DATA INFERENCE AND APPLIED MACHINE LEARNING (18_785)

CARNEGIE MELLON UNIVERSITY AFRICA

ASSIGNMENT 7

Janvier Muvunyi | jmuvunyi | December 02, 2019

Question 1.

1.1 Give a qualitative description of Principal Component Analysis (PCA) and its applications in machine learning. Why might it be useful to consider PCA to transform a set of explanatory variables?

Answer

Principal Component Analysis (PCA) performs an orthogonal transformation to provide linearly uncorrelated variables called principal components (PCs). The PCs are ordered in terms of the amount of variance captured with the first PC explaining the maximum variance. PCA also provides an eigenvalue spectrum where each value indicates the amount of variance represented by successive PCs. PCA can be used for noise reduction where the contribution of higher components is deleted. Options are to include certain number of components or require fraction of variance. This assumes that important signals are related to high levels of variance and that noise corresponds to low levels of variance. PCA is known for dimensionality reduction where it performs a linear mapping of dataset to a lower dimensional space so that the variance of the data in the new dimension representation is maximum.

In addition, principal component analysis (PCA) is defined as a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities each of which takes on various numerical values) into a set of values of linearly uncorrelated variables called principal components. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. The resulting vectors (each being a linear combination of the variables and containing *n* observations) are an uncorrelated orthogonal basis set. PCA is sensitive to the relative scaling of the original variables. [1]

Applications of PCA

PCA is applied in Quantitative finance, neuroscience, computer vision, etc.

Principal Component Analysis is also applied in:

Compression (Image compression): when we are dealing with big data where we have more data that exceed the memory, we need to use PCA to decompose the features into lower dimension. PCA reduces the memory needed for storing data. furthermore, it speeds up learning algorithm. Using PCA we can convert 3D to 2D and 2D to 1D.
Visualization: PCA is also used for visualizing higher dimensions. “Principal components or eigen vectors” in which the first eigen vector is the first eigen vector is used. Thus, PCA allows us to view the distribution of data along principal components.
PCA reduction is mostly done to either 2 or 3 dimensions.
Feature selection: PCA is also used for feature selection, what is done here is investigating which feature or explanatory variables have dominant mode of variations. [2]

1.2 Write down the mathematical equations for PCA explaining how one transforms the raw input data matrix X into a new set of variables. Give an interpretation of each matrix.

Answer
Construct an NxM data matrix **X** corresponding to N measurements and M variables
The covariance matrix **C** of the data matrix **X** is subjected to an eigenvalue decomposition:
$\mathbf{C = V\Sigma^2V^T}$
Where **V** is orthogonal ($\mathbf{VV^T=I}$) and $\mathbf{\Sigma}^2$ is a positive definite diagonal matrix
Columns of **V**, denoted by **v**m, are the Mx1 eigenvectors of the MxM covariance matrix **C**
Eigenvalues $\boldsymbol{\sigma}_{\mathbf{m}}^{\mathbf{2}}$ on diagonal of $\mathbf{\Sigma}^2$ represent the variance associated with each eigenvector $\mathbf{v_m}$
The PCA of **X** can therefore be given as **Y** = **XV.** Where the vectors of weights or *loadings* $\mathbf{v_m}$ map each row vector $\mathbf{x_m}$ of **X** to a new vector of principal component *scores* $\mathbf{y_m}$. The new variables in the columns of **Y** successively capture the maximum possible variance from the data matrix **X.**

Interpretation summary of each matrix

X is a NxM data matrix where N is measurements and M are variables.
V is the Mx1 eigenvectors of the MxM covariance matrix **C**.
Y contains the new variables.

1.3 Use at least one year of daily returns to calculate the correlation matrix for the 30 stocks that are constituents the Dow Jones Index. Matlab’s “BlueChipStockMoments” can be used to calculate the correlation matrix. Use this correlation matrix for PCA and construct bar graphs to show the weight of each stock for the first and second principal components. Is the first or second principal component similar to the market (equal weight on each stock) and discuss why?

Answer:

