CARNEGIE MELLON UNIVERSITY - AFRICA

DATA, INFERENCE & APPLIED MACHINE LEARNING

(COURSE 18-785)

Professor Patrick McSharry

Assignment 6

MUNYANEZA Kenny Roger

20th November 2023

Question 1

1. Non-linearity

1.1.

Non-linear relationship between variables is best used to describe a situation whereby changes in the output (dependent variable) do not change in direct proportion to changes in any of inputs (independent variables). Mathematically, these relationships are polynomial equations that take at least one curve or turn to fit the all output values. [1] [2]

1.2.

An equation for non-linear model can be a Quadratic equation given by $ax^2 + bx + c = 0$. [3]

One example of quadratic equation might be total weights lifted in the Gym and physical fitness.

As weights lifted in the gym increase from zero, the over fitness of the body tends to increase, but beyond a certain threshold more weights lifted lead decreased body fitness due to muscles and connective tissue injuries.

1.3.

Yes, a nonlinear model uses fewer parameters hence it is more parsimonious than a linear model.

Linear model equation : $y = a + bx$, where x is the explanatory variable and y is the dependent variable.

Where:

- x = predictor variable
- y = target variable
- b = slope of the linear
- a = constant value of y in case all predictor variables are zero.

Nonlinear Regression: $Y = f(X,\beta) + \varepsilon$,

Where:

- X = a vector of p predictors
- β = a vector of k parameters,
- f() = a known regression function
- ε = an error term

[4]

1.4.

1.4.1. Polished surrogates attempt to preserve both the unconditional distribution and the linear correlations.

1.4.2. There are two surrogate techniques, namely:

- Parametric which generates surrogate data series using a model that has been fit to the original data.
- Non-parametric which generates surrogate data series directly from the original data.

1.4.3. Below are different approaches for implementing the above techniques:

Random Shuffle:

This approach involves generating surrogates by randomly rearranging the original data. The goal is to maintain the original data's distribution while breaking any linear correlations.

Random Phases:

Surrogate data is created by applying the inverse Fourier Transform to the amplitudes obtained from the Fourier Transform of the original data, with the inclusion of new (uniformly random) phases. This method is designed to retain the linear correlations within the data while introducing randomness through the new phases.

Amplitude Adjusted Fourier Transform (AAFT)

This approach merges the principles of random shuffle and random phases, preserving both the distribution and linear correlations. An iterative form of AAFT is also accessible to refine the surrogates, aiming for an improved alignment with both the distribution and the linear correlations, including the autocorrelation function. [2]

1.5.

1.5.1.

- Information is measured by:

  $I(x) = -\log p_x(x)$

  Where p is the probability of observing the variable x.
- Entropy is measured by:

  $H(x) = -\int p_x(x)\log[p_x\,(x)]dx$
- Mutual Information:

  $I(x,y) = H(x) + H(y) - H(x,y)$

1.5.2.

Originally, regularity was assessed through precise regularity statistics focused on entropy measures. However, calculating entropy accurately demands extensive data, and outcomes are significantly affected by system noise. Consequently, employing such methods on experimental data is impractical. A modification called Approximate entropy (ApEn) enables entropy estimation based on empirical observations.[5] (ApEn) is a technique used to quantify the amount of regularity and the unpredictability of fluctuations over time-series data. Notably, Sample Entropy (SampEn) is advantageous as it remains independent of the quantity of data available. [2]

Example application of regularity measurement using entropy:

- Approximate entropy can be applied to analyze time-series data in meteorology and climatology. It would help assess the regularity or irregularity in weather patterns or climate variables.[5]

1.5.3.

Minimum-redundancy-maximum-relevance (mRMR) as a feature selection method relies on measuring relevancy and redundancy. **Given a feature set S containing features x_i ,**

- The average value of all mutual information values between individual features x_i and the class Y defines the relevance of the feature set S for the class y.

$$D(S,y)=\frac{1}{|S|}\sum_{x_i\in S} I(x_i,y)$$

- The average value of all mutual information values between the feature x_i and the feature x_j defines the redundancy of all features in the set S.

$$R(S)=\frac{1}{|S|^2}\sum_{x_i,x_j\in S} I(x_i,x_j)$$

- Mutual Information might be better than Correlation because it can capture nonlinear relationships between variables, whereas correlation only measures linear dependence.[2]

Question 2

2. Classification using trees

2.1.

