ASSIGNMENT 7

Andrew-ID: parmenin

DATA INFERENCE AND APPLIED MACHINE LEARNING (18-785)

12/7/22

Niyomwungeri Parmenide ISHIMWE

I, the undersigned, have read the entire contents of the syllabus for course 18-785 (Data Inference and Applied Machine Learning) and agree with the terms and conditions of participating in this course, including adherence to CMU's AIV policy.

Signature: **Niyomwungeri Parmenide ISHIMWE**
Andrew ID: **parmenin**
Full Name: **Niyomwungeri Parmenide ISHIMWE**

The libraries used:

- import pandas as pd
- import numpy as np
- import matplotlib.pyplot as plt
- %matplotlib inline
- import yfinance as yf
- import seaborn as sns
- from sklearn import tree
- from sklearn.tree import DecisionTreeClassifier
- from sklearn.preprocessing import LabelEncoder
- from sklearn.neighbors import KNeighborsClassifier
- from sklearn.linear_model import LogisticRegression, LinearRegression
- from sklearn.metrics import r2_score, mean_squared_error, roc_curve, roc_auc_score
- from sklearn.model_selection import train_test_split
- from sklearn.decomposition import PCA
- from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor
- from sklearn.cluster import AgglomerativeClustering
- from scipy.spatial import distance
- from scipy.cluster import hierarchy

QUESTION 1: PCA

1.1) **The PCA or Principal Component Analysis** is a method for decreasing the dimensionality of datasets, improving interpretability, and minimizing information loss. It accomplishes this by producing fresh, uncorrelated variables that maximize variance one after another[1]. According to the course notes, PCA is described as a method that uses an orthogonal transformation to produce variables with linear uncorrelation called principal components (PCs), where PCs are ordered in terms of the amount of variance captured with the first PC explaining the maximum variance. Moreover, PCA also offers an eigenvalue spectrum, where each value represents the variance that each subsequent PC represents.

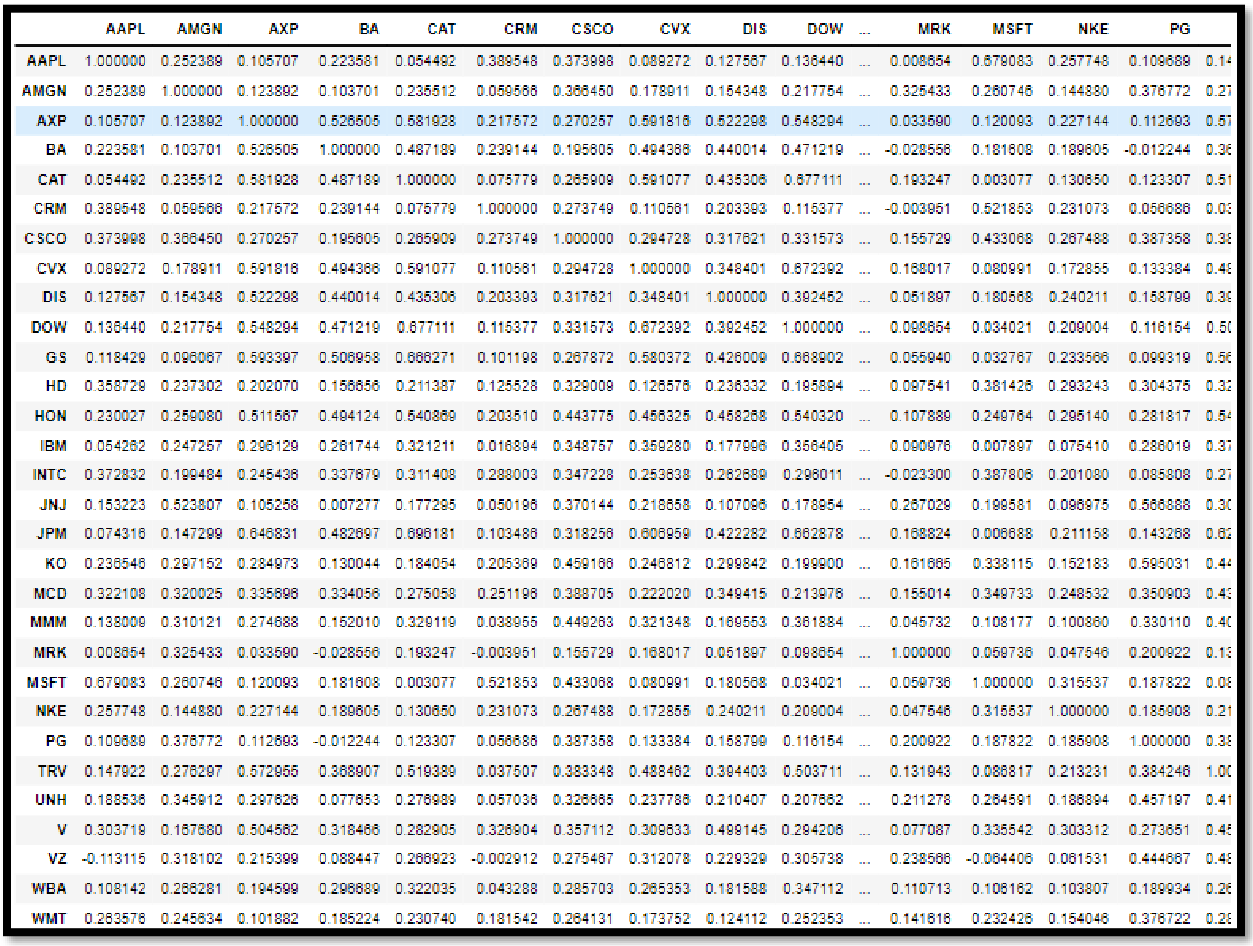
PCA applications include but are not limited to noise reduction where the contribution of higher components is deleted when options include a certain number of components or require a fraction of variance. This presupposes that noise corresponds to low levels of variance, whereas significant signals are connected to high levels of variance. In the real world, the number of dimensions in healthcare data is decreased using PCA. PCA can support image resizing. It might be utilized in finance to assess stock data and project returns. In high-dimensional datasets, PCA aids in pattern recognition[2].

It might be helpful to consider PCA while transforming a set of explanatory variables because it helps reduce noise in data variables to be used, helps to select effective features, and help to produce independent, uncorrelated data features.

1.2) The mathematical equation for PCA of X is given by: **Y = XV,** where **X** is an NxM data matrix consisting of N measurements and M variables, and **V** or the vectors of weights or loadings vm map each row vector xm of X to a new vector of principal component scores ym. The new variables in the columns of Y successively capture the maximum possible variance from the data matrix X

The covariance matrix **C** of the data matrix X is subjected to an eigenvalue decomposition and is given by: **C = VΣ^2V^**T, where **V** is orthogonal (**VV^T =I**) and **Σ^2** is a positive definite diagonal matrix. Columns of V, denoted by Vm, are the Mx1 eigenvectors of the MxM covariance matrix C. Eigenvalues **σm^2** on the diagonal of Σ^2 represent the variance associated with each eigenvector Vm.

1.3) It was required to load the Dow Jones Index (BeautifulSoup) dataset available on Yahoo Finance, and then determine the correlation matrix for the 30 stocks on Dow Jones Index, then construct bar graphs to show the weight of each stock for the first and second principal components using the correlation matrix for PCA.