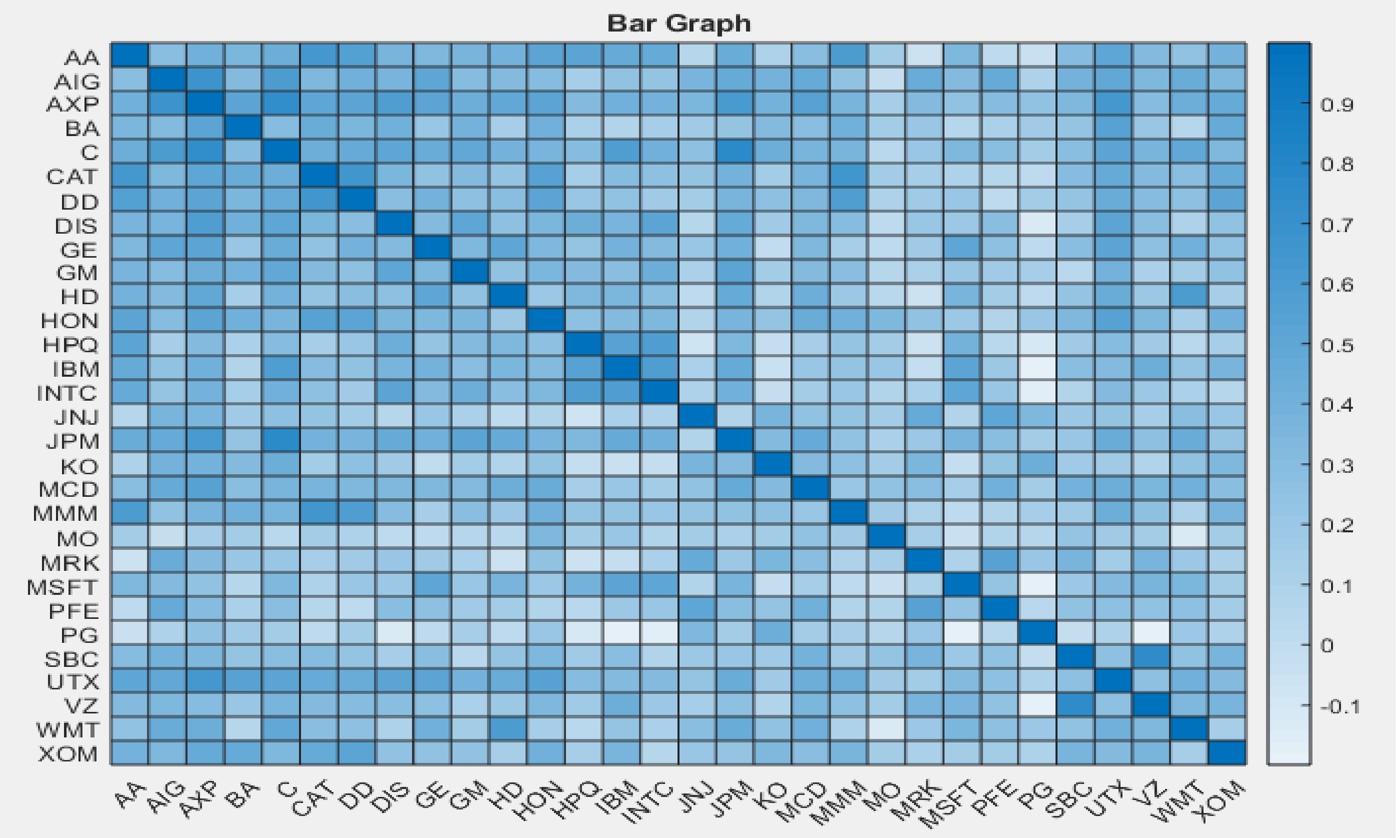


Figure 1. Heatmap showing correlation matrix of 30 stocks

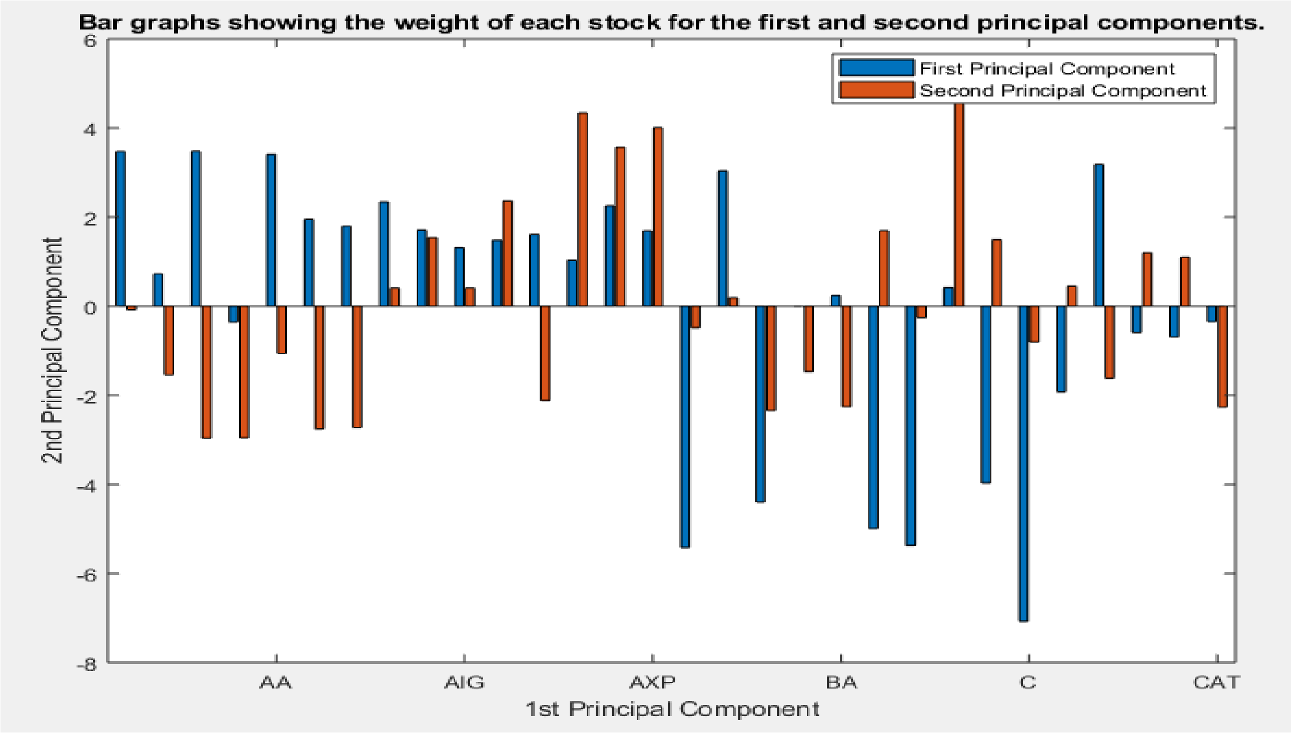


Figure 2. Bar graph to show the weight of each stock for the first and second principal components

Is the first or second principal component similar to the market (equal weight on each stock) and discuss why?

Neither first nor second principal component have equal weight on each stock. The graph reveals us that each weight of stock on first principal component is not equal to the stock's weight of the second principal component.

1.4 Calculate the amount of variance explained by each principal component and make a 'Scree' plot. How many principal components are required to explain 95% of the variance?

Answer

Below shown is the amount of variance explained by each principal component:

First component Variance: **8.9599**
Second component Variance: **5.1855**

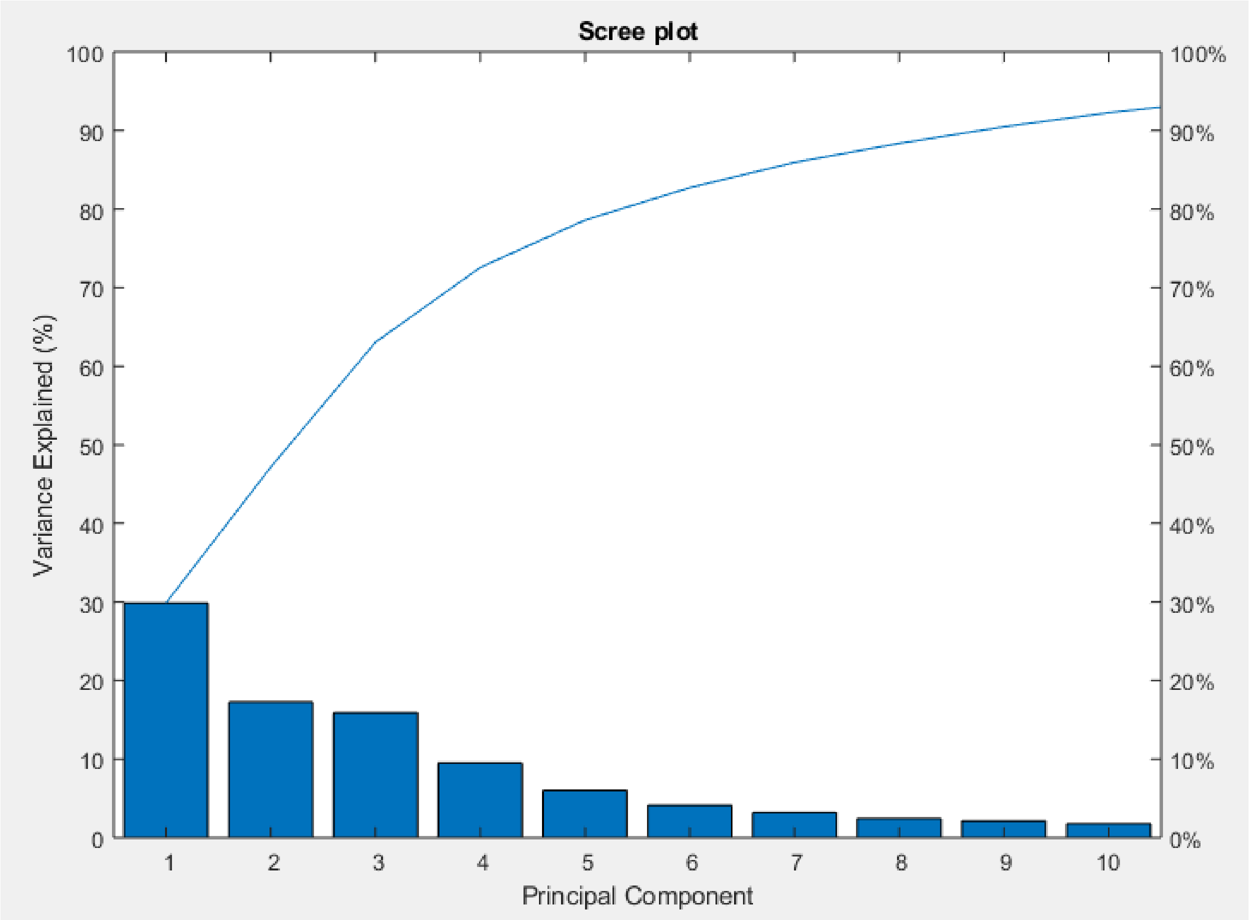
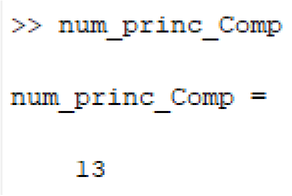


Figure 3. Scree plot for variance and PCs

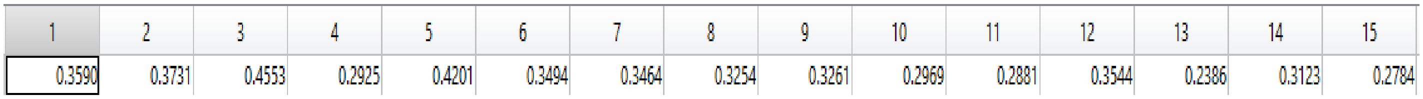


```
>> num_princ_Comp

num_princ_Comp =

    13
```

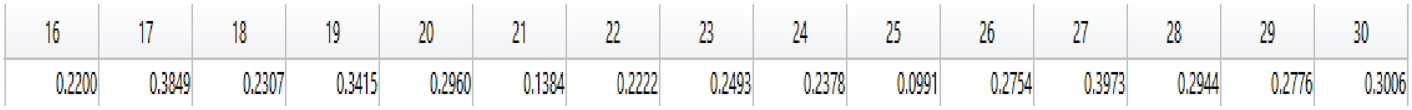
As it is computed, the number of principal components required to explain **95%** of the variance is **13.**

1.5 Investigating the scatter plot of the first two principal components and calculating the average of all 30 stocks. Based on Euclidean distances away from this average, identify the three most distant stocks. Can you explain why these stocks are unusual?

Answer:



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.3590	0.3731	0.4553	0.2925	0.4201	0.3494	0.3464	0.3254	0.3261	0.2969	0.2881	0.3544	0.2386	0.3123	0.2784



16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0.2200	0.3849	0.2307	0.3415	0.2960	0.1384	0.2222	0.2493	0.2378	0.0991	0.2754	0.3973	0.2944	0.2776	0.3006

