Discriminant Analysis of LIBRAS Hand Movement

Xiaoxi Ma

Department of Statistics and Biostatistics, Rutgers University

xiaoxi.ma1207@rutgers.edu

1 Background

1.1 Executive Summary

Data set contains 15 types of hand movement performed by Libras, the task is to classified those hand movements. Hand movement is a type a gestures commands, nowadays, technology of recognizing gestures commands is more and more important to our human's daily life eg: Disabled people may use head movements to control the switch or computer; People can take a hands-free selfie; People can play Somatosensory game like X box 360. To analysis, I have used both linear and non-linear classification method, it turns out that SVM performs best for this type of data. This findings can help those companies mentioned above to recognize gestures commands better.

1.2 Introduction

Previously, a hybrid architecture and a multiple classifier architecture using fuzzy-connectionist and heuristic strategies to recognize each type of primitive have been used to recognize gesture movement [1], recognition rate is quite satisfactory. Also a real-time tracking method and hidden Markov models have been used to recognize hand gesture [2]. Those methods are more related to computer science. This paper, I want to based on some common statistical methods to solve the classification problem.

The rest of this report is organized as following: Dimension reduction techniques like PCA and FDA are presented in Section 2.1. Sections 2.2-2.6 includes a brief introduction of five candidate methods: K-nearest neighbors algorithm, LDA, Lasso regularization of generalized linear model, support vector machine, tree-base method. Cross-Validation has been used to select parameters and get the test error. Section 3 is the result. Section 4 is the conclusion.

1.3 Data

The Libras Movement Data Set have 91 attributes (1 response and 90 variables) and 360 instances, which contains 15 classes of 24 instances each. 15 classes are 15 different signs described by their characteristic hand movement over 45 frames and the current x- and y- positions of hand were recorded. The 90 variables are abscissa and ordinate of 45 frames of the hand movement. Since all record signs are mapped into normalized in the unitary space, no transformation of data are needed. I randomly choose 240 instances as the train data set, the rest 120 instances as the test data. The original data set is available on UCI Machine Learning Repository:

http://archive.ics.uci.edu/ml/datasets/Libras+Movement

Table 1. : 15 classes of Libras hand movement

15 classes of Libras hand movement		
1: curved swing	2: horizontal swing	3: vertical swing
4: anti-clockwise arc	5: clockwise arc	6: circle
7:horizontal straight-line	8: vertical straight-line	9: horizontal zigzag
10: vertical zigzag	11: horizontal wavy	12: vertical wavy
13: face-up curve	14: face-down curve	15: tremble

Figure 1 is the visualization of 10 randomly picked actual hand movement. We can tell that there are some vertical zigzag, horizontal swing and so on.