2.1.1. Components of a decision tree:

- **Root Node :** This is the topmost node in a tree representing the complete dataset. Decision making process starts at this point.
- **Decision / Internal Node :** Representing a decision related to an input feature, this node branches out to connect with leaf nodes or other intermediate nodes.
- **Branch/Sub-Tree:** A subsection of the decision tree that starts at an internal node and ends at the leaf nodes.
- **Leaf/Terminal Node:** A node without any child that represents the result of the algorithm. The result may be a class label or a numerical value.

[6]

2.1.2. Pruning is the process of removing unnecessary branches that do not provide additional information from the decision tree. Such branches are removed to prevent overfitting of the decision tree.

2.1.3.

Why decision trees are an attractive method for classification in practical applications:

- They are easy to interpret and explain.
- They are robust to outliers and missing values.
- They are non-parametric and non-linear, meaning they do not depend on any distribution assumptions or linear relationships.
- They can perform feature selection and handle categorical data.
- They require minimal data preparation and preprocessing.[7]

2.2.

2.2.1. Steps to creating a data-driven classifier[8] [9]:

1. **Data Collection:** Gather a large and diverse dataset that represents the problem domain comprehensively. Ensure the dataset includes a mix of different scenarios and conditions.
2. **Data Preprocessing:** Clean the collected data to handle missing values, outliers, and inconsistencies and convert categorical variables into numerical representations using techniques like one-hot encoding. Also, Normalize or scale numerical features, as necessary.
3. **Feature Engineering:** Identify relevant features that contribute to the classification task and create new features if needed, based on insights from the data.
4. **Model Selection:** Choose a suitable data-driven classification algorithm based on the nature of the problem (e.g., decision trees, random forests, support vector machines, neural networks).
5. **Model Training:** Split the dataset into training and validation sets and train the selected model on the training set.
6. **Model Evaluation:** Use metrics such as accuracy, precisions and F1 score to determine the performance of the trained model.
7. **Model Tuning:** Optimize the model's hyperparameters using techniques like grid search or random search to enhance performance.

2.2.2. To test the validity of the new model[10]:

Cross-Validation:
Employ techniques like k-fold cross-validation to assess the model's performance across multiple subsets of the data.

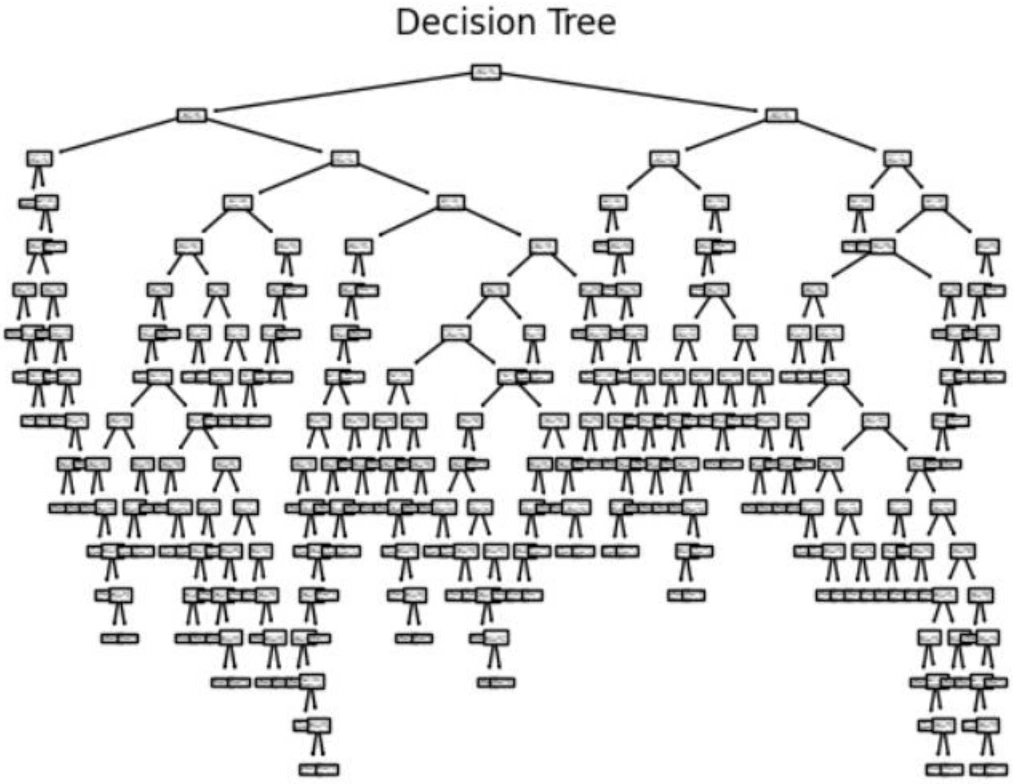
Holdout Testing:
Reserve a portion of the data for a final test after model training and validation.