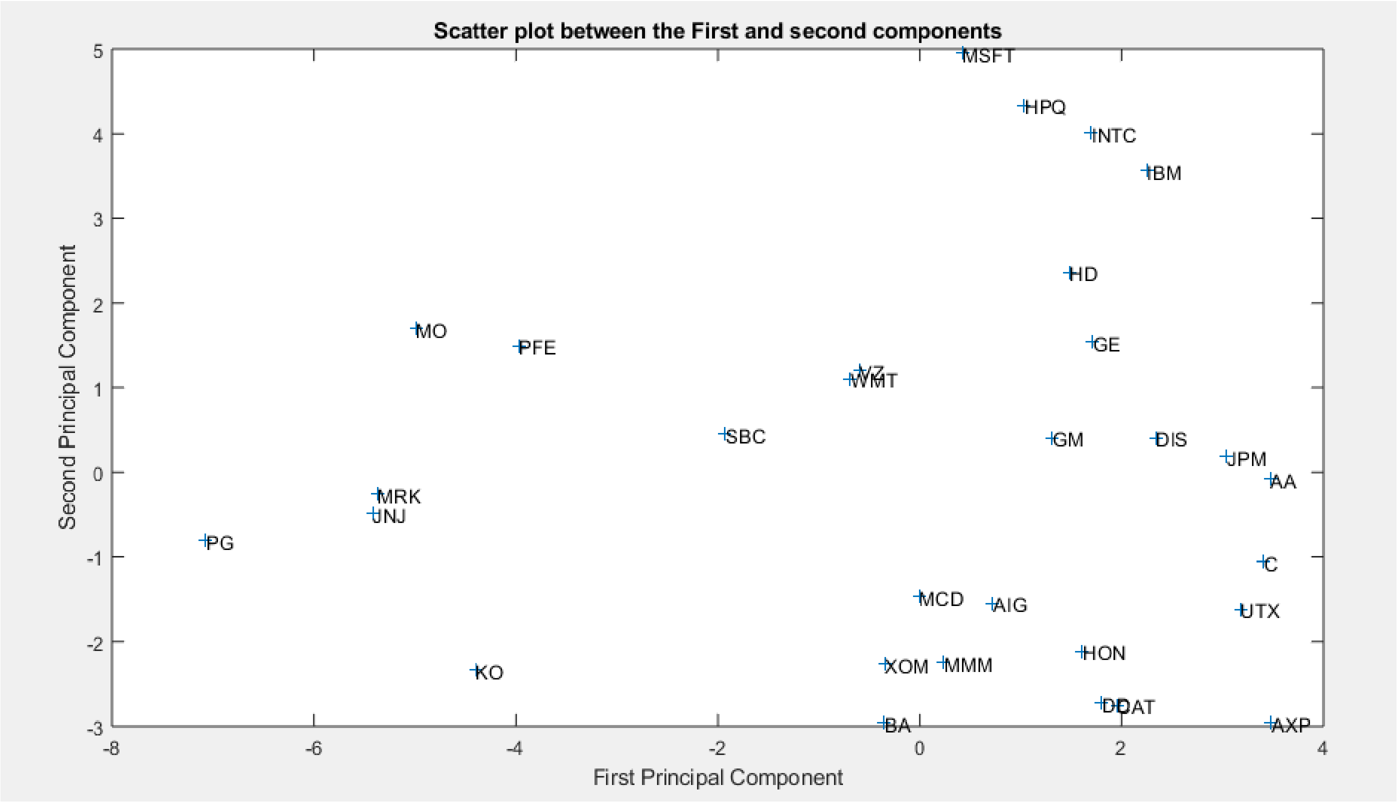
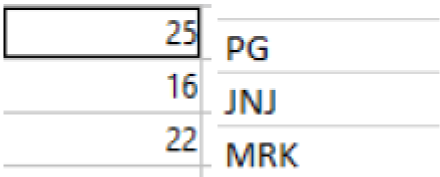


Figure 4. Scatter plot between the first and second components

Three most distant stocks are:



25	PG
16	JNJ
22	MRK

These stocks are unusual because they are uncorrelated.

QUESTION2. Dendrogram

2.1. Describe the components of a dendrogram, how it is constructed and how it is interpreted.

Answer

The components of a dendrogram:

Clusters, **links** and **nodes**

Dendrogram is the result of the hierarchical methods, it represents the nested grouping of objects and similarity levels at which groupings change.
A dendrogram can be considered as a graphical interface of linkages. In Figure below, the component of the dendrogram is shown. Dendrograms consists of many U-shaped lines connecting objects in the hierarchical tree and defines **links**. The horizontal axis in the dendrogram represents the indices of objects in the data set and the vertical axis shows the distances between the grouped objects (clusters). These distances can be interpreted as height of the links, which connects clusters to each other. Each node in the diagram represents one object if the total number of objects does not exceed 30, otherwise each node may represent more nodes. [3]

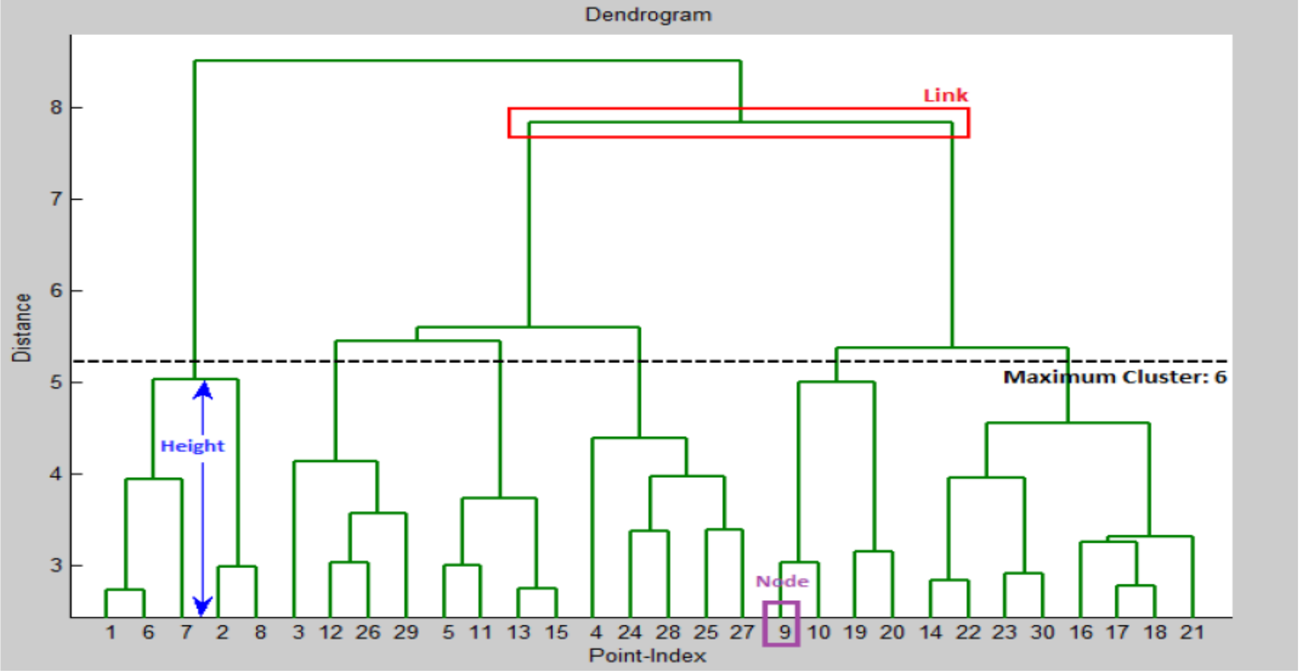


Figure 5. The Dendrogram and its components

How a dendrogram is constructed:

Find the similarity or dissimilarity between every pair of objects in the data set.

In this step, you calculate the *distance* between objects using the **pdist** function.
The **pdist** function supports many different ways to compute this measurement.

Group the objects into a binary, hierarchical cluster tree.

In this step, we link pairs of objects that are in close proximity using the linkage function.
The linkage function uses the distance information generated in step 1 to determine the proximity of objects to each other. As objects are paired into binary clusters, the newly formed clusters are grouped into larger clusters until a hierarchical tree is formed.

Determine where to cut the hierarchical tree into clusters.

In this step, we use the cluster function to prune branches off the bottom of the hierarchical tree, and assign all the objects below each cut to a single cluster. This creates a partition of the data.

The cluster function can create these clusters by detecting natural groupings in the hierarchical tree or by cutting off the hierarchical tree at an arbitrary point. [4]

Some function used:

Dendrogram(tree,Name,Value) uses additional options specified by one or more name-value pair arguments.

Dendrogram(tree,P) generates a dendrogram plot with no more than P leaf nodes. If there are more than P data points in the original data set, then dendrogram collapses the lower branches of the tree. As a result, some leaves in the plot correspond to more than one data point.

Dendrogram (tree,P,Name,Value) uses additional options specified by one or more name value pair arguments.

H = dendrogram (___) generates a dendrogram plot and returns a vector of line handles. You can use any of the input arguments from the previous syntaxes.

[H,T,outperm] = dendrogram(___) also returns a vector containing the leaf node number for each object in the original data set, T, and a vector giving the order of the node labels of the leaves as shown in the dendrogram, outperm.

It is useful to return T when the number of leaf nodes, P, is less than the total number of data points, so that some leaf nodes in the display correspond to multiple data points.

The order of the node labels given in outperm is from left to right for a horizontal dendrogram, and from bottom to top for a vertical dendrogram. [5]

Interpretation of dendrogram

To best interpret the dendrogram,

We start from the bottom on each node, to view the which datapoints are alike, we also find the heights of links, when two datapoints have same height, it means they have a similarity. We continue going up the dendrogram, if we find two datapoints which have different height but same links, that tells us that they might be in the same cluster which means that datapoints in that cluster also have a similarity.
Some datapoints might also have dissimilarity with others, which can be seen from a dendrogram.