	AAPL	AMGN	AXP	BA	CAT	CRM	CSCO	CVX	DIS	DOW	...	MRK	MSFT	NKE	PG	
AAPL	1.000000	0.252389	0.105707	0.223581	0.054492	0.389548	0.373998	0.089272	0.127567	0.136440	...	0.008654	0.679083	0.257748	0.109689	0.14
AMGN	0.252389	1.000000	0.123892	0.103701	0.235512	0.059566	0.366450	0.178911	0.154348	0.217754	...	0.325433	0.260746	0.144880	0.376772	0.27
AXP	0.105707	0.123892	1.000000	0.526505	0.581928	0.217572	0.270257	0.591816	0.522298	0.548294	...	0.033590	0.120093	0.227144	0.112693	0.57
BA	0.223581	0.103701	0.526505	1.000000	0.487189	0.239144	0.195605	0.494366	0.440014	0.471219	...	-0.028556	0.181608	0.189605	-0.012244	0.36
CAT	0.054492	0.235512	0.581928	0.487189	1.000000	0.075779	0.265909	0.591077	0.435306	0.677111	...	0.193247	0.003077	0.130650	0.123307	0.51
CRM	0.389548	0.059566	0.217572	0.239144	0.075779	1.000000	0.273749	0.110561	0.203393	0.115377	...	-0.003951	0.521853	0.231073	0.056686	0.03
CSCO	0.373998	0.366450	0.270257	0.195605	0.265909	0.273749	1.000000	0.294728	0.317621	0.331573	...	0.155729	0.433068	0.267488	0.387358	0.3
CVX	0.089272	0.178911	0.591816	0.494366	0.591077	0.110561	0.294728	1.000000	0.348401	0.672392	...	0.168017	0.080991	0.172855	0.133384	0.4
DIS	0.127567	0.154348	0.522298	0.440014	0.435306	0.203393	0.317621	0.348401	1.000000	0.392452	...	0.051897	0.180568	0.240211	0.158799	0.3
DOW	0.136440	0.217754	0.548294	0.471219	0.677111	0.115377	0.331573	0.672392	0.392452	1.000000	...	0.098654	0.034021	0.209004	0.116154	0.5
GS	0.118429	0.096067	0.593397	0.506958	0.666271	0.101198	0.267872	0.580372	0.426009	0.668902	...	0.055940	0.032767	0.233566	0.099319	0.5
HD	0.358729	0.237302	0.202070	0.156656	0.211387	0.125528	0.329009	0.126576	0.236332	0.195894	...	0.097541	0.381426	0.293243	0.304375	0.3
HON	0.230027	0.259080	0.511567	0.494124	0.540869	0.203510	0.443775	0.456325	0.458268	0.540320	...	0.107889	0.249764	0.295140	0.281817	0.5
IBM	0.054262	0.247257	0.296129	0.261744	0.321211	0.016894	0.348757	0.359280	0.177996	0.356405	...	0.090976	0.007897	0.075410	0.286019	0.37
INTC	0.372832	0.199484	0.245436	0.337679	0.311408	0.288003	0.347228	0.253638	0.262689	0.296011	...	-0.023300	0.387806	0.201080	0.085808	0.27
JNJ	0.153223	0.523807	0.105258	0.007277	0.177295	0.050196	0.370144	0.218658	0.107096	0.178954	...	0.267029	0.199581	0.096975	0.566888	0.3
JPM	0.074316	0.147299	0.646831	0.482697	0.696181	0.103486	0.318256	0.606959	0.422282	0.662878	...	0.168824	0.006688	0.211158	0.143268	0.6
KO	0.236546	0.297152	0.284973	0.130044	0.184054	0.205369	0.459166	0.246812	0.299842	0.199900	...	0.161665	0.338115	0.152183	0.595031	0.4
MCD	0.322108	0.320025	0.335698	0.334056	0.275058	0.251198	0.388705	0.222020	0.349415	0.213976	...	0.155014	0.349733	0.248532	0.350903	0.4
MMM	0.138009	0.310121	0.274688	0.152010	0.329119	0.038955	0.449263	0.321348	0.169553	0.381884	...	0.045732	0.108177	0.100860	0.330110	0.4
MRK	0.008654	0.325433	0.033590	-0.028556	0.193247	-0.003951	0.155729	0.168017	0.051897	0.098654	...	1.000000	0.059736	0.047546	0.200922	0.1
MSFT	0.679083	0.260746	0.120093	0.181608	0.003077	0.521853	0.433068	0.080991	0.180568	0.034021	...	0.059736	1.000000	0.315537	0.187822	0.0
NKE	0.257748	0.144880	0.227144	0.189605	0.130650	0.231073	0.267488	0.172855	0.240211	0.209004	...	0.047546	0.315537	1.000000	0.185908	0.21
PG	0.109689	0.376772	0.112693	-0.012244	0.123307	0.056686	0.387358	0.133384	0.158799	0.116154	...	0.200922	0.187822	0.185908	1.000000	0.3
TRV	0.147922	0.276297	0.572955	0.368907	0.519389	0.037507	0.383348	0.488462	0.394403	0.503711	...	0.131943	0.086817	0.213231	0.384246	1.0
UNH	0.188536	0.345912	0.297626	0.077653	0.276989	0.057036	0.326665	0.237786	0.210407	0.207662	...	0.211278	0.264591	0.188894	0.457197	0.41
V	0.303719	0.167680	0.504562	0.318466	0.282905	0.326904	0.357112	0.309633	0.499145	0.294206	...	0.077087	0.335542	0.303312	0.273651	0.4
VZ	-0.113115	0.318102	0.215399	0.088447	0.266923	-0.002912	0.275467	0.312078	0.229329	0.305738	...	0.238566	-0.064406	0.061531	0.444667	0.4
WBA	0.108142	0.266281	0.194599	0.296689	0.322035	0.043288	0.285703	0.265353	0.181588	0.347112	...	0.110713	0.106162	0.103807	0.189934	0.2
WMT	0.263576	0.245634	0.101882	0.185224	0.230740	0.181542	0.264131	0.173752	0.124112	0.252353	...	0.141616	0.232426	0.154046	0.376722	0.2

Figure 1: Snapshot of the correlation matrix for the 30 stocks

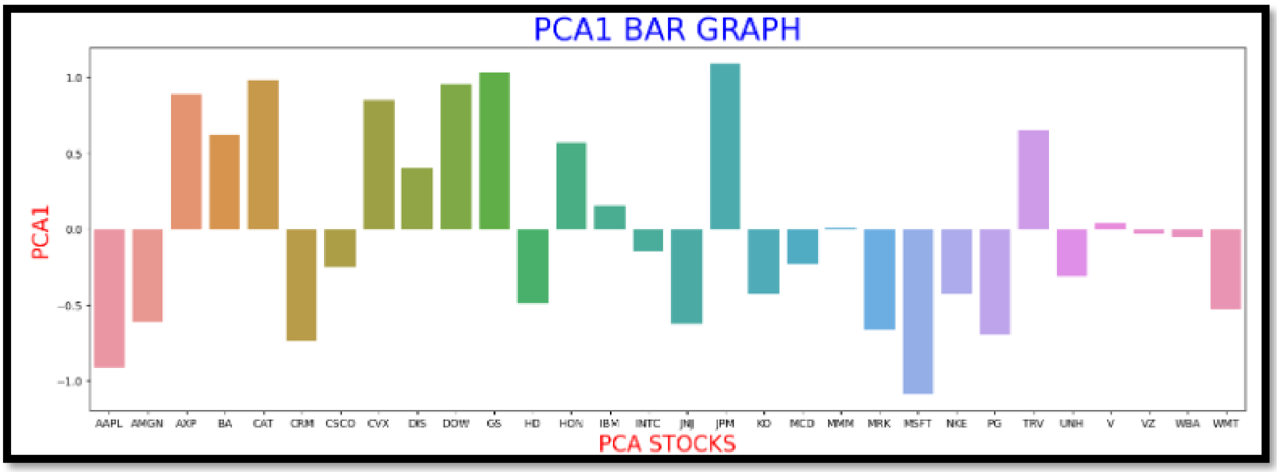


Figure 2: Bar graph to show the weight of each stock for the first principal components