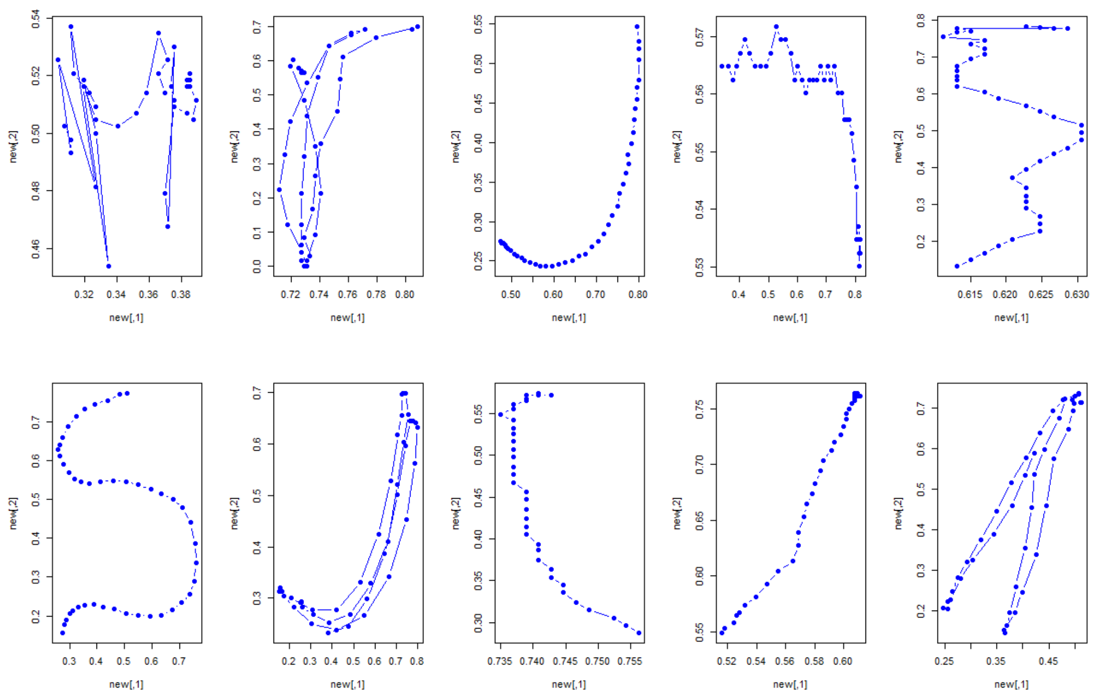

Figure 1:Visualization of 10 randomly picked actual hand movement

2 Analysis Methods

2.1 Dimension Reduction

Since data set have 91 attributes, it is important to apply dimension reduction techniques, and here principal component analysis (PCA) and Fisher discriminant analysis (FDA) are performed. Figure 2 shows the cumulative percentages of sum of singular values of the training data. As the largest five singular values explain more than 80 percentages of variation in the training data, I take the corresponding first five principle components.

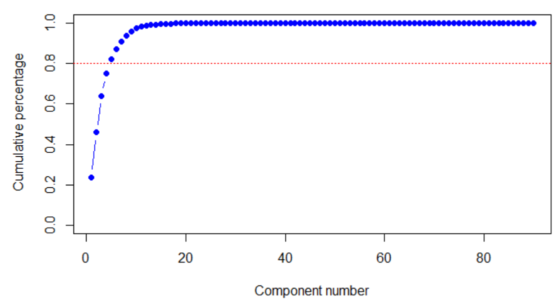

Figure 2: Cumulative percentages of singular values of the design matrix for training data.

Figure 3 shows the PCA projection and FDA projection with only the first two components. It seems that the PCA projection do not perform well because classes are all mixed together. FDA projection provides a better separation, we can clearly see the separation of 15 classes. Thus, the following analysis will use the FDA projection instead of PCA projection. However, there are still some classes mingle with each other, thus, only first two components of FDA is not enough, I used Cross-Validation to choose the best number of components.

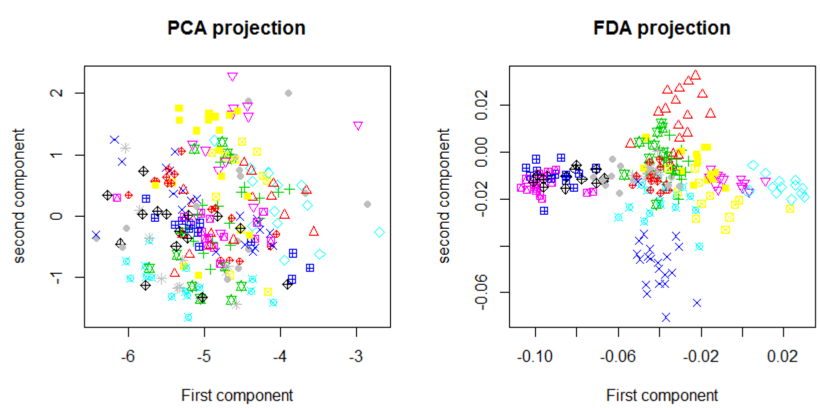

Figure 3: PCA projection and FDA projection for training data.