2.2. Given a collection of pairwise dissimilarity values, describe the steps involved in constructing a dendrogram.

Answer

Steps involved in constructing a dendrogram

Given a collection of pairwise dissimilarity values:
This means that the distance or dissimilarity matrix is given.
The next thing is to determine how objects in the dataset are going to be grouped into clusters. This is done by linking the pairs of datapoints that are close together (according to the given pairwise dissimilarity value). This forms a cluster of two objects.

Then, the newly formed clusters are linked to each other and to other datapoint to create bigger clusters, this is done until all the datapoint from original dataset are linked together to form a hierarchical tree.
We can use "**linkage**" function in MATLAB to achieve the previous stated step.
The following figure shows how the datapoints are forming clusters, depending on the distance between each and every one. [6]

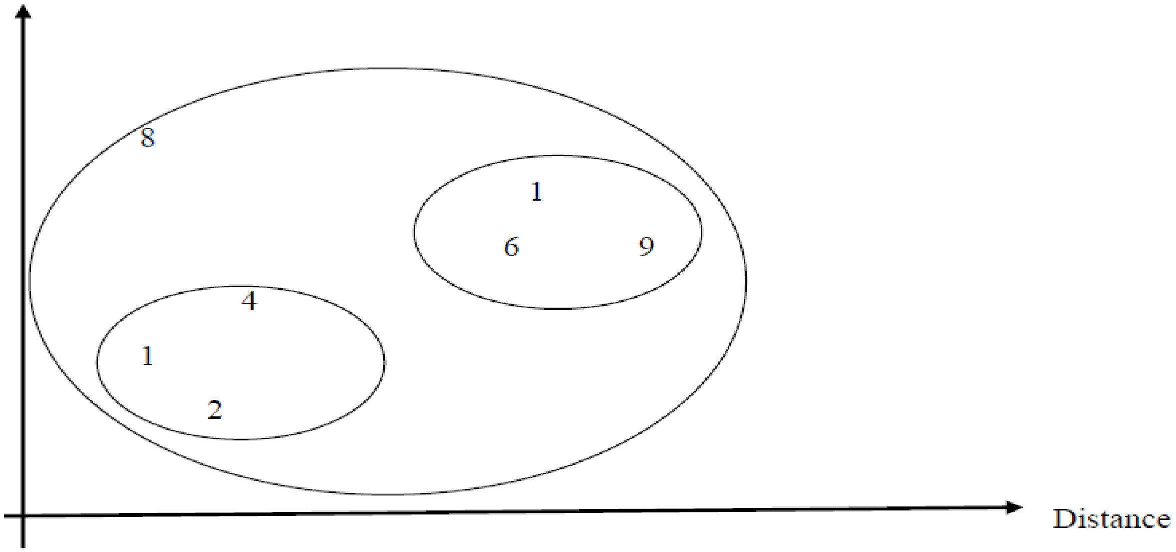
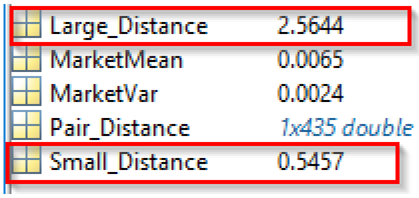


Figure 6. How the datapoints are forming clusters, depending on the distance between each and every one

2.3. Use the correlation matrix from question (1.3) above to provide pairwise distances between the 30 stocks. Give the formula for this rescaled distance and provide an interpretation of small and large distances.

Rescaled distance formula

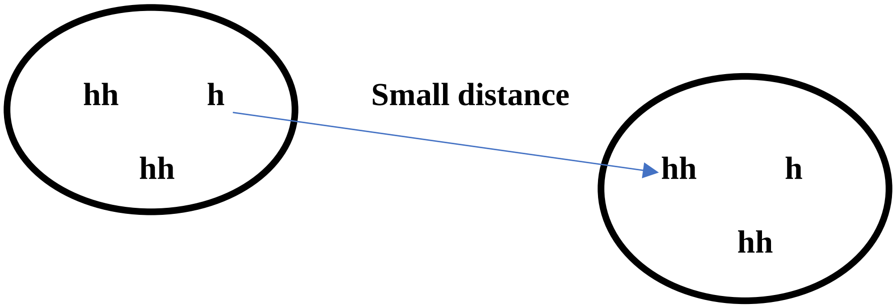
$$d_{st}^2 = (x_s - x_t)(x_s - x_t)'$$



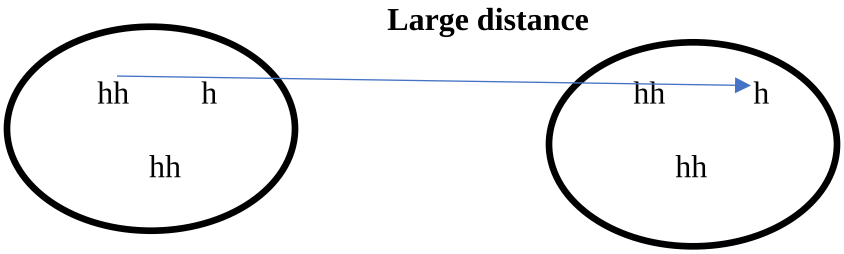
Large_Distance	2.5644
MarketMean	0.0065
MarketVar	0.0024
Pair_Distance	1x435 double
Small_Distance	0.5457

Largest distance = **2.5644**
Smallest distance = **0.5457**

Small distance is the smallest distance from 2 nearest point from 2 clusters.



Large distance: is the furthest distance from 2 farthest distance within two clusters



2.4. Constructing a horizontal dendrogram using the average linkage approach, carefully labeling the graphic with the names of the 30 stocks.

Answer

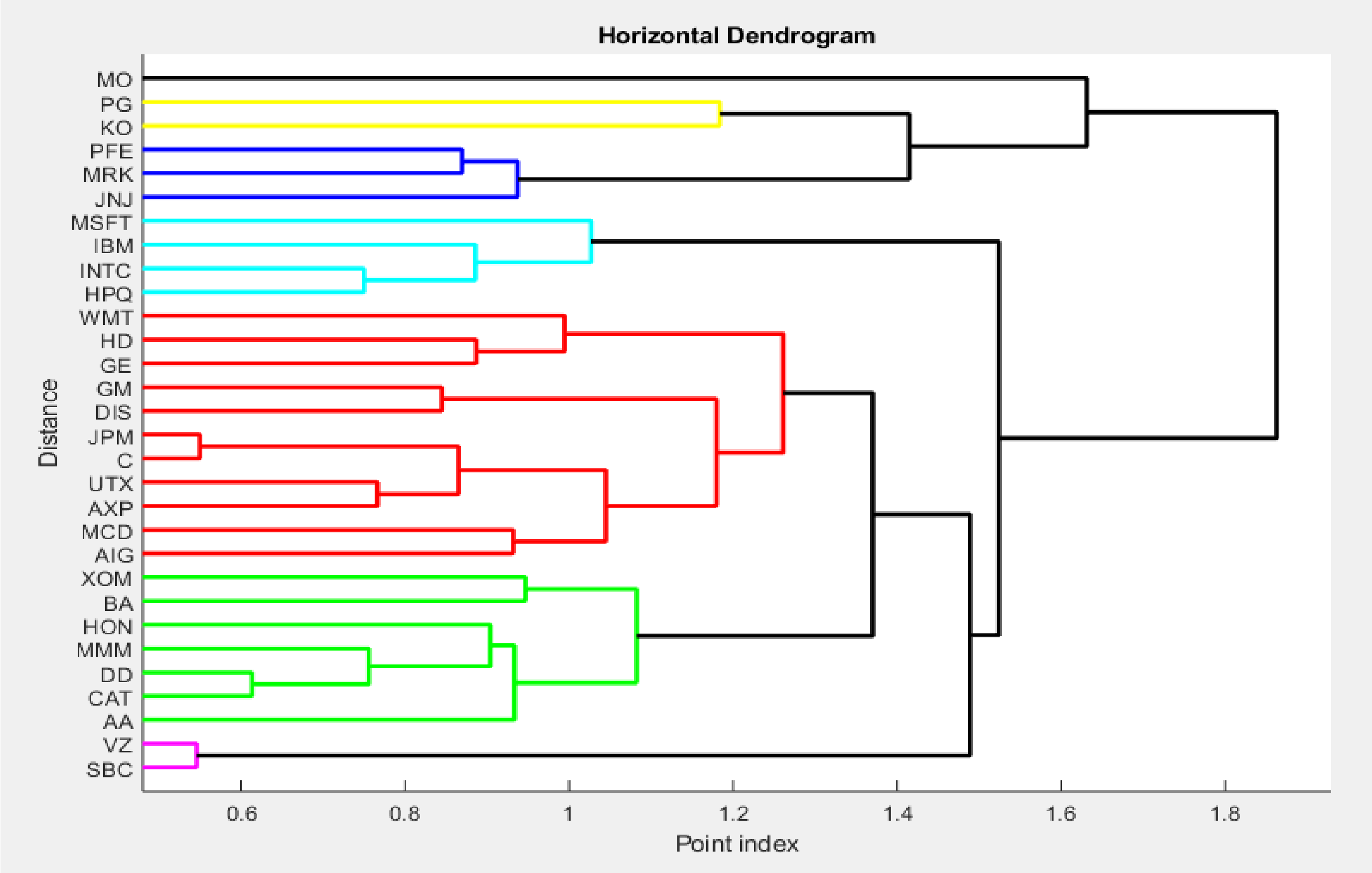


Figure 7. Horizontal dendrogram

2.5. Use the dendrogram to provide a few clusters of stocks and list the stocks that are members of each cluster. Can you provide a description of each cluster and relate it to industrial sectors such as Financials, Energy etc.?