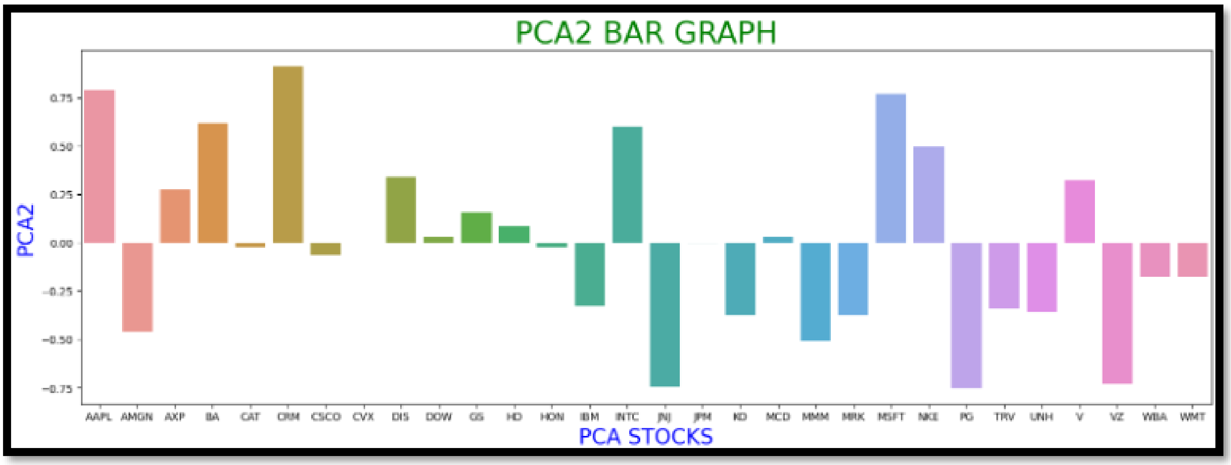


Figure 3: Bar graph to show the weight of each stock for the second principal components

It is found that the first and second principal components are like the market and have equal weight on each stock. This is because all weigh in both positive and negative directions and only a few have a big value stock. In addition, the PCA's first components are nearly like the original data or market. As the number of principal components increases, the similarity decrease

1.4) By determining the number of principal components required to explain 95% of the variance, it is found that the principal components required to explain 95% of the variance is 16 and the details are plotted in the following scree plot.

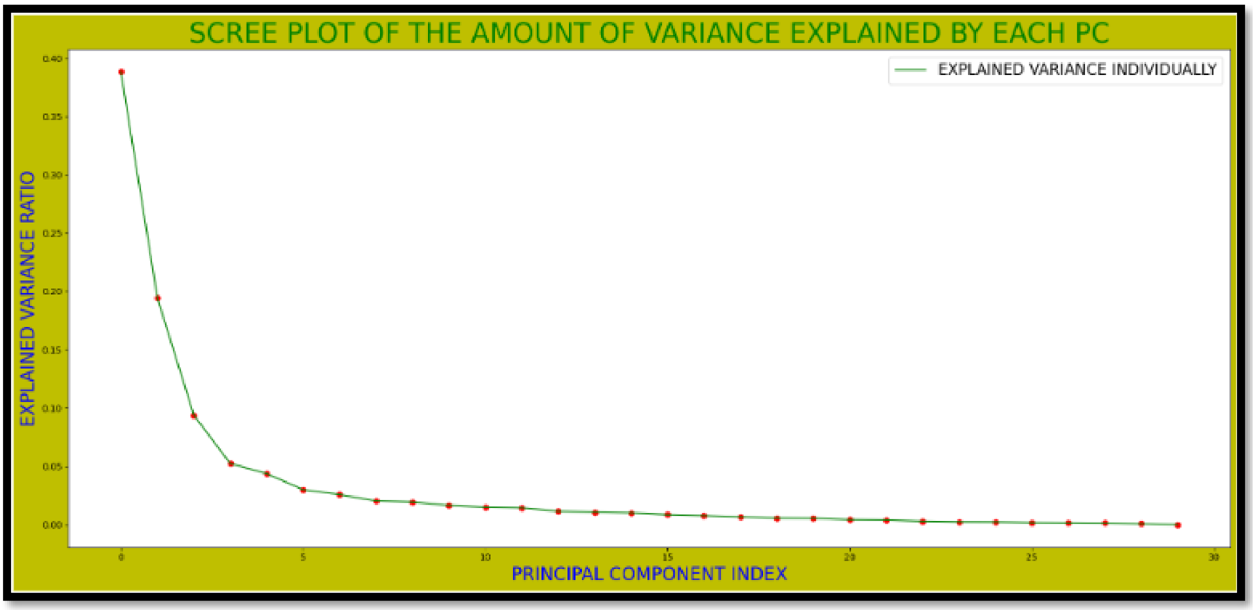
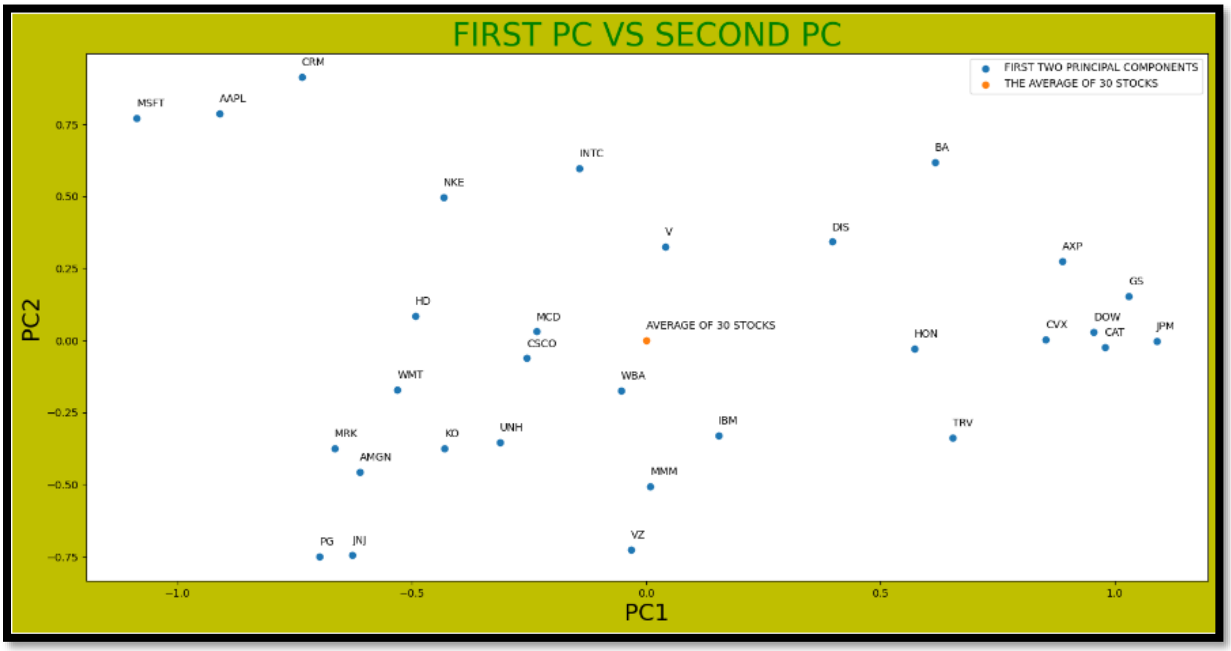
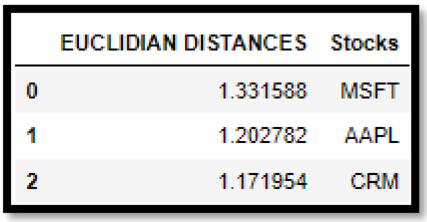


