Summary of Paper — Distance Metric Learning for Large Margin Nearest Neighbor Classification

Yan Chen
May 22, 2021

1 Introduction

Here we discuss the paper "Distance Metric Learning for Large Margin Nearest Neighbor Classification" [3] (Link of the paper). The authors learned a Mahanalobis distance metric for k-nearest neighbor (kNN) classification by semidefinite programming. The metric is optimzied with the goal that k-nearest neighbors always belong to the same class while examples from different classes are separated by a large margin.

The problem of kNN classification is that the distance metric should be adapted to the particular problem being solved. However it can hardly be optimal to use the same distance for different tasks. Previous approaches rely on learning from labeled examples, or even just linear transformation of input features to improve the kNN classification. And this paper builds in a novel direction on the success of these previous approaches. This paper is inspired by recent work on component analysis [2] and metric learning by energy-based models [1].

2 Model

Let $\{(x_i, x_j)\}$ denote a training set of n labeled examples with $x_i \in \mathbb{R}^d$ and discrete class labels y_i. use the binary matrix $y_{ij} \in \{0, 1\}$ to indicate whether or not the labels y_i and y_j match. The goal is to learn a linear transformation $L : \mathbb{R}^d \to \mathbb{R}^d$ to compute squared distances as:

$$D(x_i, x_j) = \|L(x_i - x_j)\|^2$$

The authors want to learn the linear transformation that optimizes kNN classification when the distances are measured in this way. And we use "target" neighbors to specify the inputs with the same label y_j that we wish to have minimal distance to x_i. Use η_{ij} to indicate whether input x_j is a target neighbor of input x_i.

The cost function is

$$\epsilon(L) = \sum_{ij} \eta_{ij} \|L(x_i - x_j)\|^2 + c \sum_{ijl} \eta_{ij}(1 - y_{il})[1 + \|L(x_i - x_j)\|^2 - \|L(x_i - x_l)\|^2]_+$$

Note that the first term penalizes large distances between each input and its target neighbors, which is obvious to see. And the second term penalizes small distances between each input and all other inputs that do not share the same label, which incorporates the idea of hinge loss. For each input x_i, the hinge loss is incurred by differently labeled inputs whose distances do not exceed the distance from input x_i to any of its target neighbors by one absolute unit of distance, i.e. $\forall j$ such that $\eta_{ij} = 1, \forall l$,

$$[1 + \|L(x_i - x_j)\|^2 - \|L(x_i - x_l)\|^2]_+ = \max(0, 1 - [\|L(x_i - x_l)\|^2 - \|L(x_i - x_j)\|^2])$$

Intuitively this can better help with classification because while it maintains a large margin of difference between differently labeled inputs, it also preserves the local neighborhood and makes sure that differently labeled inputs are separated "far" enough from each other.

,

The problem can be formulated as a semidefinite program (SDP). We can rewrite the squared distance above as

$$D(x_i, x_j) = (x_i - x_j)^T M (x_i - x_j)$$

where $M = L^T L$ parameterizes the Mahalanobis distance meetric induced by the linear transformation L. And by definition, if we set the slack variable ξ_{ijl} as

$$\xi_{ijl} = [1 + \|L(x_i - x_j)\|^2 - \|L(x_i - x_j)\|^2]_+$$

Then the resulting SDP can be given by (optimizing over semidefinite matrix M)

$$\textbf{Minimize}_{M} \sum_{ij} \eta_{ij}(x_i - x_j)^T M (x_i - x_j) + c \sum_{ijl} \eta_{ij}(1 - y_{il})\xi_{ijl} \textbf{ subject to:}$$

1. $(x_i - x_l)^T M (x_i - x_l) - (x_i - x_j)^T M (x_i - x_j) \geq 1 - \xi_{ijl}$
2. $\xi_{ijl} \geq 0$
3. $M \succcurlyeq 0$