Some clusters are firstly selected, and it is shown on the below graph.

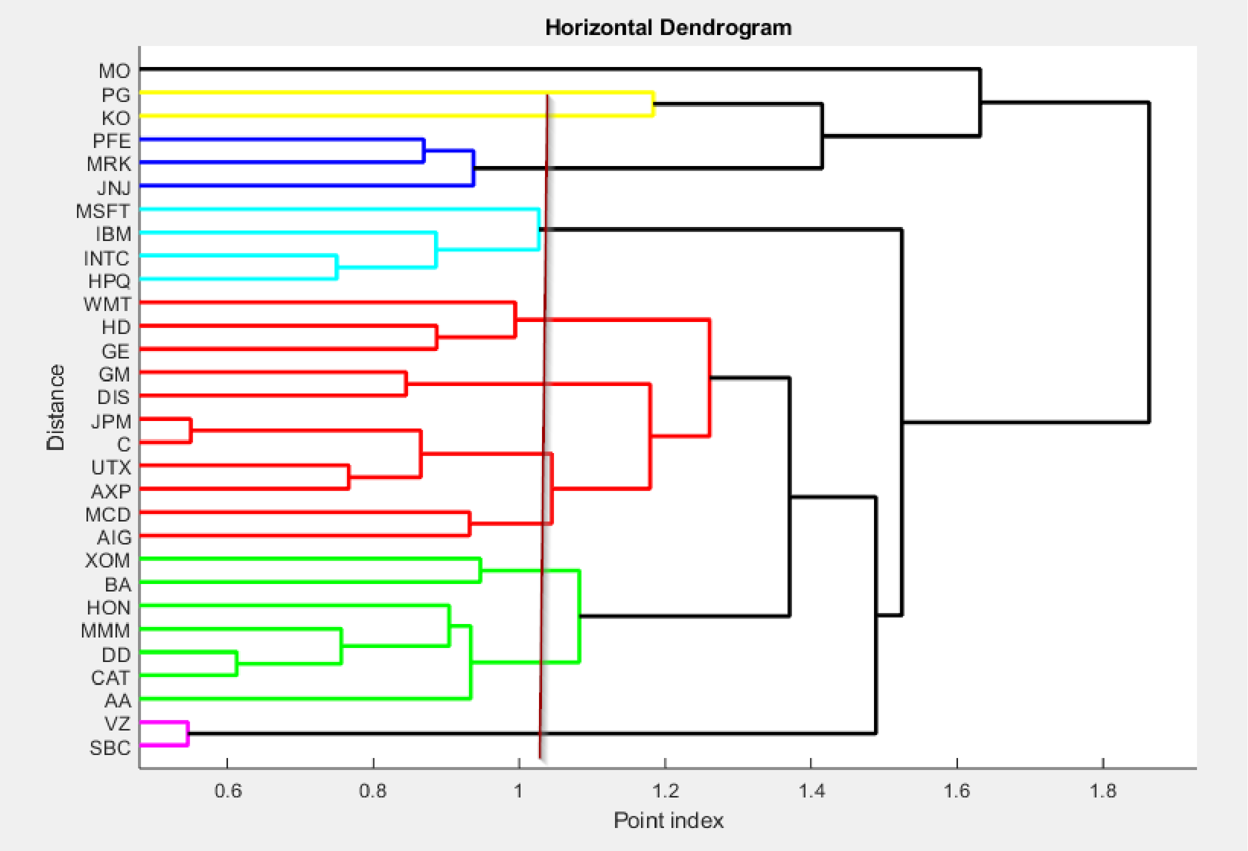


Figure 8. Few clusters of stocks

First cluster: the blue cluster is composed by MSFT, IBM, INTC and HPQ stocks.
Second cluster: the violet cluster is composed by PFE, MRK and JNJ
Third cluster: the pink cluster is composed by VZ and SBC

Blue cluster:

- MSFT (Microsoft corporation)
- IBM (IBM)
- INTC (Intel corporation)
- HPQ (Hewlett Packard)

This blue cluster is composed by stocks mostly applied in the technology sector. Particularly, it is a set of companies that offers computer, electronic chips, etc.

Violet cluster:

- PFE (Pfizer which is a pharmaceuticals business)
- MRK (Monica Rich Kosann)
- JNJ (Johnson and Johnson)

This violet cluster consists of stocks applied in pharmaceuticals business. They are all in drug business and specifically in medical services.

Pink cluster:

- VZ (Verizon communications)
- SBC which is owned by AT&T

This pink cluster consists of stocks applied in telecommunications sector. They offer communications services.

QUESTION3. Ensembles for classification

3.1 Name three sources of uncertainty and explain how they impact on the modelling process when using machine learning approaches.

Answer:

Three sources of uncertainty

- Observational uncertainty
- Parametrical uncertainty
- Structural uncertainty

How they impact on the modelling process when using machine learning approaches.

Observational uncertainty: when creating a model using machine learning approach, you might find some missing data points which gives rises to uncertainty. These uncertainties are called observational uncertainty. They might result in outliers. Those missing data points might also cause inaccuracy in the prediction using machine learning models.
Structural and parametrical uncertainty: a model used in modelling process might be biased (what you already know). Used software in modelling process might also be a cause of uncertainty. In fact, these two uncertainties affect the model selection.

3.2 What is the concept behind model averaging and give some examples of how this technique can be implemented in practice when generating predictions?

Answer:

Model averaging provides a coherent and systematic mechanism for accounting for model uncertainty. Basically, model averaging is described as a technique designed to help account for the uncertainty inherent in the model selection process. Model averaging works by averaging over many different competing models, and then the model uncertainty can be put into conclusions about parameters and prediction. As a result, it improves the predictive performance. Model averaging is used in ***model selection***, ***combined estimation*** and ***prediction which all helps in model choice*** and provide a way to get a less risky predictions. It provides a straightforward model choice criterion.

Giving some examples of how this technique can be implemented in practice when generating predictions.

1. **Financial time series**: Model averaging can be used in financial time series for successful predicting financial data.

2. **Weather forecasting**: to get the accurate prediction in weather forecasting, the model averaging is used.
3. **ECG (electrocardiogram)**: Model averaging can be used in classification of ECG signals. [7]

3.3 What kind of ensemble methods can be used to reduce the effects of uncertainty and improve on individual models? How do they achieve this goal?

Answer

Ensemble methods that can be used to reduce the effects of uncertainty and improve on individual models:

- Bootstrap aggregating or bagging (e.g.: Random forest for regression and classification.)
- Boosting
- Bayesian parameter averaging.
- Bayes optimal classifier.
- Bayesian model combination.

These methods use multiple learning algorithms to obtain better predictive power and performance. This provides an advantage over any single learning algorithms.
For example:

- Firstly, bootstrap uses random sampling to create many random sub samples from the original dataset with replacement (which means selecting the same value multiple times).
- Secondly, we calculate the mean of each created subsample.
- Lastly, we calculate the average of all the means and use this as the estimated mean for the original data.

How bagging and boosting achieve this goal:

1. **Bagging:**

Mostly known as ***bootstrap aggregating***.

Bootstrap refers to random sampling with replacement. Bootstrap allows us to better understand the bias and the variance with the dataset. Bootstrap involves random sampling of small subset of data from the dataset. It is a general procedure that can be used to reduce the variance for those algorithms that have high variance, typically decision trees. Bagging makes each model run independently and then aggregates the outputs at the end without preference to any model.
Basically bagging:

- Reduces the variance
- Helps in avoiding overfitting,
- Improves estimates from unstable procedures.

Example:
Having a training dataset X with size N, the process of bagging generates a new training dataset also called bootstraps, with each of size C by sampling from original X uniformly with replacement. [8]

2. **Boosting**

Boosting refers to a group of algorithms that utilize weighted averages to make weak learners into stronger learners. Boosting is all about “teamwork”. Each model that runs, dictates what features the next model will focus on.

In **boosting** as the name suggests, one is learning from other which in turn **boosts** the learning. In boosting, we incrementally build an ensemble by training each new model instance to emphasize the training instances that may mis qualified by previous models. [9]

3.4 Construct a random forest (RF) model and apply this to the Titanic dataset. Explain how you selected the optimal number of trees and support your choice using a graph.

Answer

The optimal number of trees is almost 33. These corresponds to the minimum out of bag classification error. So, the way I choose it is to look at the graph for the lowest out of bag classification error and see the corresponding number of grown trees.

