CEG7570 Project

Part III

March 14, 2017

Given a data set of N points from two or more classes where each point has d = 3 or more features, in Part I of the project, the outliers were removed and each feature was normalized to have a mean of 0 and standard deviation of 1. In Part II of the project, by principal component analysis, the number of features was reduced from d to 2. In this last part of the project, we will build a Bayes classifier to classify new points.

To simplify implementation of the classifier, we assume the same number of points in each class is available. Therefore, after removing the outliers, the data sets are modified to have the same number of points in each class. To do this, after removing the outliers, find the number of points in the smallest class. Suppose this is n. Then, keep the top n points in each class and run your programs for Part I and Part II on the updated data sets.

Since PCA removes linear dependency between features, we can consider the new features independent of each other (although nonlinear dependencies still exist). This means the covariance matrix of the points obtained after PCA will be diagonal (if nonlinear dependencies do not exist) with diagonal entries showing the variances of the features.

To create a minimum distance classifier, determine the mean and standard deviation of each feature obtained from PCA and normalize the feature to have a standard deviation of 1. Do not change its mean.

The 2-D normal distribution obtained for different classes in the new data set will have different means but the same standard deviation, and so the decision boundary between two classes will be the plane normal to the line connecting the means of the two classes and passing midway between the two means.

Points falling within the decision boundary are of the same distance to the means of the classes on the two sides of the boundary. Points falling on the right side of the boundary are closer to the class-mean on the right side of the boundary and points falling on the left side of the boundary are closer to the class-mean on the left side of the boundary.

Therefore, for any new point **p**, the point is normalized using the means and standard deviations of the features obtained in Part I to obtain normalized point **q**. Then, point **q** is projected to the two principal components obtained in part II. Let's call the new point **r**. By normalizing the two features of **r** with respect to the means and standard deviations of the features obtained after PCA analysis a new point **s** is obtained. To determine the class **s** belongs to, the distance of **s** to the mean of each class is determined. The class-mean closest to **s** determines the class of point **s**, and consequently the class of point **p**.

The following summarizes the steps involved in classifying an unknown point:

1. Suppose a data set with N points from C classes, where each point containing d > 2 features is available.
2. Remove possible outliers. These are points that are father from the mean by more than 3 standard deviations.
3. Remove points from the larger classes so that all classes have the same number of points.
4. Normalize features in the modified data set so that each feature has a mean of 0 and a standard deviation of 1.
5. Find the covariance matrix of the points in Step 4.
6. Through PCA analysis of the covariance matrix, reduce the number features from d to 2.
7. Find the mean and standard deviation of each transformed feature and normalize the features so each has a standard deviation of 1. DO NOT CHANGE THE MEANS OF THE FEATURES.
8. Find the mean of points in each class.
9. The program is now ready to read in a new point and classify it. Read the coordinates of a point **p** (a vector of d values) from input.
10. Normalize the features using the means and standard deviations obtained in Step 4 to obtain new point **q**.
11. Transform point **q** using the two principal components obtained in Step 6 to obtain point **r**.
12. Normalize point **r** using the standard deviations found in Step 7 to obtain point **s**. Note that this normalization is with respect to the mean of points obtained after PCA analysis.
13. If point **s** is closer to the mean of class w_i than to the mean of any other class, point **p** is considered to belong to class w_i.

Allow the user to enter new points, one at a time. Your program should then determine the class of each point. Note that each point will have d features in their original scales.

Write a short report (5 to 10 pages), evaluating the implemented classifier. Include snapshots of the results from each Part of the project. What is the recognition rate when using the training data as the input? That is, what percentage of points in the training data set is correctly classified by the implemented classifier?

Upload your program and report to Pilot under Project Part III by **April 18, 2017, 4:30 PM**.