Machine Learning Assignment Final Project

Name: Que Shuhao
NetID: sque
Student Number: 4739124
Word Count: 1000

July 4, 2018

General Steps of the Best-Performance Method

1. Step 1: Perform Principle Component Analysis(PCA) on *train.csv*
2. Step 2: Standardize the dataset *train.csv* using Z-Score Transformation
3. Step 3: Initialize K-means Clustering Algorithm with two pre-defined centers, which are calculated based on *train.csv* (without standardization)
4. Step 4: Perform Principle Component Analysis(PCA) (fitted on *train.csv*) on *test.csv*. Apply K-means Clustering on the dataset *test.csv* and store the predicted labels.
5. Step 5: Standardize *test.csv* the same way as *train.csv*
6. Step 6: Use *test.csv* and predicted labels plus labeled *train.csv* to train LDA(Linear Discriminant Analysis) classifier
7. Step 7: Apply the trained LDA classifier to *test.csv* and get the final 20000 predicted labels.

Estimate of the Performance on *test.csv* (20000 unlabeled samples)

Since the true labels for *test.csv* are not provided, the accuracy of my method cannot be said with definite certainty. But according to the experiment results, which will be specified in the next section, it is estimated that the my training method and classifier can achieve around 10% of error rate.

Justification and Evaluation

0.1 Standardization of Labeled Dataset *train.csv*

Since the data are contaminated with uniform noise, whose parameters are unknown, instead of trying to eliminate the noise, I perform z-score transformation to standardize the data using the following formula:

$Z = \frac{X-mean}{sd}$
where sd denotes the standard deviation, which is given by
$\sqrt{\frac{\sum(X-mean)^2}{N}}$, where N denotes the number of samples in the data
After standardization, the data are expected to be placed on a normal distribution, which makes it easier to be classified.

	Leave-one-out cross validation on train.csv Test Error	Leave-one-out cross validation on train.csv Train Error
LDA	22.5%	0%
SVM('RBF')	100%	0%
SVM('Linear')	25%	0%
SVM('Sigmoid')	27.5%	46.0%

Table 1: Classificaton Error on *train.csv* using leave one out cross validation; without standardization; with PCA

As shown in Tables 1, 2, 3 and 4, non-standardized dataset has negative impact on the performance of SVM classifier with kernels "RBF" and "Sigmoid". Although it seems that it has little effect on LDA classifier, which is also my final choice, considering the implementation of k-means clustering, I chose to implement standardization of dataset after all.

	Leave-one-out cross validation on train.csv Test Error	Leave-one-out cross validation on train.csv Train Error
LDA	40%	0%
SVM('RBF')	100%	0%
SVM('Linear')	23.5%	0%
SVM('Sigmoid')	100%	49.7%

Table 2: Classificaton Error on *train.csv* using leave one out cross validation; without standardization; without PCA

0.2 Principle Component Analysis

The number of principle components are tuned as 148(which holds 98.4% of variance of the total dataset). As shown in Table 3 and Table 4, it is observed that besides for LDA and K-NN classifiers, principle component analysis does not improve the accuracy rates much and sometimes it even worsens the results. However, PCA can relatively reduce the effect of the noise[1] (not eliminating them) to improve the performance of clustering. According to the experiment, with PCA, 2-means clustering with given initialized centers achieves good result of predicting labels for *train.csv.* The predicted labels only have 10% (20/200) error compared with the true labels. Without PCA, 2-means clustering gives predicted labels with very bad quality, whose error rate reaches more than half. Therefore, in order to reduce the effect of noise and to guarantee the performance of k-means clustering on *test.csv*, I decided to implement PCA on both datasets.

0.3 K-means Clustering on *test.csv*

Since there are no true labels provided for *test.csv*, considering there are 20000 samples in this very dataset, it occurred to me that it might be useful to use this dataset to train the classifier. Therefore, K-means Clustering is applied to obtain predicted labels, where K is defined as 2. If 2-means is directly applied to *test.csv* without initializing the 2 centers, the error rate obtained by using *train.csv* as test dataset is 39.5%, which is very high. Therefore, the tactic that I thought of was to initialize the 2-means clustering with 2 given initial centers, which are calculated by averaging the feature vectors belonging to label 1 and those belonging to label 2. Using these two initialized centers, the error rate obtained is reduced to only 11%.

0.4 Choosing Optimal Classifer

When choosing the optimal classifier, I tried LDA, K-NN, Adaboost and SVM. The test results are shown in Table 4. It is observed that the distribution of data in both *train.csv* and *test.csv* to certain degree are linearly separable(not completely), in which case linear classifiers all achieve relatively good results.

It is also observed that K-NN classifier is not suitable for this very dataset considering the fact that the samples even with different labels inside the dataset are very similar to each other in terms of Euclidean distance, which makes choosing the nearest neighbors not effective to correctly label one sample.

	Leave-one-out Cross Validation on train.csv	test.csv as training dataset train.csv as test dataset
LDA	24%	11%
Adaboost	39%	16.5%
K-NN (K = 30)	35.5%	20.5%
SVM('RBF')	24.5%	11.5%
SVM('Linear')	25.5%	11%
SVM('Sigmoid')	28%	10.5%

Table 3: Test error rates achieved by different classifiers based on two different training and testing schemes; with PCA

	Leave-one-out Cross Validation on train.csv	test.csv as training dataset train.csv as test dataset
LDA	40%	9%
Adaboost	33%	20.5%
K-NN (K = 30)	35%	26.5%
SVM('RBF')	24.5%	11%
SVM('Linear')	24%	12.5%
SVM('Sigmoid')	22.5%	11.5%

Table 4: Test error rates achieved by different classifiers based on two different training and testing schemes; without PCA

0.5 Semi-supervised Learning

Another test and training scheme I applied is semi-supervised learning. First I apply 2-means clustering to predict labels for unlabeled dataset *test.csv*, and then I concatenate this dataset with randomly selected 20 samples from *train.csv* to perform 10-fold cross validation, which implies that each time 20000 + 20 samples are used as training data and the other 180 samples from *train.csv* is used as test data. Test results are shown in Table 5

	10-fold cross validation Test Error	10-fold cross validation Train Error
LDA	9%	1.16%
SVM('RBF')	10%	0.04%
SVM('Linear')	11%	0.23%
SVM('Sigmoid')	11%	2.86%

Table 5: Error Rate Based on Semi-Supervised Learning; with PCA

0.6 Conclusion

Considering all the influencing factors discussed in previous sections, the final chosen method is given by

Learning Scheme: After principle component analysis, label prediction by clustering and standardization on the dataset, both *train.csv* and *test.csv* are applied to train the LDA classifier.

Classifier: LDA(Linear Discriminant Analysis) classifier achieves the best results in general and it takes way less time to compute. Therefore, the chosen classifier in the end to predict labels for 20000 samples in *test.csv* is LDA classifier.

References

[1] Lan Du, Baoshuai Wang, Penghui Wang, Yanyan Ma, and Hongwei Liu. Noise reduction method based on principal component analysis with beta process for micro-doppler radar signatures. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(8):4028–4040, 2015.