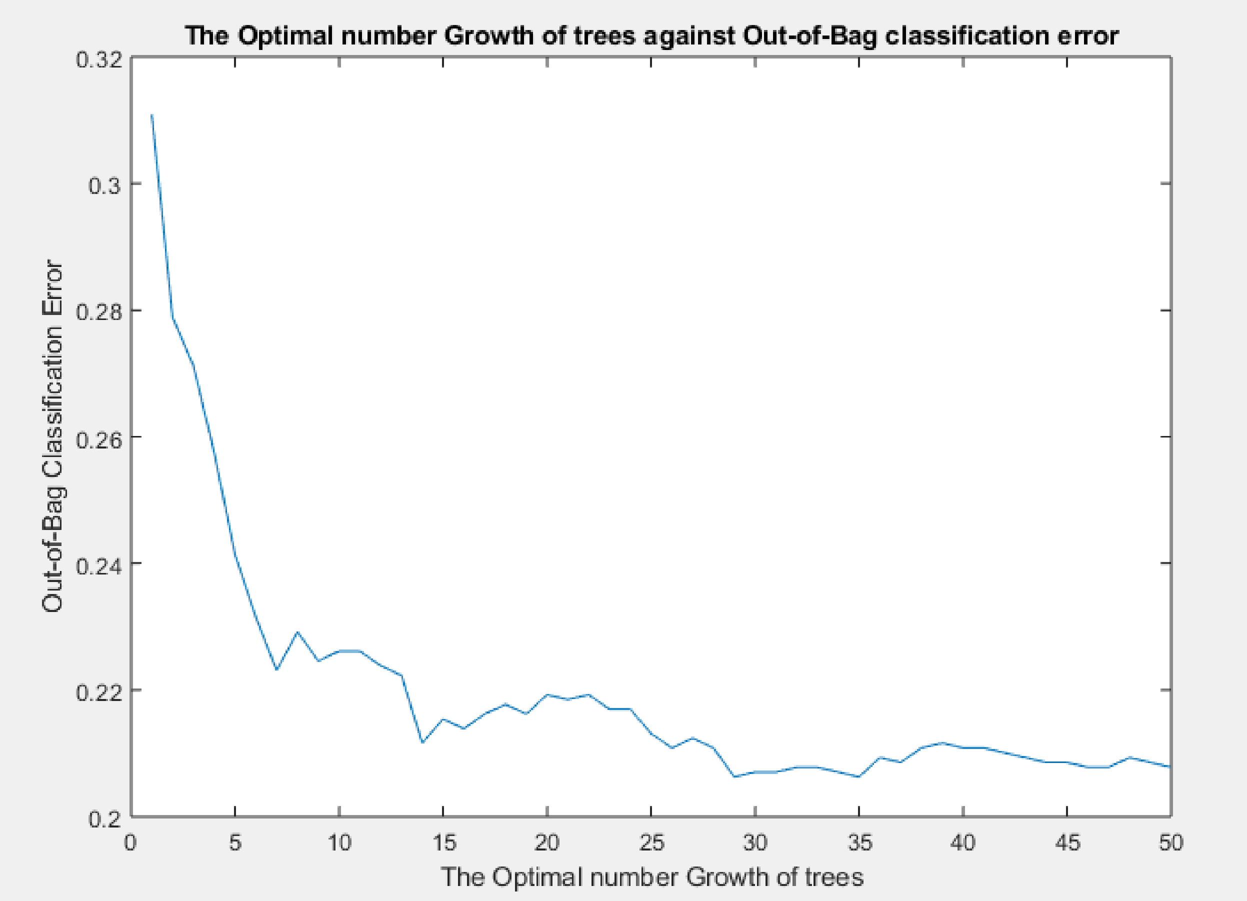


Figure 9. Optimal number of trees vs error

3.5 Undertake a ROC analysis and show how the RF performs relative to the previous models (logistic regression, classification tree and KNN). Provide evidence to show as clearly as possible which model is best for classifying survival on the Titanic.

Answer

For ROC analysis
Approaches made:

The ROC analysis is carried out for every model including random forest, logistic regression, classification tree and KNN.
Therefore, the plot graph obtained is shown below:

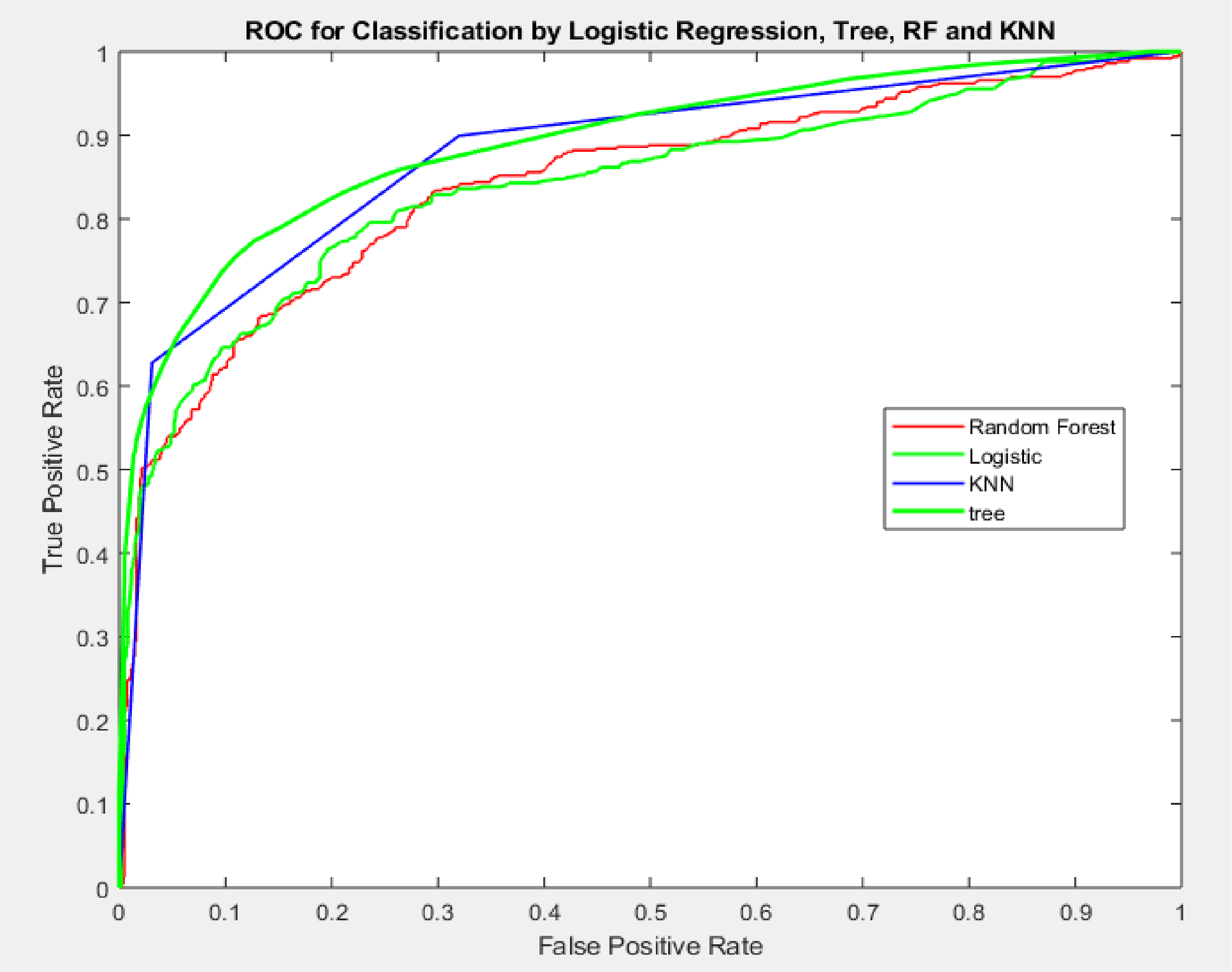


Figure 10. ROC analysis

The graph reveals us that the classification tree is the best model for classifying survival on the titanic. Its graph (the deep green) is the one which is above others.

QUESTION4. Ensembles for regression

The wine quality database provides information about the quality of wine. There are two datasets, one for red wine and one for white wine, which contain quality ratings, from one to ten, along with their physical and chemical properties. The challenge is to use these features to predict the rating for a wine and to assess performance. It is advisable to study white and red wine separately:
https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality

4.1 Describe the concept of a random forest (RF) regression model.

Answer:
Random forest (RF) regression model:

Random forest (RF) regression model is a bagging technique and not a boosting technique. The trees in random forests are run in parallel. There is no interaction between these trees while building the trees. It operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

Basically, a random forest is a meta-estimator (i.e. it combines the result of multiple predictions) which aggregates many decision trees, with some helpful modifications:

1. The number of features that can be split on at each node is limited to some percentage of the total (which is known as the hyperparameter). This ensures that the ensemble model does not rely too heavily on any individual feature and makes fair use of all potentially predictive features.

2. Each tree draws a random sample from the original data set when generating its splits, adding a further element of randomness that prevents overfitting. [10]

4.2 Construct a random forest (RF) model for the red wine dataset and show how the optimal number of leafs was estimated.