Figure 4: Scree plot of the amount of variance explained by each PC

1.5) Based on Euclidian distance, the three most distant stocks from the average for each principal component are shown below images is an American cloud-based software company Salesforce Inc (CRM), an American multinational technology company Apple Inc (AAPL), and Microsoft Corporation (MSFT) as well. Those stocks are most distant because they are wealthy and valuable companies in the world. almost all of them are technology companies that make a lot of money through the internet, software, and technological devices. Therefore, their value on the stock market should be very high compared to other companies.



<u>Figure 5</u>: First two principal components with the average



	EUCLIDIAN DISTANCES	Stocks
0	1.331588	MSFT
1	1.202782	AAPL
2	1.171954	CRM

<u>Figure 6 & 7</u>: Euclidian distances for 1st and 2nd principal component

QUESTION 2: DENDROGRAM

2.1) Dendrogram or a cluster tree is a tree-like structure created to group data over a variety of scales in a process called Hierarchical Clustering. This dendrogram structure is a multilevel hierarchy that helps to visualize the hierarchy of clusters rather than a single set of clusters, where clusters at one level are combined to form clusters at the next level. This strategy enables you to select the clustering scale or level that is best for your application.

The dendrogram operates by first calculating the separations between each pair of data points in the input, after which it iteratively combines the closest pair of data points, followed by the next closest pair of data points, and so on until all the data points have been merged. Following each merger, the distances between all data points are updated. Finally, a dendrogram is built using the distances at which the data points join.

In detail here are some steps:

- The first step is when data point A and data point B have the smallest distance, therefore, they are combined first.
- To build the following cluster, the following shortest distance between data points, let's say C and D, is calculated.
- After combining data, points A and B into a cluster, they are separated from another data point C by the same distance, which is equal to the smaller of the distances AC and BC.
- Then, this architecture is expanded to encompass clusters with more than two points.
- Any two clusters are separated by the shortest path between any two data points within those clusters.
- Progressively more points and clusters are combined in this way, with some points combined with existing clusters, until all the points are united into a single cluster.

To interpret a dendrogram, the height at which any two data points are connected is the key to understanding the results of the built dendrogram. For instance, the height of the link connecting A and B in the example is the smallest, which demonstrates their greatest similarity. Therefore, the data points can be compared by using the distance between them, and the shorter the distance between points, the higher the chance that they are assigned to the same hierarchical cluster[3].

2.2)

A dendrogram is a tree-like diagram that's frequently used to show how hierarchical clustering is performed. The procedures taken to create a dendrogram from a set of pairwise dissimilarity values are as follows:

- Start by seeing each data point as a distinct cluster.
- Determine how distinct each pair of clusters is from the others.
- Create a new cluster from the union of the two most related clusters.
- Repetition of step 2 with the newly created cluster being regarded as a single entity.
- Repeat steps 2-4 until there is just one cluster left.
- Create the dendrogram using the cluster merge sequence.

The algorithm used in step 2 will determine how dissimilarity is calculated and clusters are compared. For instance, some algorithms determine the dissimilarity of clusters using a linkage function, such as single-linkage or complete-linkage. Other techniques measure the similarity of data points using a distance metric, such as the Manhattan distance or the Euclidean distance.

The dendrogram can be created by plotting the cluster merges as a diagram resembling a tree after the cluster merge order has been established. The dendrogram's branches each represent a cluster, and the length of the branch indicates how distinct the merged clusters were from one another. The dendrogram can be used to comprehend the data's structure and spot groups of related data points.

2.3)

Hierarchical Clustering Algorithm

• Determine whether any two objects in the data set are similar or dissimilar to one another: The separation between items is calculated.

• Create a binary, hierarchical cluster tree out of the objects: The proximity of items to one another is determined by the linkage function using the distance information, and a dendrogram is created as a result.

• Identify the clustering points where the hierarchical tree should be cut: When using the cluster function, the hierarchical tree's bottom branches are pruned, and all the objects below each cut are given their cluster.

The following formula is used:

dij = (2(1 − ρij))1/2 where: dij is the distance and ρij is Pearson's correlation coefficient which measures the closeness of each pair of the stocks. This formula in python for instance as I used is **np.sqrt(2 * (1 - adjCloseCorr)).** The variables with distances that are almost 0 are the most correlated. Less correlation exists between values that are nearer to 2 and those that are nearer to zero.

2.4)

