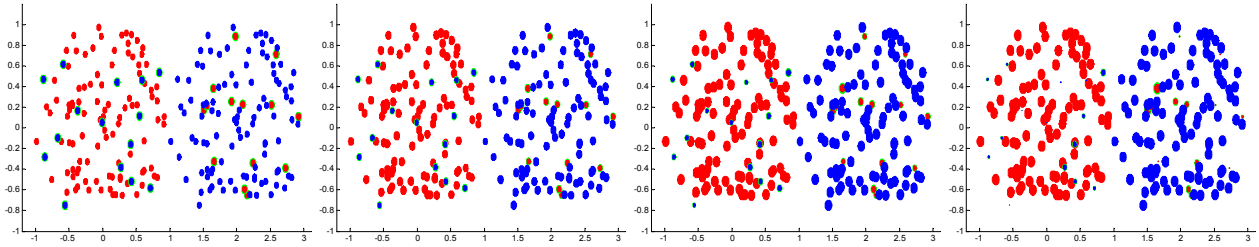

Figure 3: Illustration of the evolving of the weight of each point after 4 iterations of the EM algorithm. The size of point is proportional to its weight, and the data point with label noise is highlighted with a green circle.

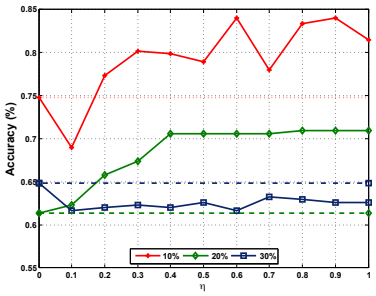

Figure 2: The effect of mixture coefficient η on the performance with different noise level ($10\% \sim 30\%$). The dash line is the corresponding baseline performance (i.e., $\eta = 0$) at each noise level.

with noisy labels shrink. By "weight" we mean the confidence of the model's estimation concerning the true label of each point (see E.q.(12)).

Robustness Against Simulated Label Noise

To compare our method with other distance learning methods in the presence of label noise, we use 6 datasets from the UCI database, with 3 multi-class datasets (i.e., Iris, balance and wine) and 3 datasets with binary labels (i.e., Heart, vote and ionosphere). Two types of label noise with different levels (ranging from 0% to 30%) are simulated: asymmetric noise is added to the multi-class datasets and symmetric noise to the binary-label datasets. Besides NCA, we compare our method with two classic distance learning methods as well, i.e., ITML(Information-Theoretic Metric Learning, (Davis et al. 2007)) and LMNN (Large Margin Nearest Neighbors,(Blitzer, Weinberger, and Saul 2005)), and use the KNN without distance learning as the baseline. All the parameters involved in these methods are either chosen through 5 cross validation or using the default settings.

Fig. 4 gives the results. It can be seen that the performance of all the methods declines with the increasing of noise level. However, our RNCA method performs best consistently over all the datasets. Especially when the noise level is relatively high (30%), our method significantly outperforms other distance learning methods without taking label noise into account, such as NCA, ITML and LMNN.

Evaluation on Real World Dataset

To demonstrate the effectiveness of our approach on real world dataset with unknown label noise patterns, we constructed a database 'Horseface'[3] quickly and cheaply by searching for images using keyword 'horse face' and 'not horse face'[4], respectively. This results in 600 weakly labeled images with 300 images for each class. Suppose that we want to train a classifier to distinguish an image of horse face from other type of images using these. However, as illustrated in Fig. 5, images provided by the search engine are somewhat unreliable and we do not know exactly the patterns of label noise as well.

To evaluate the performance we relabel all the 600 images

[3]Collected by Google image search engine: available on request

[4]We use the query without the quotes and remove some portion of false positive samples manually to satisfy the noise level needed.