Answer

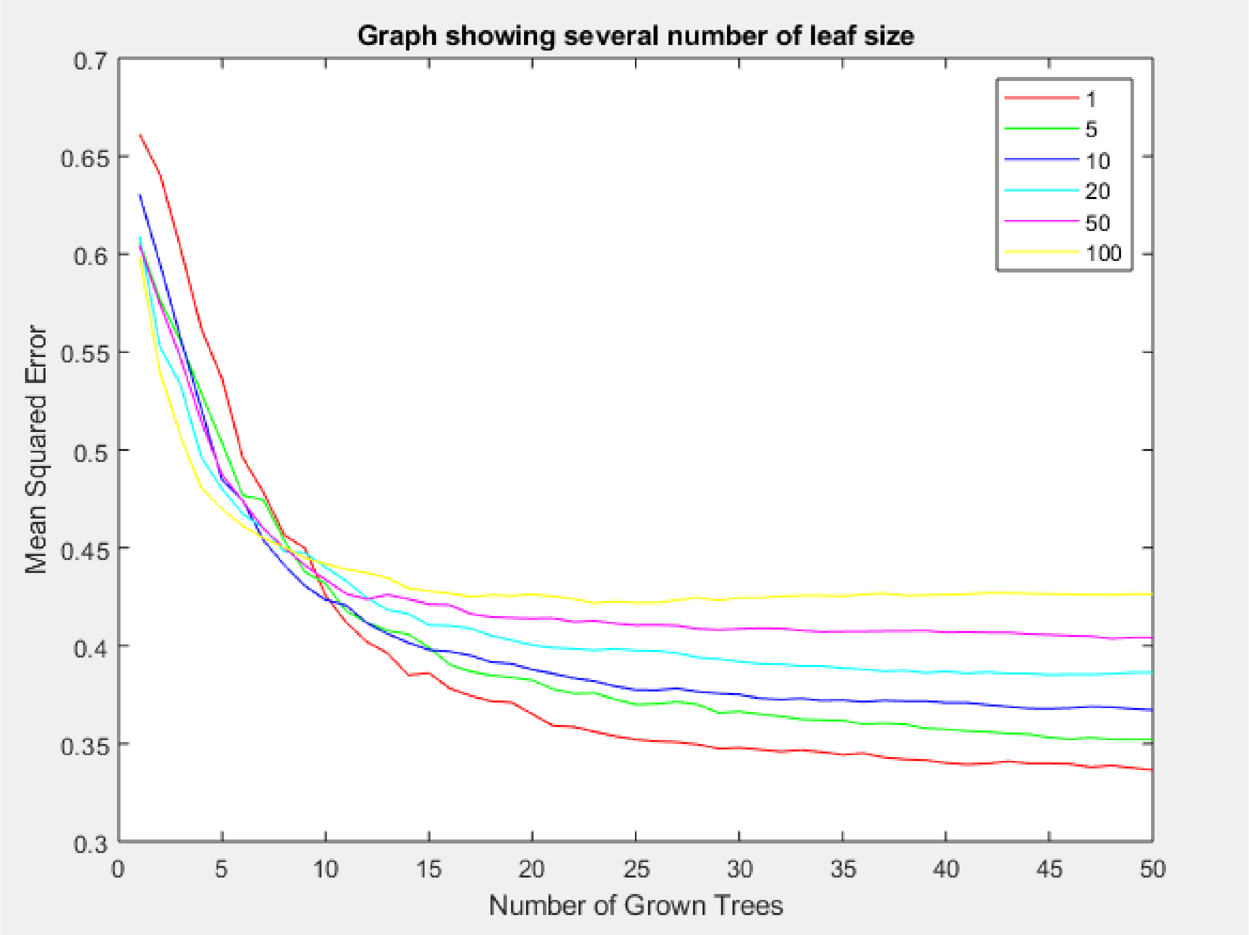


Figure 11. Optimal number of leafs

The optimal number of leafs is given by the minimum mean squared error (MSE). As can be seen on the graph, red graph gives the smallest mean squared error, which means optimum leaf = 1.

4.3 Explaining and showing how the optimal number of trees was computed.

Answer

Optimal number of trees is 100 because it is one with smallest out of bag mean squared error.

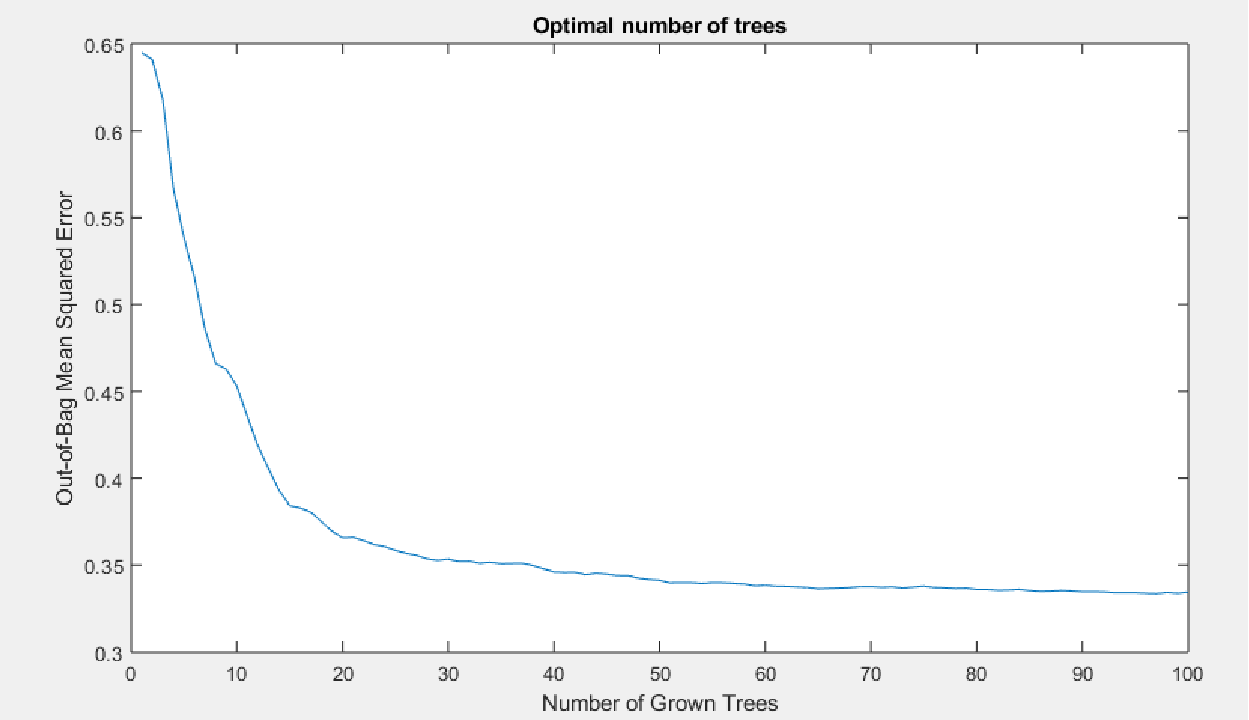


Figure 12. Optimal number of trees.

Approaches made:

- treebagger MATLAB function is used to create a model.
- Finding the out of bag error of the model in order to find out of bag error and number of grown trees. Thus, the number of trees which corresponds to smallest error is the optimal number of those trees.

4.4 Provide a bar graph showing the importance of each feature and compare this with the results from assignment 2 (using correlation and LASSO).