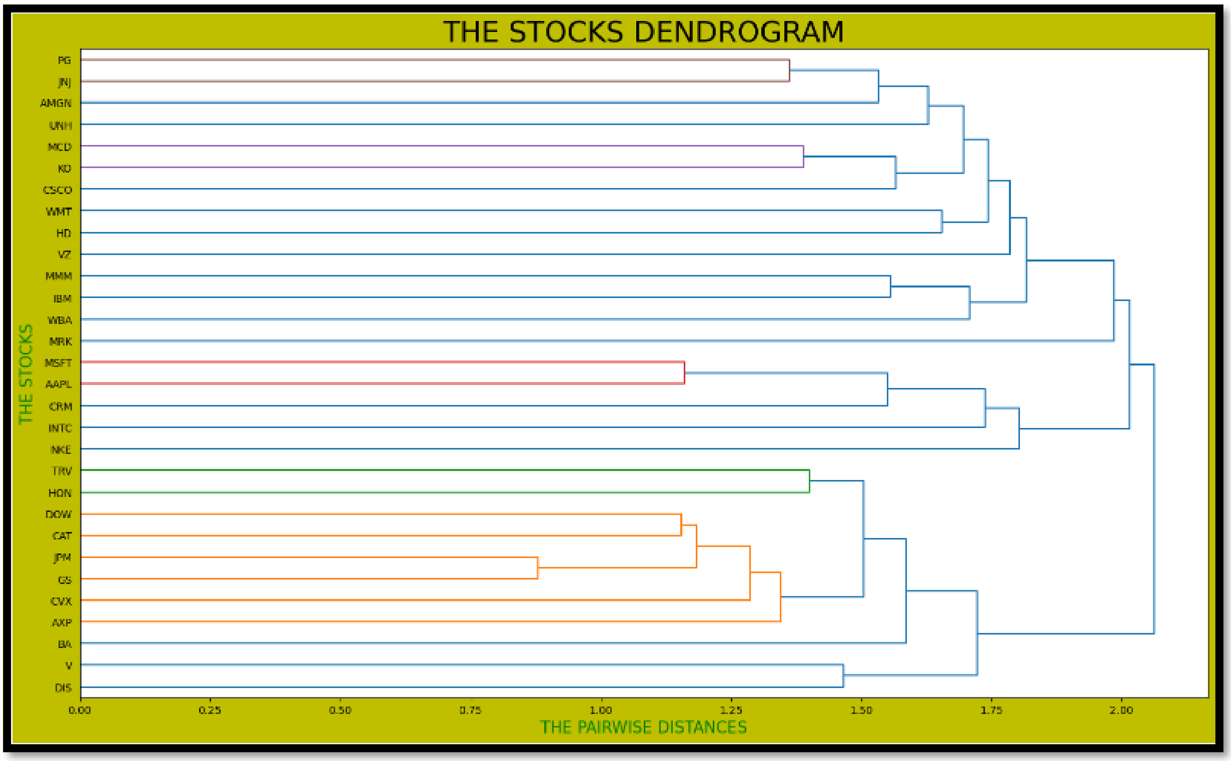


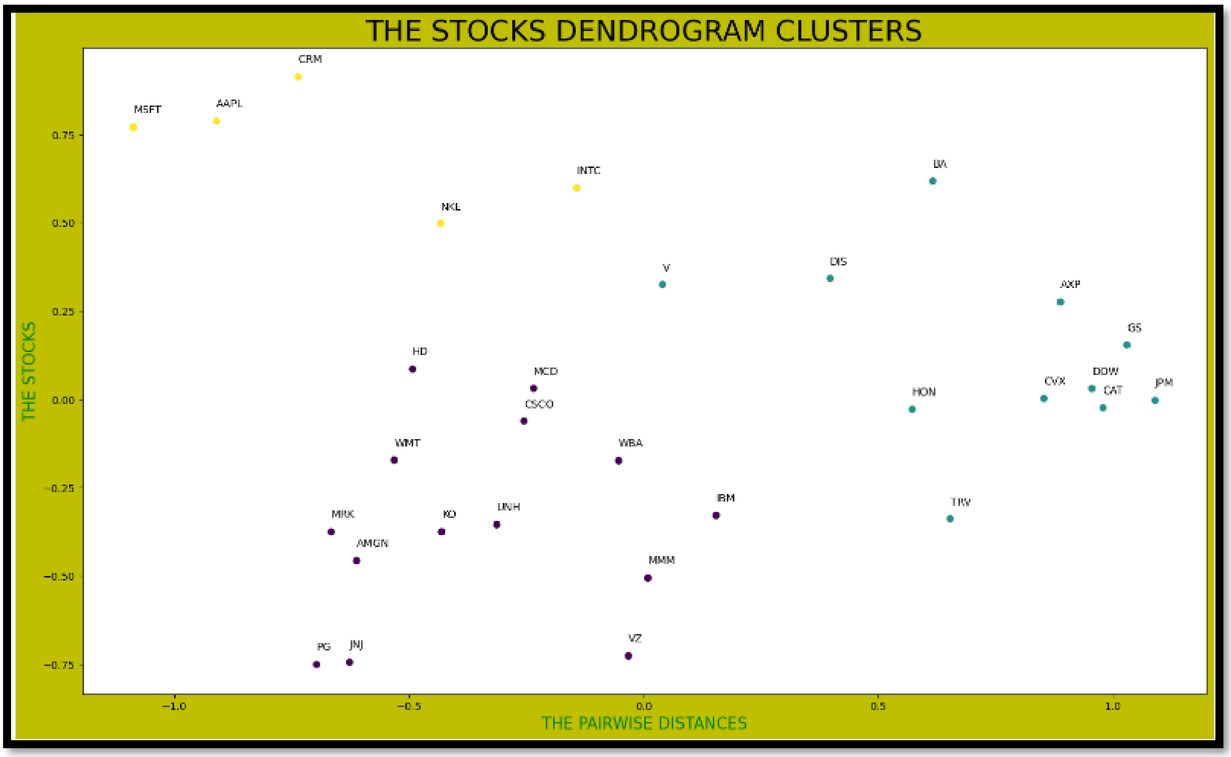
Figure 8: Horizontal Dendrogram for the stocks

2.5)

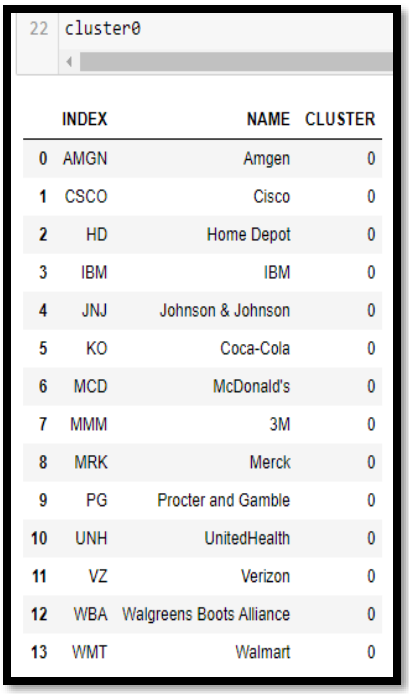
The first cluster contains one stock such as Amgen, Cisco, Home Depot, IBM, Johnson & Johnson, Coca-Cola, McDonald's, 3M, Merck, Procter and Gamble, UnitedHealth, Verizon, Walgreens Boots Alliance, and Walmart. In this cluster, most companies that operate in the health and pharmaceutical fields like Walgreens Boots Alliance, 3M, Amgen, Johnson & Johnson, and UnitedHealth.

- The second cluster contains American Express, Boeing, Caterpillar, Chevron, Disney, Dow, Goldman Sachs, Honeywell, JPMorgan Chase, Travelers, and Visa. Most of these companies are financial companies, like JPMorgan, Visa, American Express, and Goldman Sachs. There are also energy and engineering companies like Boeing, Caterpillar, and Chevron.

• The last cluster includes companies like Apple, Salesforce, Intel, Microsoft, and Nike. Most of these companies are technology related. They are more valuable on the stock market, and they make a lot of money nowadays as technology is advancing.

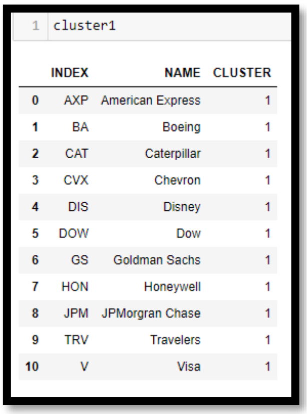


<u>Figure 9</u>: Three clusters of stock from the dendrogram



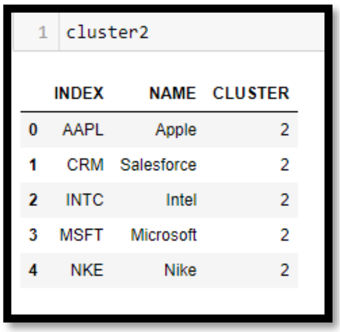
```
cluster0
```

	INDEX	NAME	CLUSTER
0	AMGN	Amgen	0
1	CSCO	Cisco	0
2	HD	Home Depot	0
3	IBM	IBM	0
4	JNJ	Johnson & Johnson	0
5	KO	Coca-Cola	0
6	MCD	McDonald's	0
7	MMM	3M	0
8	MRK	Merck	0
9	PG	Procter and Gamble	0
10	UNH	UnitedHealth	0
11	VZ	Verizon	0
12	WBA	Walgreens Boots Alliance	0
13	WMT	Walmart	0



```
cluster1
```