3 Evaluation

The authors evaluated the algorithm on seven data sets. The authors found that

1. Except on the smallest data set, the Mahalanobis distance metrics led to significant improvements in kNN classification compared to Euclidean distance metrics, both in training and testing.
2. Energy-based classification with this assignment rule generally led to even furthere reductions in test error rates, where the energy-based classification is done by finding the label that minimizes the cost function $y_t = \arg\min_{y_t} \sum_j \eta_{tj} \|L(x_t - x_j)\|^2 + c \sum_{j, i=t \vee l=t} \eta_{ij}(1 - y_{il})[1 + \|L(x_i - x_j)\|^2 - \|L(x_i - x_l)\|^2]_+$
3. multiclass SVMs also achieved impressive results on most of the tasks.
4. According to figure 2 in the below, seems that Mahalanobis distance, energy based classification and multiclass are comparable in their testing error, while SVM outperforms all of the other methods except on the WINE datasets (a small dataset with less than 500 training examples and just three classes).
5. Mahalanobis distance also achieves impressive results in face recognition, spoken letter recognition, text classification and handwritten digit recognition compared to Euclidean distance.

	Iris	Wine	Faces	Bal	Isolet	News	MNIST
examples (train)	106	126	280	445	6238	16000	60000
examples (test)	44	52	120	90	1559	2828	10000
classes	3	3	40	3	26	20	10
input dimensions	4	13	1178	4	617	30000	784
features after PCA	4	13	30	4	172	200	164
constraints	5278	7266	78828	76440	37 Mil	164 Mil	3.3 Bil
active constraints	113	1396	7665	3099	45747	732359	243596
CPU time (per run)	2s	8s	7s	13s	11m	1.5h	4h
runs	100	100	100	100	1	10	1

Table 1: Properties of data sets and experimental parameters for LMNN classification.

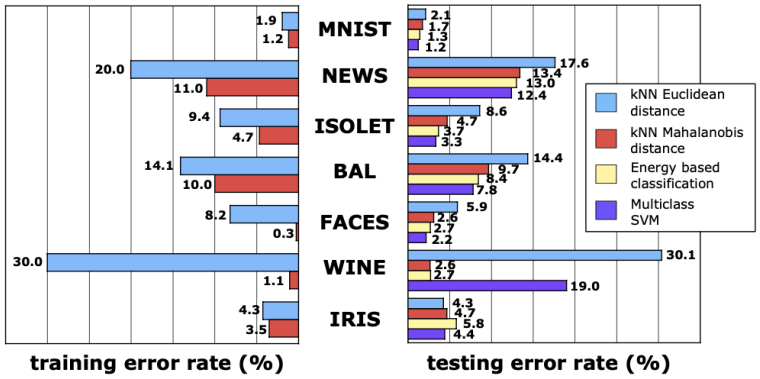

Figure 2: Training and test error rates for kNN classification using Euclidean versus Mahalanobis distances. The latter yields lower test error rates on all but the smallest data set (presumably due to over-training). Energy-based classification (see text) generally leads to further improvement. The results approach those of state-of-the-art multiclass SVMs.

4 Comment

This method is innovative in learning a novel distance matrix in kNN classification problem to better adapt to particular problem being solved. As the authors point out, it can hardly be optimal to use the same distance metric for different tasks, for example, even for face recognition and gender identification, even if in both tasks, distances are computed between the same fixed-size images. So both intuitively and empirically, we can see that this data-adapted distance metric learning for classification algorithm outperforms the simple Euclidean metrics in many tasks, and it's comparable to some other well-performing methods, including energy-based, multi-class SVM, etc. This is not very surprising meanwhile, by considering the fact that it's greatly inspired by the energy-based model and incorporated the hinge-loss function penalty as in SVM. And it's also very nice to use the idea of semidefinite programming (SDP) in this algorithm.

For future directions, the authors pointed out three directions of the LMNN classification:

1. apply LMNN classification to problems with very large number of classes (hundreds or thousands of), where it seems that its advantages are most apparent.
2. kernel trick to perform LMNN in nonlinear feature spaces.
3. learn locally adaptive distance metrics that vary across the input space.

References

[1] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[2] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. *Advances in neural information processing systems*, 17:513–520, 2004.

[3] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, 10(2), 2009.