Answer

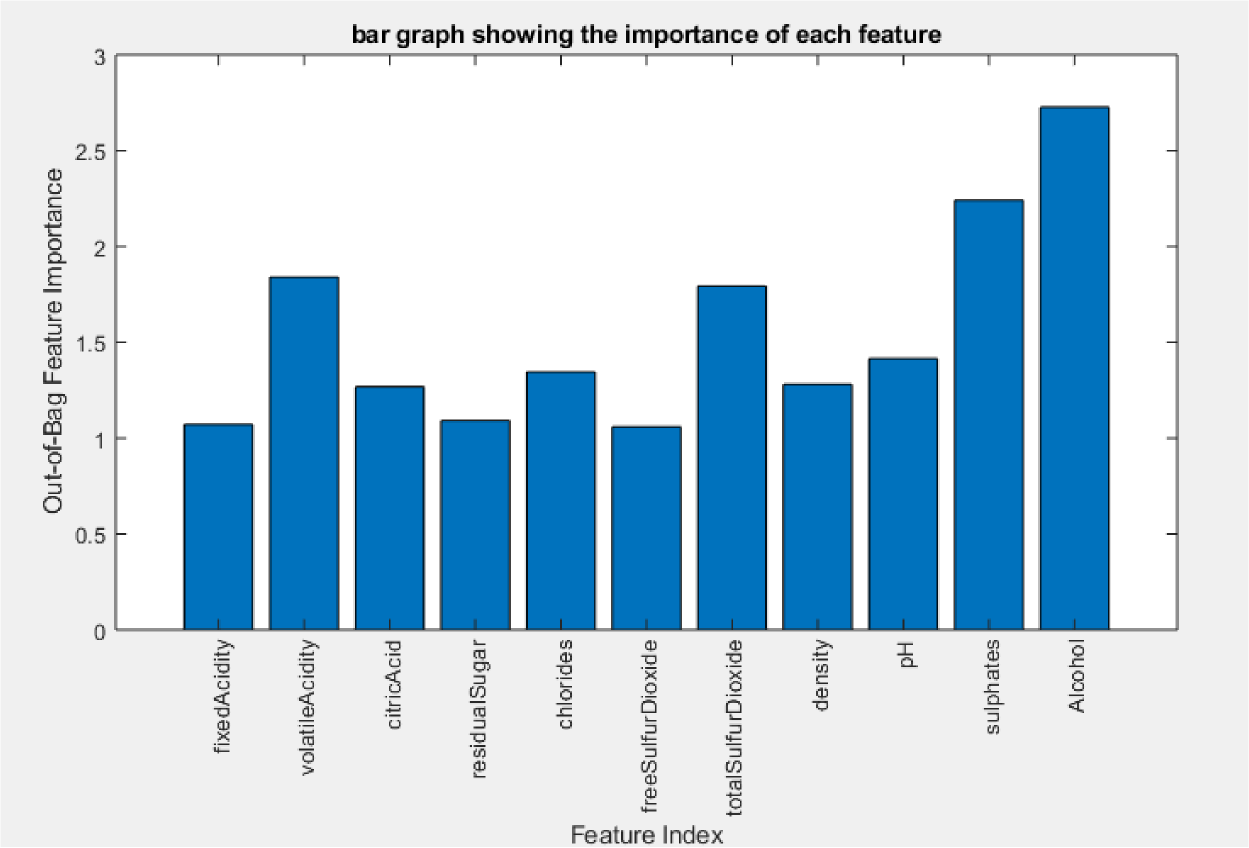


Figure 13. Importance of each feature

The graph reveals us that the alcohol is the most important feature because it is the highest in the result. The second most important is sulphates, the third most important is volatile acidity, total Sulphur dioxide.

Comparing this with the results from assignment 2 (using correlation and LASSO).

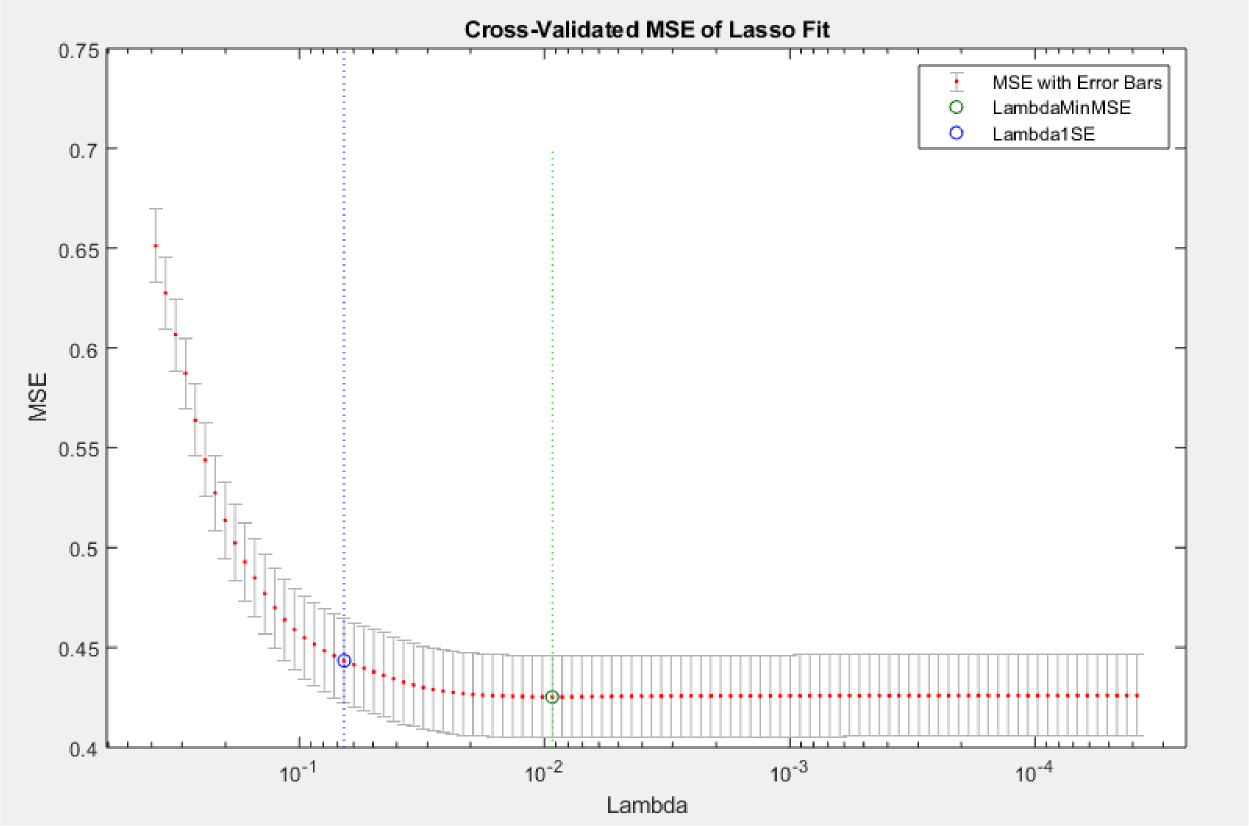


Figure 14. Lasso comparison

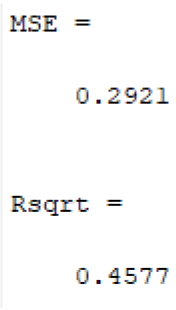
While for LASSO, selected the following: volatile acidity, residual sulphates, chlorides, free sulfur, total sulfur, density, pH, sulphates and alcohol.

Both of the models (LASSO and Random forest) select same variables, because LASSO eliminated free acidity and citric acid which is almost the same case with random forest which also gives the low importance to those two variables (free acidity and citric acid), in addition, residual sugar and Ph also have the low importance.

4.5 What is the performance of the RF model and compare it with the linear regression and KNN models constructed during assignment 2. Present sufficient information to support your conclusion about the best model for the red wine dataset.

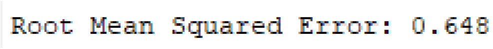
Answer

The MSE and R square of the random forest model are:

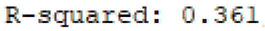


```
MSE =

    0.2921


Rsqrt =

    0.4577
```

For linear regression



```
Root Mean Squared Error: 0.648
```



```
R-squared: 0.361
```

For KNN
MSE = 0.3577
Rsquare = 0.3499

Therefore, the random forest is the best model for red wine dataset because it provides the lowest mean square error (MSE).

References

[1] Wikipedia, "Principal component analysis," Wikipedia, 28 November 2019. [Online]. Available: https://en.wikipedia.org/wiki/Principal_component_analysis. [Accessed 2 December 2019].

[2] S. S. Nazrul, "The DOs and DON'Ts of Principal Component Analysis," Medium, 31 March 2018. [Online]. Available: https://medium.com/@sadatnazrul/the-dos-and-donts-of-principal-component-analysis-7c2e9dc8cc48. [Accessed 27 November 2019].

[3] M. A. Alizadeh-Khameneh, "Linkages and Dendrograms," in *Tree Detection and Species Identification using LiDAR Data*, Stockholm, Sweden, ResearchGate, 2013, p. 17.

[4] MathWorks, "Hierarchical Clustering," MathWorks, 2019. [Online]. Available: https://www.mathworks.com/help/stats/hierarchical-clustering.html. [Accessed 28 November 2019].

[5] MathWorks, "dendrogram," MathWorks, 2019. [Online]. Available: https://www.mathworks.com/help/stats/dendrogram.html. [Accessed 28 November 2019].

[6] MathWorks, "Documentation," MathWorks, November 2019. [Online]. Available: https://www.mathworks.com/help/stats/hierarchical-clustering.html. [Accessed 28 November 2019].

[7] Tiago M. Fragoso1,Wesley Bertoli2 and Francisco Louzada3, "Bayesian Model Averaging: A Systematic Review and Conceptual Classification," in *International Statistical Review*, Oxford, John Wiley & Sons Ltd, 2017.

[8] A. Chakure, "Random Forest Regression," Medium, 29 June 2019. [Online]. Available: https://towardsdatascience.com/random-forest-and-its-implementation-71824ced454f. [Accessed 29 November 2019].

[9] J. A. V. B. A. Robinson, "Treatment of uncertainty using ensemble methods: Comparison of sequential data assimilation and Bayesian model averaging," AGU100, 17 January 2007. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2005WR004838. [Accessed 29 November 2019].

[10] A. Chakure, "Random Forest Regression," Medium, 29 June 2019. [Online]. Available: https://towardsdatascience.com/random-forest-and-its-implementation-71824ced454f. [Accessed 29 November 2019].