2.2 Lasso Regularization of Generalized Linear Models

Since there are 15 classes, logistic regression alone does not perform well, train error is close to 0.5. Thus I tried Ridge Regularization of Generalized Linear Models and Lasso Regularization of Generalized Linear Models, which turns out that Lasso Regularization of Generalized Linear Models performs well.

2.3 LDA

I performed Dimension Reduction to LDA since LDA do not perform best when number of variables are big. I randomly choose four fifths of the samples for training, the rest for testing, the result of Cross-Validation suggests that LDA using first 22nd PCA component gives the smallest CV test error and LDA using first 10th FDA component gives the smallest CV test error.

2.4 K-NN-1

k-Nearest Neighbors algorithm is a non-parametric method for classification. That is to say K-NN method does not make any assumptions on the underlying data distribution. This method is very simple to understand but very powerful in practice use. I set train error as the criteria to choose the best k. As we can see from the Figure 4, when k=1, the train error is the smallest which is 0.000, the corresponding test error is also the smallest :0.2083.

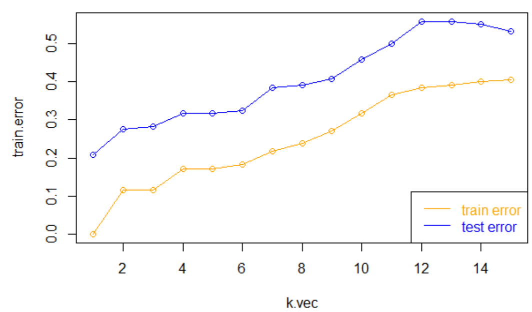

Figure 4: Train error corresponding to the k

2.5 Multi-class Support Vector Machine

SVM can have Non-linear decision boundaries for support vector classier by replacing the linear kernel with other kernels and non-linear decision boundaries would perform better than linear decision boundaries for my data set. I choose to use the polynomial kernel since this kernel gives me the smallest test error which is 0.1583. Visualization of SVM classifier can only be provided when there are only two classes.

2.6 Tree-Based Methods

Although decision tree is not the most accurate compared to other classification method, it is quite simple and very useful for interpretation. Furthermore, it's accuracy can be improved by applying random forest and boosting. I applied two methods: Maboost and Random Forest. Gini index shows that variable 28, 30, 2, 4 are most important nodes.

3 Results

Table 2. : Train and test error for all methods

Methods	Train	Test Error
K-NN-1	0.0000	0.2083
Ridge Regularization of Logistic regression	0.2792	0.4583
Lasso Regularization of Logistic regression	0.0625	0.2333
LDA	0.1208	0.325
LDA using first 22^{nd} PCA components	0.1833	0.325
LDA using first 10^{th} FDA components	0.1667	0.1875
SVM	**0.0000**	**0.1583**
Tree- Base: Maboost	0.025	0.2833
Tree- Base: Random Forest	0.2292	0.2333

As we can see from the Table 2, we can see that SVM performs best with zero training error and 0.1583 test error; K-NN-1, Lasso Regularization of Logistic regression, LDA using first 10th FDA component and Random Forest perform well with test error around 0.2; Accuracy of LDA improved by dimension using first 10^{th} FDA components.

4 Conclusions

By using dimension reduction technical-FDA can largely improve the preforms of LDA. SVM gives the smallest test error and preforms the best during the five candidate methods. In order to improve the accuracy of the classification, I would like to do more research on the body movement with respect to time and do more experiment to get the data for cross-validation on the number of frames and get the results of the most suitable number to be the variables. Also I would try other aspects of methods that can apply to this type of data.

References

[1] D. B. Dias, R. C. B. Madeo, T. Rocha, H. H. Bíscaro, and S. M. Peres, “Hand movement recognition for brazilian sign language: A study using distance-based neural networks,” in Proceedins of International Joint Conference on Neural Networks, pp. 697–704, June 2009.

[2] F.-S. Chen, C.-M. Fu, and C.-L. Huang, "Hand gesture recognition using a real-time tracking method and hidden markov models", Image and Vision Computing, vol. 21, no. 8, pp. 745-758, August 2003.