15.071: The Analytics Edge **Fall 2022**
Homework 5: Clustering

Out: November 2; Due: November 16, 11:59 pm.
Please post the assignment in pdf format with file name "Lastname_15071_HW4.pdf".
For each question, please include the main `R` *commands that you used in your submission.*

Problem 1: Clustering Stock Returns [50 pts]

When building portfolios of stocks, investors seek to obtain good returns while limiting their variability. This can be achieved by selecting stocks that show different patterns of returns—a technique known as *diversification*. To support these decisions, we will identify clusters of stocks that exhibit similar patters.

For this problem, we will use the dataset `returns.csv`. This file contains monthly returns from some stocks among the S&P500 from March 2006 through February 2016. Each observation (row) corresponds to a company. The variables in the dataset are described in Table 1.

Table 1: Variables in the dataset `returns_final.csv`.

Variable	Description
`symbol`	The symbol identifying the company of the stock.
`Industry`	The industry sector under which the stock is classified.
`avg200603 - avg201602`	The return for the stock during the variable's indicated month. The variable names have format "avgYYYYMM", where YYYY is the year and MM is the month. For instance, variable avg200902 refers to February 2009. The value stored is a net increase or decrease of the end of month stock price over the stock price at the beginning of the month. For instance, a value of 0.05 means the stock had a net increase on average of 5% during the month, while a value of -0.02 means the stock had a net decrease on average of 2% during the month. There are 120 of these variables, for the 120 months in our dataset.

Import the dataset using the following command.

```
data = read.csv("returns.csv")
```

Note that stock returns are provided in Columns 3 through 122. You may find it useful to create a second dataset with only these values.

```
returns = data[,3:122]
```

a) How many companies are there in each industry sector? Entering the "Great Recession" of 2008-2009, most stocks lost significant value, but some sectors were hit harder than others. For each sector, plot the average stock return between 01/2008 and 12/2010. In September 2008, which industries had the worst average return across the stocks in that industry? Is industry information sufficient for investors to build a portfolio diversification strategy? **[10 pts]**

   [**Hint**: To average quantities by industry, consider using the `aggregate` function.]

b) Let us now cluster the stocks according to their monthly returns. In this analysis, we will not normalize our data prior to clustering. Why is this a valid approach for this problem and dataset? Cluster the data using hierarchical clustering, using the "Ward D2" measure of cluster dissimilarity. Plot the dendrogram and the scree plot. What do you think is a reasonable number of clusters for this problem? Justify your choice. **[10 pts]**

c) Use the **`cutree`** function to cut the dendrogram tree to 8 clusters and extract cluster assignments. Compute the number of companies in each cluster and the number of companies per industry sector in each cluster. What are the average returns by cluster in October 2008 and in March 2009? Characterize each cluster qualitatively based on these results, and any other relevant information. **[10 pts]**

   [**Hint**: To compute the number of companies in each cluster and the number of companies per industry sector in each cluster, consider using the **`table`** function.]

d) Cluster the data using the k-means clustering algorithm, using the same number of clusters as in Question c. Please run "**`set.seed(2407)`**" before running k-means clustering, and set the maximum number of iterations of k-means clustering to 100.
   What are the cluster centroids in October 2008 and in March 2009? Extract the cluster assignments and compare them to those obtained with hierarchical clustering. In what ways are the clusters similar vs. different across the two models? **[10 pts]**

   [**Hint**: To answer this question, you should look at the number of observations in each cluster and the industry sectors of the companies in each cluster. Do some k-means clusters match a hierarchical cluster? Do some k-means clusters look very different from every one of the hierarchical clusters?]

e) The **silhouette metric** measures how similar an observation is to its own cluster compared to other clusters; this is done by comparing the distance from the observation to other observations in its cluster with the distance from the observation to other observations in the second closest cluster. More precisely, the silhouette metric for observation i is computed as

$$s(i) = \frac{b(i) - a(i)}{\max(b(i), a(i))},$$

   where $a(i)$ is the average distance from observation i to the other points in its cluster, and $b(i)$ is the average distance from observation i to the points in the second closest cluster. This score ranges from -1 to 1 and a higher score is better. Observe that the silhouette metric is computed individually for each observation in the data - these individual scores can be averaged to reflect the quality of the global assignment.