A/B Testing (if applicable):
Compare the performance of the new model against the rule-based classifier in a controlled experiment.

2.3.

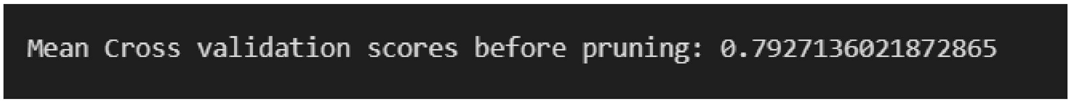
I split the dataset into train and test and fit a decision tree classifier model using the train set of the data.

Decision Tree classifier of the titanic dataset



Using cross validation, I assessed the accuracy of the decision tree classifier as the **mean cross validation score** of the model.

Results obtained:



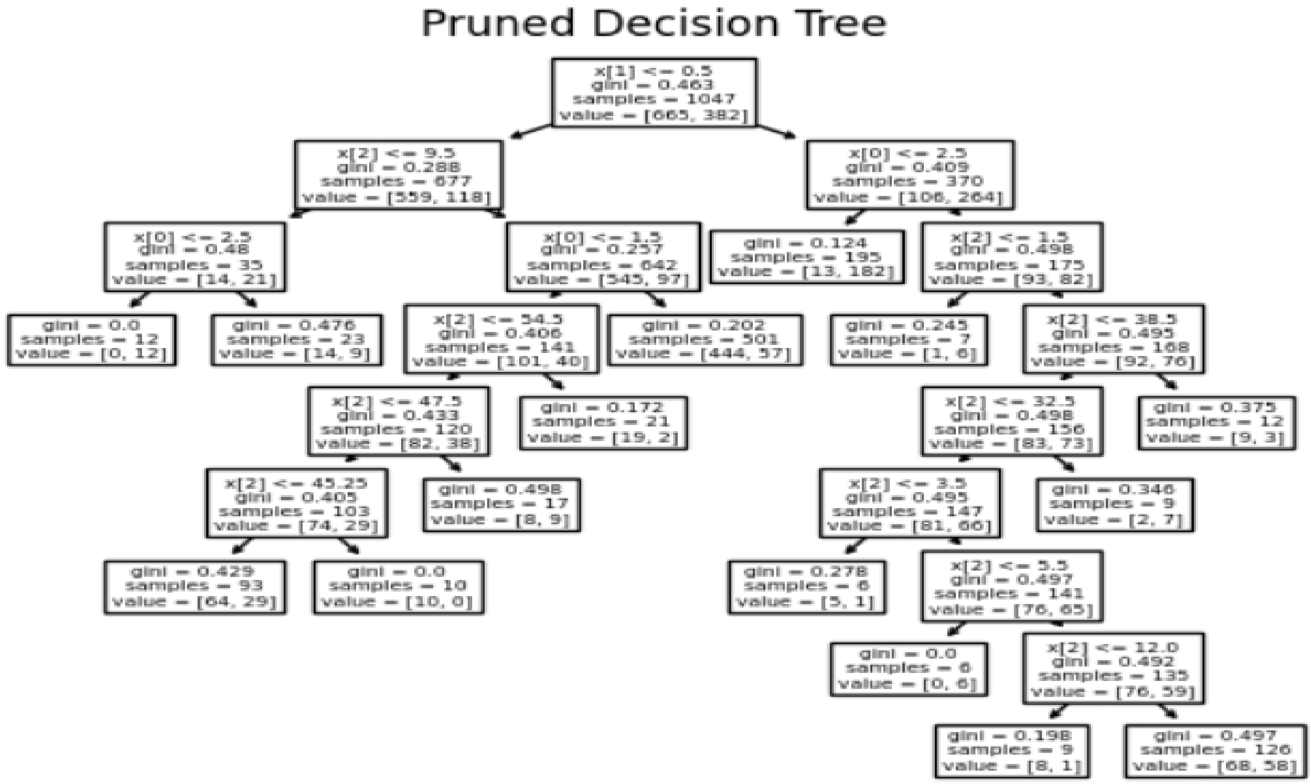
```
Mean Cross validation scores before pruning: 0.7927136021872865
```