	INDEX	NAME	CLUSTER
0	AXP	American Express	1
1	BA	Boeing	1
2	CAT	Caterpillar	1
3	CVX	Chevron	1
4	DIS	Disney	1
5	DOW	Dow	1
6	GS	Goldman Sachs	1
7	HON	Honeywell	1
8	JPM	JPMorgran Chase	1
9	TRV	Travelers	1
10	V	Visa	1



```
cluster2
```

	INDEX	NAME	CLUSTER
0	AAPL	Apple	2
1	CRM	Salesforce	2
2	INTC	Intel	2
3	MSFT	Microsoft	2
4	NKE	Nike	2

Figure 10: Three clusters' lists

QUESTION 3: ENSEMBLES FOR CLASSIFICATION

3.1) The three sources of uncertainty are Observational uncertainty which deals with data, Parametrical uncertainty, and Structural uncertainty which both deal with model selection. To start, observational uncertainty arises when dealing with the data and the pattern in them. For instance, samples can be defined as observations to experiment on, and then the data gathered from observations can be called the result which probably may have some noise, and then the model can use those data to forecast the outcome. This mixture of signal and noise can bring inaccuracies in the data and observations. This influences both the measurements and the prediction outcomes' accuracy. Next, the parametrical uncertainty could be due to the domain's incomplete coverage. Regardless of how generalized the model is, some scenarios in the training dataset are covered while others are not. Because some of the factors were not considered, this will have an impact on the model's performance. Moreover, structural uncertainty, when working on a model, it is necessary to prepare it, which includes data preparation and model prediction interpretation. Model error can arise in modeling, resulting in erroneous predictions, such as quantity prediction in a regression problem. This could lead to inaccurate predictions and a shift in the model's expectations.

3.2) Model averaging is the process of developing several models and putting them together to get the desired result. Often, a collection of models outperforms any single model because the various errors of the model "average out. [4]". According to course notes, model averaging is a method created to assist in taking into consideration the inherent uncertainty in the model selection process as this uncertainty is frequently overlooked in classical statistical analysis.

Model averaging can be implemented in practice when generating predictions by considering the model uncertainty when concluding the parameters and forecasts by averaging results from numerous competing models. Model averaging has been effectively employed in numerous instances of statistical modeling, and it is effective at enhancing stability and prediction performance. It does this by decreasing the variance of an estimate using Bagging or Bootstrapping, decreasing bias using Boosting, and improving prediction using Stacking. Moreover, instead of learning a single, extensive complex model, model averaging helps to learn several smaller, simpler models and combine the results to arrive at the final judgment. In addition, individual model biases and variances are balanced by the model's overall strength. It offers a combined prediction whose accuracy is higher than that of the constituent models [5].

3.3) To reduce the effects of uncertainty and improve individual models, popular ensemble methods such as bagging, boosting, and stacking are employed to combine several models to make one very reliable model. By bagging, also called bootstrap aggregating, the aims are to improve the stability and accuracy of machine learning models to make correct decisions. This is done by reducing the variance of estimates and helping to avoid overfitting. Next, Boosting, which is an ensemble technique that learns from past predictor mistakes to improve future predictions, combines several weak small learners to form one powerful learner, hence greatly increasing the predictability of models. Boosting works by placing weak learners in sequential order so that they can learn from the subsequent learner to improve their predictive models. Although boosting has occasionally been proven to produce better accuracy than bagging, it also tends to overfit the training data. Lastly, stacking works by allowing a training algorithm to combine several other similar learning algorithm predictions[5].

3.4) It was asked to construct a random forest (RF) model by applying it to the Titanic dataset and hence describing the optimal number of trees finding process. To this, the titanic dataset is first brought, prepare for usage, and then split into training and testing datasets. After that, a range of list of trees is defined and used to estimate the best model using the

RandomForestClassifier from sklearn. ensemble to build, fit and evaluate the model using the score function and store all scores in a list. After evaluating the scores from the list, it is found that 11 trees are the optimal number with an accuracy score of 0.7900763358778626 and all the details are depicted below.

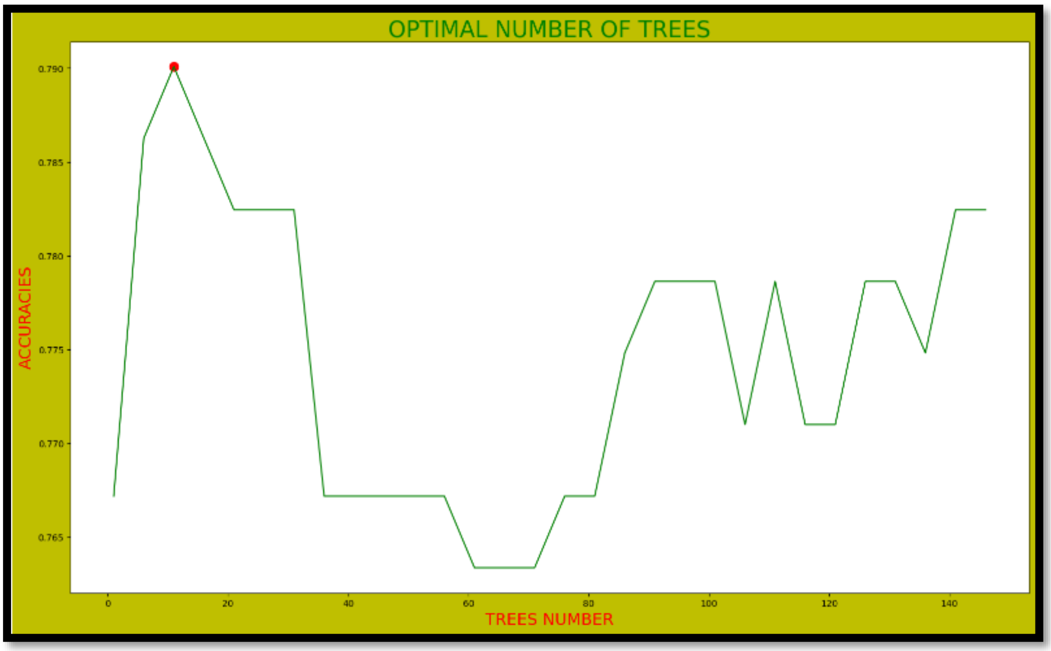


Figure 11: Optimal number of trees

3.5) It was also required to undertake a ROC analysis and to compare the RF performance to the previous logistic regression, classification tree, and KNN models. This was done by fitting these four models with the same titanic data and then predicting class probabilities of the x test data, and finally, the Area Under the Receiver Operating Characteristic Curve (ROC AUC) is computed from prediction scores using the roc_auc_score function from sklearn. metrics. The next step is to determine the false positive rate and the true positive rate using the roc_curve function for sklearn. metrics as well. It is obtained that the roc curve for the decision tree classifier is 0.7654, 0.8003 for the random forest, 0.7559 for the KNN classifier, and 0.8417 for the logistic classifier. From this, we can infer that the logistic model appears to be the best compared with other models because it has the greatest value for the area under the curve.