   Use the function **`silhouette(cluster_assignment,distances)`** to compute the silhouette scores that correspond to the assignments obtained via hierarchical clustering (question c)) and via k-means clustering (question d)). The first argument of the method is a cluster assignment (e.g., the $**`cluster`** component of the object returned by the **`kmeans`** function) and the second argument is the matrix of pairwise distances between all data points. Report the mean silhouette score of each assignment, both per cluster (using the **`aggregate`** function) and overall. Plot the results using **`plot(cluster_assignment, col=1:8, border=NA)`**. **[5 pts]**

   [**Hint**: Run **`?silhouette`** to see exactly what is returned by the **`silhouette`** function.]

f) Write a short paragraph to an investor describing your model and how it can be used to inform investment strategies. **[5 pts]**

Problem 2: eBay.com [50 pts]

The file `eBayAuctions.xls` contains information on 1972 auctions transacted on eBay.com during May-June 2004. The goal is to use these data to build a model that will classify competitive auctions from noncompetitive ones. A competitive auction is defined as an auction with at least two bids placed on the item auctioned. The data include variables that describe the item (auction category), the seller (his/her eBay rating), and the auction terms that the seller selected (auction duration, opening price, currency, day-of-week of auction close). In addition, we have the price at which the auction closed. The goal is to predict whether or not the auction will be competitive.

Data Preprocessing. Create dummy variables for the categorical predictors. These include Category (18 categories), Currency (USD, GBP, Euro), EndDay (MondaySunday), and Duration (1, 3, 5, 7, or 10 days). Split the data into training and validation datasets using a 60% : 40% ratio. Make sure you are doing classification and note that there is a slight class imbalance you need to account for.

(a) Run AdaBoost (R function: **`adabag::boosting`**), XGBoost (R function: **`xgboost::xgboost`**), bagging (R function: **`adabag::bagging`**), random forest (R function: **`randomForest::randomForest`**) **with all predictors**. Fill in the table below with training accuracy, validation accuracy, and the most important variable (R function: **`adabag::importanceplot`**, **`xgboost::xgb.importance`**, and **`randomForest::varImp`**). **[12 pts]**

Table 2: Results with all predictors

	AdaBoost	XGBoost	bagging	random forest
Training accuracy				
Validation accuracy				
Most important variable				

(b) Run AdaBoost, XGBoost, bagging, random forest **with all predictors except "ClosePrice"**. Fill in the table below with training accuracy, validation accuracy, and the most important variable. **[10 pts]**

Table 3: Results with all predictors except "ClosePrice"

	AdaBoost	XGBoost	bagging	random forest
Training accuracy				
Validation accuracy				
Most important variable				

(c) Run AdaBoost, XGBoost, bagging, random forest **with all predictors except "OpenPrice"**. Fill in the table below with training accuracy, validation accuracy, and the most important variable. **[10 pts]**

Table 4: Results with all predictors except "OpenPrice"

	AdaBoost	XGBoost	bagging	random forest
Training accuracy				
Validation accuracy				
Most important variable				

(d) Run AdaBoost, XGBoost, bagging, random forest **with all predictors except "OpenPrice" and "ClosePrice"**. Fill in the table below with training accuracy, validation accuracy, and the most important variable. **[10 pts]**

Table 5: Results with all predictors except "OpenPrice" and "ClosePrice"

	AdaBoost	XGBoost	bagging	random forest
Training accuracy				
Validation accuracy				
Most important variable				

(e) Compute the proportion of the majority class in the test set. Comment on the result from (a), (b), (c), and (d). Which one you should use in reality and why? **[10 pts]**