Coursework 4

Please submit one file (can be any reasonable format) to the dropbox 4 on moodle. Please paste any R code, or plots or results obtained by running R code to this file, if required.

1. SVM (1%) [Textbook 9.7.8]

This problem involves the OJ data set which is part of the ISLR package.

(a) Create a training set containing a random sample of 800 observations, and a test set containing the remaining observations.

(b) Fit a support vector classifier to the training data using cost=0.01, with Purchase as the response and the other variables as predictors. Use the summary() function to produce summary statistics, and describe the results obtained.

(c) What are the training and test error rates?

(d) Use the tune() function to select an optimal cost. Consider values in the range 0.01 to 10.

(e) Compute the training and test error rates using this new value for cost.

(f) Repeat parts (b) through (e) using a support vector machine with a radial kernel. Use the default value for gamma.

(g) Repeat parts (b) through (e) using a support vector machine with a polynomial kernel. Set degree=2.

(h) Overall, which approach seems to give the best results on this data?

2. SVM and Logistic Regression (1.5%) [Textbook 9.7.5]

We have seen that we can fit an SVM with a non-linear kernel in order to perform classification using a non-linear decision boundary. We will now see that we can also obtain a non-linear decision boundary by performing logistic regression using non-linear transformations of the features.

(a) Generate a data set with n = 500 and p = 2, such that the observations belong to two classes with a quadratic decision boundary between them. For instance, you can do this as follows:

```
> x1=runif (500) -0.5
> x2=runif (500) -0.5
> y=1*( x1^2-x2^2 > 0)
```

(b) Plot the observations, colored according to their class labels. Your plot should display X_1 on the x-axis, and X_2 on the y-axis.

(c) Fit a logistic regression model to the data, using X_1 and X_2 as predictors.

(d) Apply this model to the *training data* in order to obtain a predicted class label for each training observation. Plot the observations, colored according to the *predicted* class labels. The decision boundary should be linear.

(e) Now fit a logistic regression model to the data using non-linear functions of X_1 and X_2 as predictors (e.g. X_1^2, $X_1 \times X_2$, $\log(X_2)$, and so forth).

(f) Apply this model to the *training data* in order to obtain a predicted class label for each training observation. Plot the observations, coloured according to the *predicted* class labels. The decision boundary should be obviously non-linear. If it is not, then repeat (a)-(e) until you come up with an example in

which the predicted class labels are obviously non-linear.

(g) Fit a support vector classifier to the data with X_1 and X_2 as predictors. Obtain a class prediction for each training observation. Plot the observations, colored according to the *predicted class labels*.

(h) Fit a SVM using a non-linear kernel to the data. Obtain a class prediction for each training observation. Plot the observations, coloured according to the *predicted class labels*.

(i) Comment on your results.

3. Hierarchical Clustering (1%) [Textbook 10.7.9]

Consider the USArrests data. We will now perform hierarchical clustering on the states.

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states.

(b) Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?

(c) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.

(d) What effect does scaling the variables have on the hierarchical clustering obtained? In your opinion, should the variables be scaled before the inter-observation dissimilarities are computed? Provide a justification for your answer.

4. PCA and K-Means Clustering (1.5%) [Textbook 10.7.10]

In this problem, you will generate simulated data, and then perform PCA and **K**-means clustering on the data.

(a) Generate a simulated data set with 20 observations in each of three classes (i.e. 60 observations total), and 50 variables.

Hint: There are a number of functions in R that you can use to generate data. One example is the rnorm() function; runif() is another option. Be sure to add a mean shift to the observations in each class so that there are three distinct classes.

(b) Perform PCA on the 60 observations and plot the first two principal components' eigenvector. Use a different color to indicate the observations in each of the three classes. If the three classes appear separated in this plot, then continue on to part (c). If not, then return to part (a) and modify the simulation so that there is greater separation between the three classes. Do not continue to part (c) until the three classes show at least some separation in the first two principal component eigenvectors.

(c) Perform **K**-means clustering of the observations with **K** = 3. How well do the clusters that you obtained in **K**-means clustering compare to the true class labels?

Hint: You can use the table() function in R to compare the true class labels to the class labels obtained by clustering. Be careful how you interpret the results: **K**-means clustering will arbitrarily number the clusters, so you cannot simply check whether the true class labels and clustering labels are the same.

(d) Perform **K**-means clustering with **K** = 2. Describe your results.

(e) Now perform **K**-means clustering with **K** = 4, and describe

your results.

(f) Now perform **K**-means clustering with **K** = 3 on the first two principal components, rather than on the raw data. That is, perform **K**-means clustering on the 60 × 2 matrix of which the first column is the first principal component's corresponding eigenvector, and the second column is the second principal component's corresponding eigenvector. Comment on the results.

(g) Using the scale() function, perform **K**-means clustering with **K** = 3 on the data after scaling each variable to have standard deviation one . How do these results compare to those obtained in (b)? Explain.