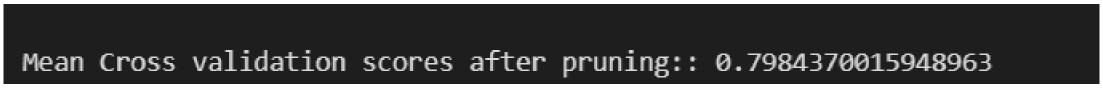
2.4.

Pruning

I used a post pruning method called Cost Complexity pruning. This method involves determining the right parameter for alpha. I used cost_complexity_pruning_path() function that returns an array of effective alphas for each subtree along the pruning path, and an array of impurities for each subtree along the pruning path. Then, I looped through the array of alpha values, and for each alpha value, I fitted a decision tree classifier and determined its accuracy based on cross validation score. At the end of the loop, I obtained the optimal alpha value to fit the decision tree classifier and used it as parameter to fit the pruned decision tree classifier.

Results obtained

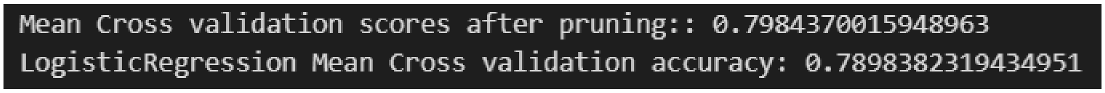




```
Mean Cross validation scores after pruning:: 0.7984370015948963
```

2.5.

I compared the final decision tree to the logistic regression model basing on the Mean Cross Validation of the two model.



```
Mean Cross validation scores after pruning:: 0.7984370015948963
LogisticRegression Mean Cross validation accuracy: 0.7898382319434951
```

Based on the above performances of the two models, the final decision tree classifier model after pruning would be better to compete in the Kaggle competition.

Advantages of Logistic regression classifier:

- Easy to implement yet provides great training efficiency.
- In a low dimensional dataset, logistic regression is less prone to over-fitting.
- Very fast at classifying unknown records.
- It provides the direction of association between features (positive or negative).

Disadvantages of Logistic regression classifier:

- Logistic regression assumes linearity between independent and dependent variable.
- Logistic regression cannot solve non-linear problems because it has a linear decision surface.
- Logistic regression is sensitive to multicollinearity between independent variables.
- On high dimensional datasets, the model may be over-fit on the training set.

Advantages of Decision tree

- Decision trees are versatile, serving both classification and regression purposes: They can effectively predict both continuous and discrete values, making them suitable for a wide range of tasks in both regression and classification domains.
- Ability to capture nonlinear relationships: Decision trees are capable of classifying data that exhibits nonlinear separability, allowing them to handle complex relationships between variables.
- Non-parametric nature: Decision trees are classified as non-parametric methods, meaning they do not rely on predefined assumptions about the spatial distribution or structure of the classifier. This flexibility makes them adaptable to various types of data without imposing strict model assumptions.