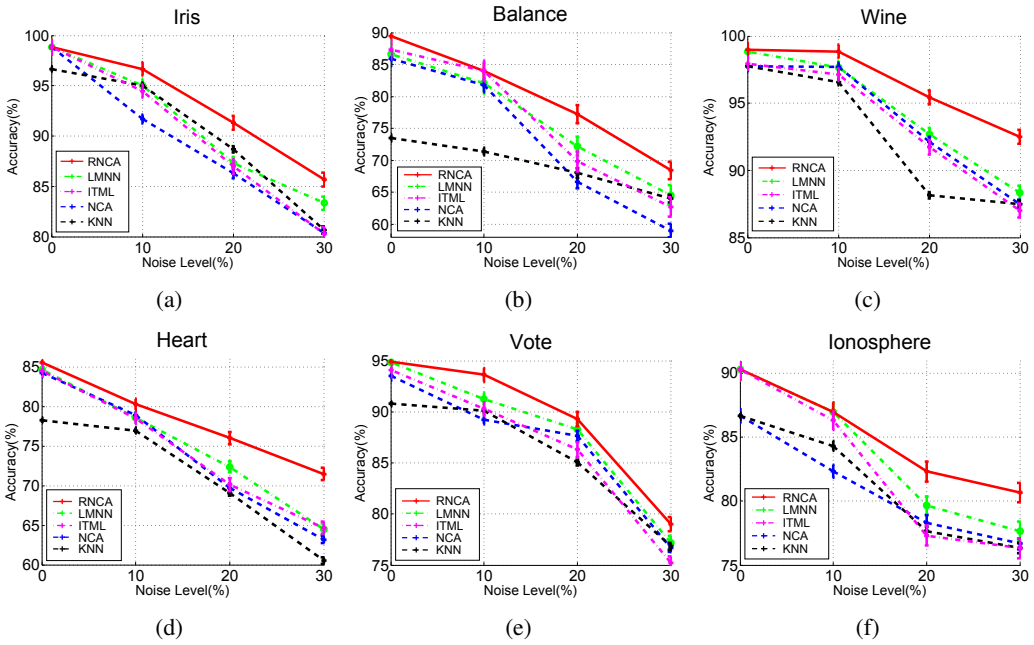

Figure 4: Comparison of the performance of our method with other distance learning methods in the present of label noise. (a-c) symmetric random label noise; (d-e) asymmetric label noise.

to provide a ground truth - an image will be labeled as 'horse face' (positive) only if there is at least one whole horse face shown in it, otherwise it is labeled as negative. Totally, there are 38 false positive images (i.e., with label noise) and 63 false negative images in the dataset.

For feature representation, we first resize each image to 100×100 without alignment or cutting. Then we extract a bag of SIFT features from each image by partitioning it into 400 overlapped grids with size 20×20 in pixels. A dictionary with 200 atoms is then constructed using these. So each image is encoded as a 200 dimensional vector. Besides comparing with several distance learning algorithms such as NCA, ITML, and LMNN, we also compare our algorithm with the SVM classifier with histogram intersection kernel (HIK), which is a widely used in object classification.

Figure.6 gives the performance with 5-folds cross validation. It can be seen that although the HIK method, which is specially designed for object classification, performs better than other distance learning algorithms such as ITML and LMNN, it may still be influenced by the label noise. On the other hand, our robust NCA distance learning algorithm performs best among the compared ones.

Conclusions

Noisy labels are almost inevitable in current machine learning applications, and they may result in suboptimal projection directions for similarity computation. We propose a robust distance learning algorithm in the presence of label noise based on the Neighbourhood Components Analysis method and demonstrate its effectiveness in both simulated data and a real world application with unknown label noise.

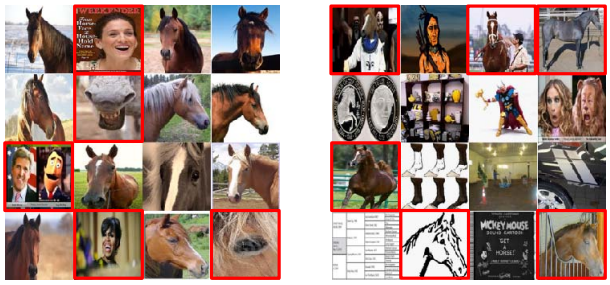

Figure 5: Illustration of typical images in the Horseface dataset harvested from the Web. (a) images in the positive category and (b) images with negative labels, where images with noisy labels are marked with a red square.

