**ISQA 8080 Assignment 5 Due: By Tuesday, Dec. 17 2018, 5:30 PM**

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**NOTES:**

1. Use R for the calculations and implementation.
2. Submit all documents in a zip file and upload it to Canvas. Name your Zip Folder with your name, A5, and the course # (Example: LastName-A5-ISQA 8080).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Non-Linear Regression (45 points)**

We consider a data set that captures the relationship between two variables, x and y.

In this problem, you will develop different non-linear regression models and test their fit on a test dataset. The data is provided in the Problem1.csv file.

1. In class, we looked at different ways of modeling non-linear relationships between our target and independent variable(s). In your own words, describe the difference between global functions (such as polynomials) and local functions (e.g., step functions).
2. Regression Splines and Smoothing Splines are two commonly used methods to build non-linear models. In your own words, describe the difference between the two methods. Which one is conceptually close to LASSO and Ridge Regression, and why?
3. Split the data into a training and test set (70% / 30%). Use a unique random seed to create the split (you can use set.seed(your\_NUID) for this). Plot the training set and comment on it.
4. Build a linear regression using y as dependent and x as independent variable. Predict y for the test set, and calculate the Root Mean Squared Error (RMSE) for the test set. Does a linear regression seem adequate for this data?
5. Build a polynomial regression model of degree 4 on the training dataset, and create the predictions on the test dataset. Calculate the RMSE for the test dataset, and how it here. (Optional: plot the polynomial regression using the training data).
6. Now, build a natural splines model on the training set, using degrees of freedom of 10. Calculate the predictions and RMSE for the test dataset. How does it compare to the previous polynomial regression?
7. Finally, calculate a local regression on the training dataset. Try two different values for the span, 0.2 and 0.5, and calculate the test set predictions and RMSE for both span values. Which one works best, and how does it compare to the models in part c and d? Optional: visualize the local regression compared to the linear regression.
8. **Unsupervised Learning – Clustering (1) (15 points)**

In this problem, you’ll manually construct a k-means clustering with k = 2. Note: You don’t have to use R for this, feel free to use the tool of your choice or even manually calculate the steps below. Use the following data:

|  |  |  |
| --- | --- | --- |
| Observation | X1 | X2 |
| 1 | 0 | 5 |
| 2 | 1 | 5 |
| 3 | 2 | 3 |
| 4 | 3 | 2 |
| 5 | 4 | 0 |

1. Assign a random number between 0 and 1 to each observation. E.g., you can create 5 random numbers by calling runif(5) in R. If the random number for an observation is <0.5 assign cluster 1, otherwise cluster 2. Show the initial cluster labels and visualize them in a scatterplot.
2. Compute the centroid for each cluster. Specifically, the cluster centroid refers to the average values of X1 and X2 for the observations in the respective cluster. Show the centroids here.
3. For each observation, calculate the squared Euclidian distance to the two cluster centroids. Assign a new cluster label for each observation based on which cluster centroid is closest (has the smallest distance). Show the new cluster labels for the 5 observations.
4. Repeat steps b and c until no cluster labels need to be changed. I.e., re-calculate the cluster centroids, similar to step b, then re-calculate the squared Euclidian distances to the two new cluster centroids for each observation. Check if any of the cluster labels need to be changed. If yes, repeat b and c again, otherwise stop. Report the final cluster labels, and visualize them in a scatterplot.
5. **Unsupervised Learning – Clustering (2) (40 points)**

Consider the CustomerData csv file. It includes information about several hundred supermarket customers and their spending on various types of products. We will now perform k-means and hierarchical clustering on the customers to segment them into groups with different spending behavior.

1. Using only the numeric variables in the data, determine a good number of clusters by creating the scree plot for the k-means approach. To do this, create k-means clusters with k = 1 to k = 15, and plot the respective within-cluster-variance as done in class. Plot the scree plot. What would be a good number of clusters in your opinion?
2. Given the number of clusters that you determined in the previous step, build a k-means clustering. Append the cluster vector to the original data set (including the categorical variables). Visualize the cluster differences. For example, you can use ggpairs() for this. Do you see differences between the clusters, i.e., different spending behavior?
3. Then, let’s create a hierarchical clustering. Use the complete linkage and single linkage approaches to create two different hierarchical clusters. Cut the dendrograms such that we get the same number of clusters that you used in steps a and b. Compare the number of observations in each cluster between the complete linkage and single linkage method (e.g., summary(complete\_linkage\_cluster) vs summary(single\_linkage\_cluster). What do you observe?
4. Due to the few observations with extreme values, management decided to remove some of the most obvious outliers in the data set. Rebuild the hierarchical clustering models (complete and single linkage) using the CustomerData\_OutlierRemoved.csv file, and cut the dendrogram such that we get the same number of clusters as before. How many observations do we have in each cluster for complete and single linkage now?
5. Finally, append the cluster variable from the complete linkage approach to the original data set, and visualize the cluster differences again (e.g., using ggpairs). Can we identify customer groups with different spending behavior? How do these clusters compare to the ones you created in step a/b?