Disadvantages of Decision tree

- Decision tree complexity increases with many features: Training time becomes more time-consuming as the number of input features grows.
- Unsuitable for big data: When dealing with large datasets, a single decision tree may grow excessively in terms of nodes, introducing complexity issues and potential overfitting.
- Overfitting challenges: Decision trees face difficulties with overfitting, a significant concern. Addressing overfitting can be achieved by imposing constraints on model parameters and utilizing pruning methods to refine the tree structure.[11]

Question 3

3. Classification using KNN

3.1.

The KNN algorithm operates like a voting mechanism, relying on the majority class label within the nearest 'k' (where 'k' is an integer) neighbors in the feature space to assign the class label for a new data point.
When presented with a test point x, identify the K nearest neighbors from the training dataset.
If Kk points are classified under class Ck, the probability that x belongs to class Ck is expressed as p(Ck|x) = Kk/K.
Subsequently, the test point x is categorized into the class with the highest p(Ck|x) value.[12]

Step by step procedure to implement this model:

- ❖ Select a value for k, representing the number of neighbors to take into account.
- ❖ Compute the distance between the new data point and all training data points using an appropriate metric (e.g., Euclidean, Manhattan).
- ❖ Arrange the distances in ascending order and choose the k nearest data points.
- ❖ For classification tasks, assign the most common label among the k neighbors to the new data point. For regression tasks, assign the average value among the k neighbors to the new data point.

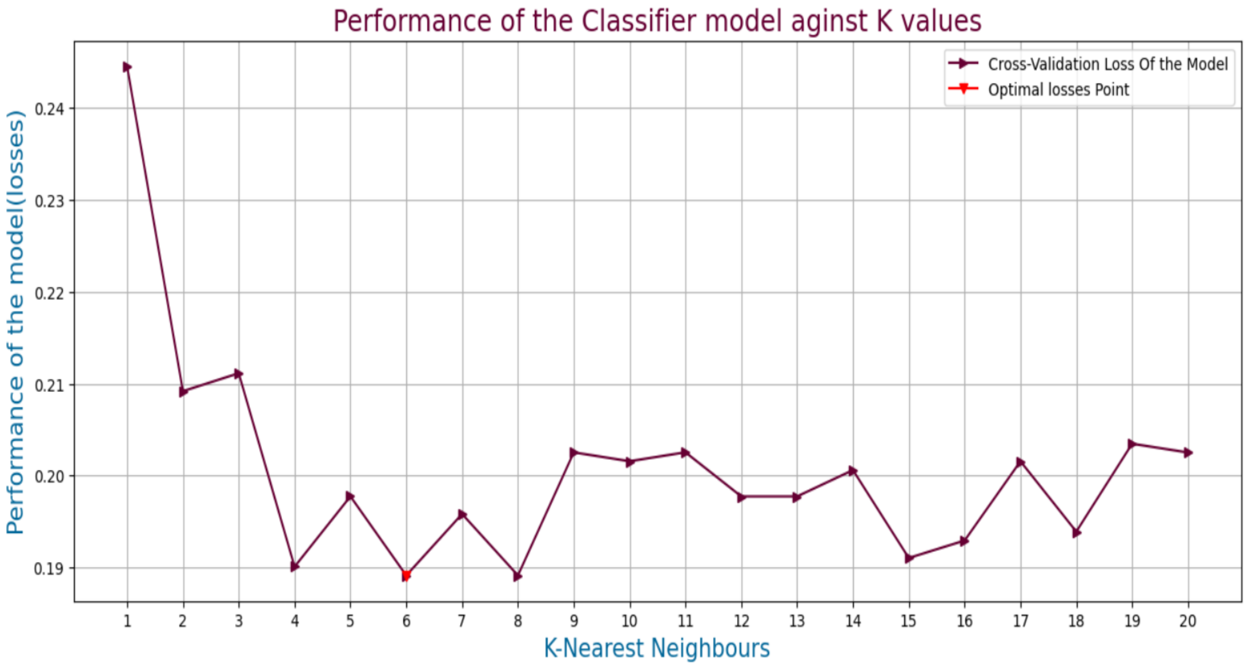
3.2.