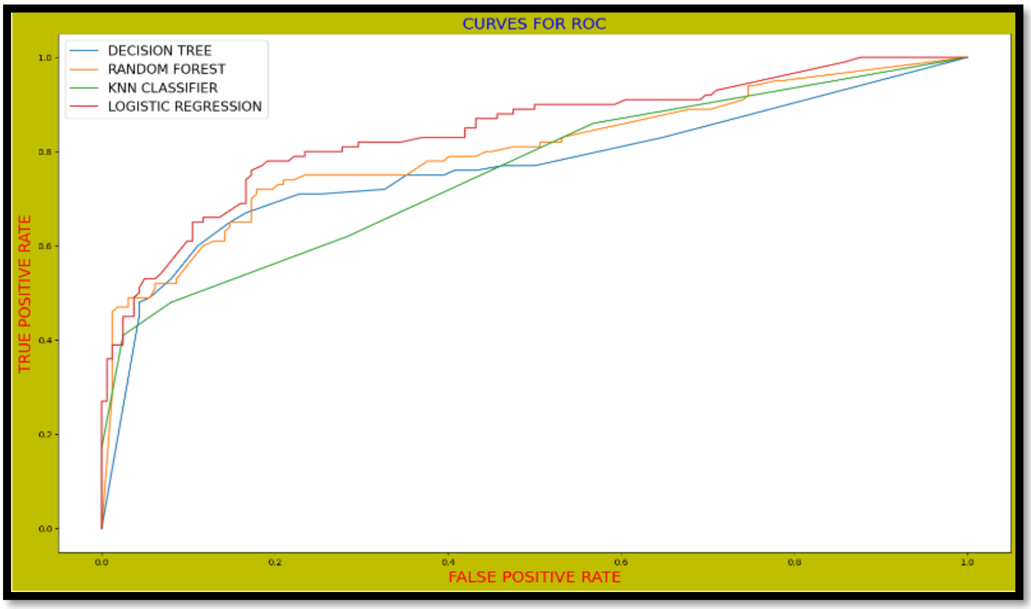


Figure 12: Logistic Regression, Decision Tree, Random Forest, and KNN models

QUESTION 4: ENSEMBLES FOR REGRESSION

4.1) Random Forest (RF) regression is a technique for supervised learning that makes use of ensemble learning and is used for classification and regression duties. This ensemble learning method combines predictions from various machine learning algorithms to provide predictions that are more accurate than those from a single model[6]. According to course notes, random forests (RF) were introduced in 2001 as a strategy that combines the random selection of features with the bagging of trees. To lower the variance, Random Forests calculates averages over several deep decision trees trained on various segments of the same training set. Although this method results in a slight bias increase and some interpretability loss, it significantly boosts the performance of the final model overall. In addition, multiple decision trees are built during the training phase of a random forest algorithm, and most of the trees' decisions are used to determine the final decision.

4.2) It was tasked to construct different random forest trees with a different numbers of leaves and to plot their errors for determining the optimal number of leaves. This was approached by reading the data, extracting columns to use, splitting the data into training and testing datasets, and then defining a range of the number of leaves as 150. The next step is to create and evaluate the models using a different number of leaves in the specified range using the mean squared error to find all the errors and then sort them to find the least error. After this whole process, it is found that 147 is the one with the least error and is annotated with a red on the graph.

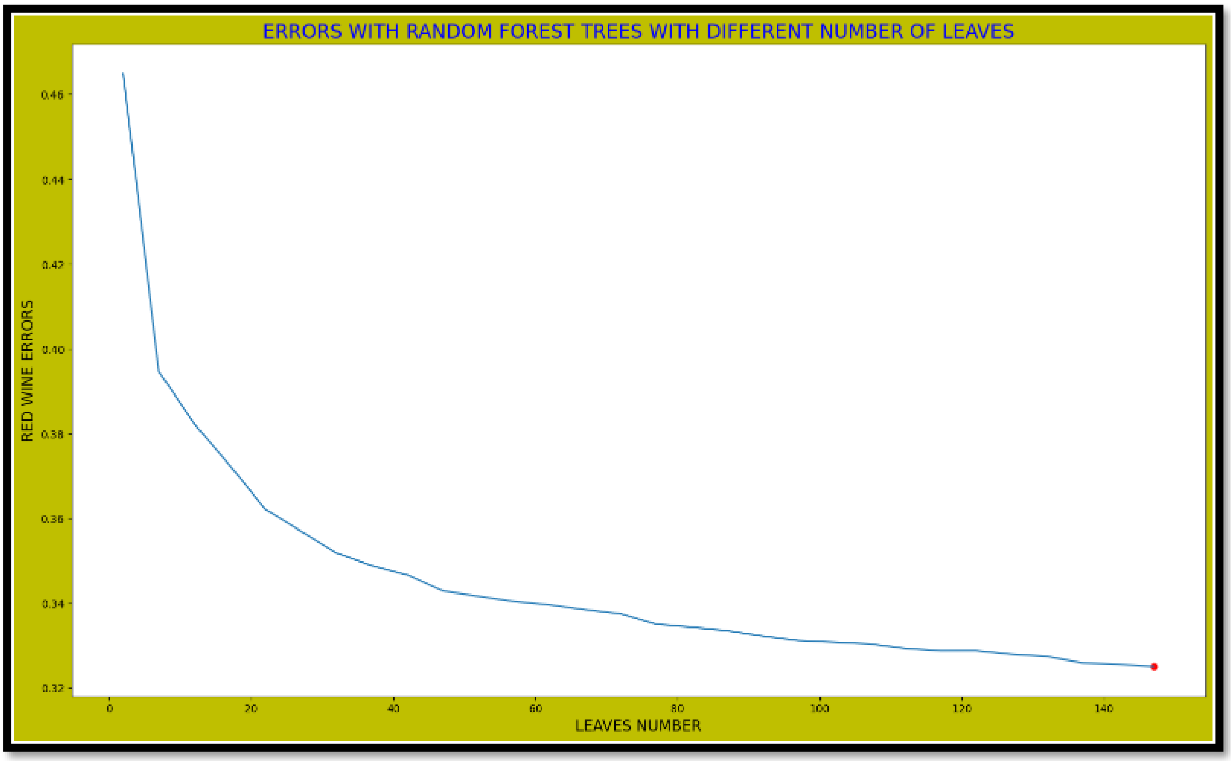
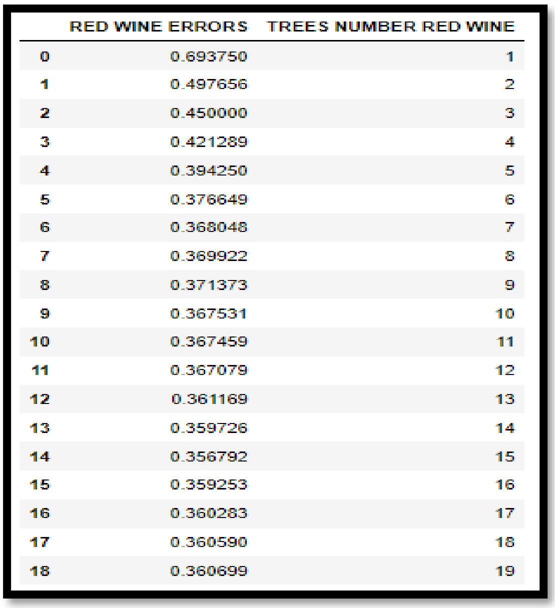


Figure 13: Random Forest errors with different numbers of leaves

4.3)

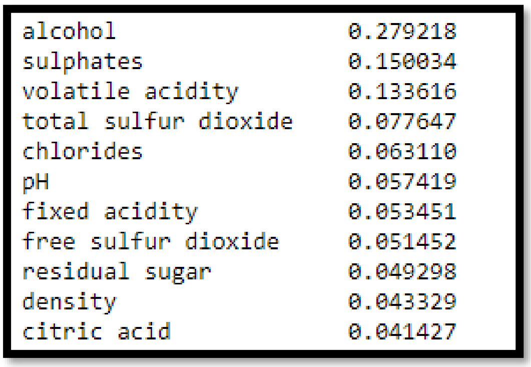
To obtain the optimal number of trees, first, we must calculate the minimum error of the whole random Forest model using the mean squared error (MSE) to evaluate all the models built by using each number of the trees list which ranges from 1 to 20 trees. The minimum error is 0.3567916666666667 which corresponds to 15 as the optimal number of trees as shown in the following table.