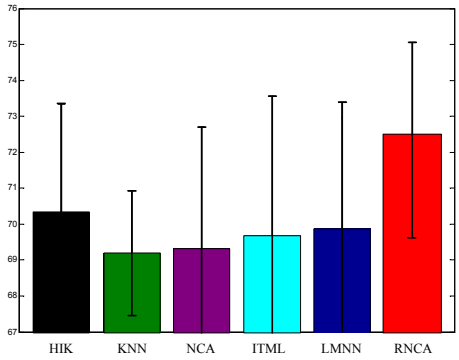

Figure 6: Comparison of classification performance of various algorithms on the Horseface dataset with label noise.

Acknowledgements

The authors want to thank the anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Science Foundation of China (61073112, 61035003, 61373060), Jiangsu Science Foundation (BK2012793), Qing Lan Project, Research Fund for the Doctoral Program (RFDP) (20123218110033).

References

Bergamo, A., and Torresani, L. 2010. Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In *Advances in Neural Information Processing Systems*, 181–189.

Biggio, B.; Nelson, B.; and Laskov, P. 2011. Support vector machines under adversarial label noise. *Journal of Machine Learning Research-Proceedings Track* 20:97–112.

Blitzer, J.; Weinberger, K. Q.; and Saul, L. K. 2005. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, 1473–1480.

Bootkrajang, J., and Kabán, A. 2012. Label-noise robust logistic regression and its applications. In *Machine Learning and Knowledge Discovery in Databases*. Springer. 143–158.

Cantador, I., and Dorronsoro, J. R. 2005. Boosting parallel perceptrons for label noise reduction in classification problems. In *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*. Springer. 586–593.

Davis, J. V.; Kulis, B.; Jain, P.; Sra, S.; and Dhillon, I. S. 2007. Information-theoretic metric learning. In *ICML*, 209–216.

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 1–38.

Fefilatyev, S.; Shreve, M.; Kramer, K.; Hall, L.; Goldgof, D.; Kasturi, R.; Daly, K.; Remsen, A.; and Bunke, H. 2012. Label-noise reduction with support vector machines. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, 3504–3508. IEEE.

Frénay, B., and Verleysen, M. 2013. Classification in the presence of label noise: a survey.

Goldberger, J.; Roweis, S.; Hinton, G.; and Salakhutdinov, R. 2004. Neighbourhood components analysis.

Howe, J. 2006. The rise of crowdsourcing. *Wired magazine* 14(6):1–4.

Kearns, M. J.; Schapire, R. E.; and Sellie, L. M. 1994. Toward efficient agnostic learning. *Machine Learning* 17(2-3):115–141.

Kulis, B. 2012. Metric learning: A survey. *Foundations & Trends in Machine Learning* 5(4):287–364.

Lawrence, N. D., and Schölkopf, B. 2001. Estimating a kernel fisher discriminant in the presence of label noise. In *ICML*, 306–313. Citeseer.

Leung, T.; Song, Y.; and Zhang, J. 2011. Handling label noise in video classification via multiple instance learning. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2056–2063. IEEE.

Natarajan, N.; Dhillon, I.; Ravikumar, P.; and Tewari, A. 2013. Learning with noisy labels. In *NIPS 26*, 1196–1204.

Pal, C.; Mann, G.; and Minerich, R. 2007. Putting semantic information extraction on the map: Noisy label models for fact extraction. In *Proceedings of the Workshop on Information Integration on the Web at AAAI*.

Park, K.; Shen, C.; Hao, Z.; and Kim, J. 2011. Efficiently learning a distance metric for large margin nearest neighbor classification. In *AAAI*.

Scott, C.; Blanchard, G.; and Handy, G. 2013. Classification with asymmetric label noise: Consistency and maximal denoising. In *COLT*, 489–511.

Van Hulse, J.; Khoshgoftaar, T. M.; and Napolitano, A. 2010. A novel noise filtering algorithm for imbalanced data. In *Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on*, 9–14. IEEE.

Wu, Y., and Liu, Y. 2007. Robust truncated hinge loss support vector machines. *Journal of the American Statistical Association* 102(479).

Yang, T.; Jin, R.; and Jain, A. K. 2010. Learning from noisy side information by generalized maximum entropy model. In *ICML*, 1199–1206.