	RED WINE ERRORS	TREES NUMBER RED WINE
0	0.693750	1
1	0.497656	2
2	0.450000	3
3	0.421289	4
4	0.394250	5
5	0.376649	6
6	0.368048	7
7	0.369922	8
8	0.371373	9
9	0.367531	10
10	0.367459	11
11	0.367079	12
12	0.361169	13
13	0.359726	14
14	0.356792	15
15	0.359253	16
16	0.360283	17
17	0.360590	18
18	0.360699	19

Figure 14: Random Forest errors showing 15 as the optimal number

4.4) It was asked to provide a bar graph that shows the importance of each feature and to compare it with the results from Assignment 6 using correlation and LassoCV. By fitting the RandomForestRegressor from sklearn. The ensemble found that the importance of the features is the following: **Alcohol** is the most important feature, the second is **sulfates**, the next is **volatile acidity**, **total sulfur dioxide**, **chlorides,** and **PH** as the top six features. On the other hand, LassoCV in assignment 6 has identified the following features as important **'fixed acidity', 'volatile acidity, 'residual sugar', 'free sulfur dioxide', 'total sulfur dioxide, and 'alcohol'.** For both models, alcohol has been identified as the best model with an estimate of 0.355502 on the LASSOCV model and 0.279218 estimates on the Random Forest regressor model in assignment 7.



```
alcohol                 0.279218
sulphates               0.150034
volatile acidity        0.133616
total sulfur dioxide    0.077647
chlorides               0.063110
pH                      0.057419
fixed acidity           0.053451
free sulfur dioxide     0.051452
residual sugar          0.049298
density                 0.043329
citric acid             0.041427
```

Figure 15: Important features

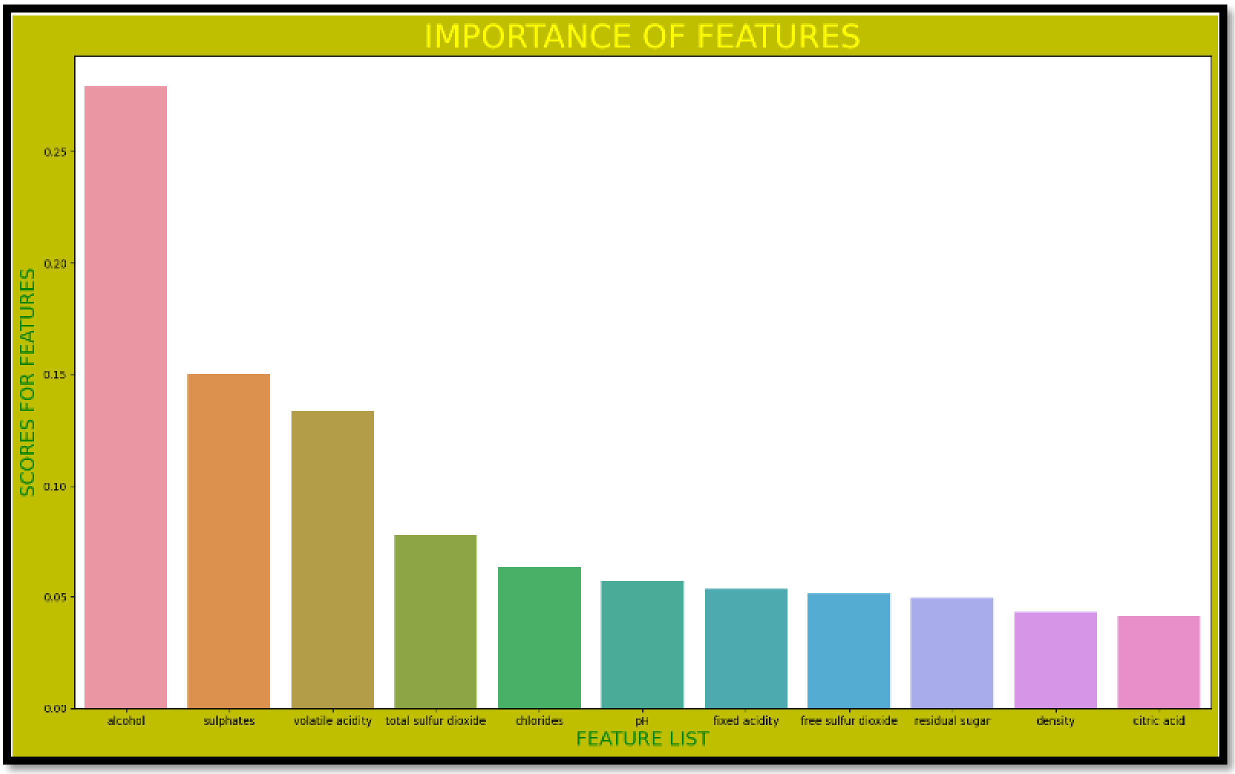


Figure 16: Horizontal Dendrogram for the stocks

4.5) It was asked to find the performance of the RF model and compare it with the linear regression and KNN models. So, by calculating the performance of the RF model using the mean squared error, it is found that the random forest model is the best model to use with the red wine dataset since it has fewer errors compared to the other two. Its mean squared error (MSE) is found to be 0.31842577166921193. Next is the linear regression model with a 0.38447119782012495 error, and last is the KNN model with an error of 0.671875.

REFERENCES

[1] ‘Principal component analysis: a review and recent developments | Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences’. https://royalsocietypublishing.org/doi/10.1098/rsta.2015.0202 (accessed Nov. 24, 2022).

[2] ‘Principal Component Analysis in Machine Learning | Simplilearn’. https://www.simplilearn.com/tutorials/machine-learning-tutorial/principal-component-analysis (accessed Nov. 24, 2022).

[3] T. Bock, ‘What is a Dendrogram?’, *Displayr*, Mar. 21, 2018. https://www.displayr.com/what-is-dendrogram/ (accessed Nov. 29, 2022).

[4] ‘Ensemble averaging (machine learning)’, *Wikipedia*. Nov. 30, 2021. Accessed: Nov. 29, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Ensemble_averaging_(machine_learning)&oldid=1057851906

[5] ‘Ensemble Methods’, *Corporate Finance Institute*. https://corporatefinanceinstitute.com/resources/data-science/ensemble-methods/ (accessed Nov. 29, 2022).

[6] chaya, ‘Random Forest Regression’, *Medium*, Apr. 14, 2022. https://levelup.gitconnected.com/random-forest-regression-209c0f354c84 (accessed Nov. 29, 2022).