- ❖ First, I will split the dataset into train and test data and then I will normalize the predictor variables using scikit-learn's standard scaler. Initially, the **" .fit_transform() "** method is applied to the training data, adjusting the scaler based on the mean and standard deviation of the training data. Subsequently, this transformation is extended to the test data using **" .transform() "** .

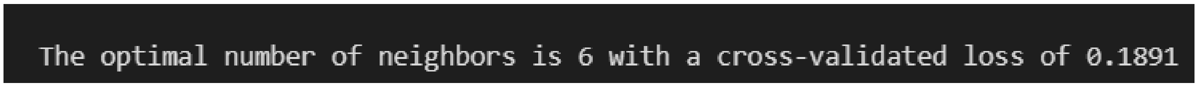
3.3.

Having split the dataset into train and test datasets and normalizing them as I mentioned above, I tested the performance of the KNeighborsClassifier model for k values ranging from 1 to 20.
I used a loop that iterates 20 times, and for each iteration, I fit the KNN model, calculated the accuracy score using cross validation function called “cross_val_score”, and I determined the model with optimal loss by subtracting the mean of accuracy scores of each model by one.
Lastly, I plotted the graph of loss against the number of k, highlighting the optimal point with minimum losses. [13]

Results obtained



Inferences:



```
The optimal number of neighbors is 6 with a cross-validated loss of 0.1891
```

Insights:

- Given the values of Losses, the cross-validation accuracy scores are greater than the cross-validation accuracy losses for each value of K.

3.4.

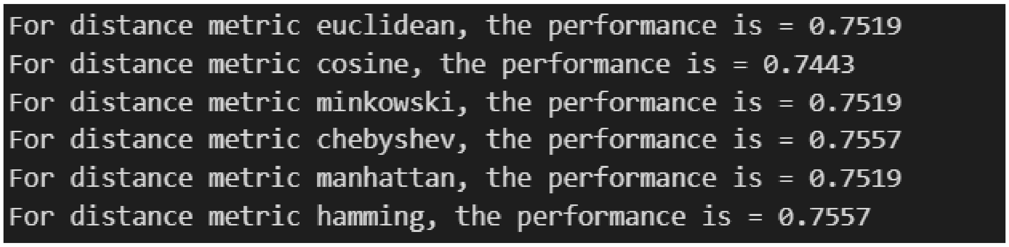
3.4.1.

Distance metrics are sensitive to the kind of features used because distances such as Euclidean Distance, Manhattan Distance and Minkowski Distance rely on the assumption that features are continuous and share a uniform scale, while Hamming distance operates under the assumption that features are binary or categorical.[14]

3.4.2.

I created a list of all types of distance metrics namely Euclidean, cosine, minkowski, Chebyshev, Manhattan and hamming, which I iterated through passing each distance metric as value for the parameter “metric” of the “KNeighborsClassifier” model.

Result Obtained

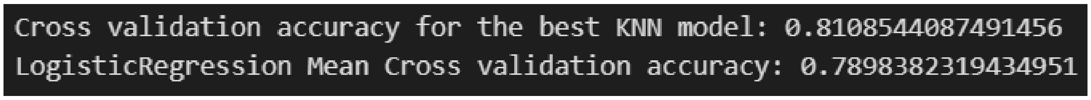


```
For distance metric euclidean, the performance is = 0.7519
For distance metric cosine, the performance is = 0.7443
For distance metric minkowski, the performance is = 0.7519
For distance metric chebyshev, the performance is = 0.7557
For distance metric manhattan, the performance is = 0.7519
For distance metric hamming, the performance is = 0.7557
```

3.5.

I fitted KNeighborsClassifier model with 6 nearest neighbors and I used cross validation to measure the cross-validation score using the train sub-dataset. Then, I repeated this process for a logistic regression model.

Results obtained



```
Cross validation accuracy for the best KNN model: 0.8108544087491456
LogisticRegression Mean Cross validation accuracy: 0.7898382319434951
```

Comparing KNeighborsClassifier and Logistic Regression model[15] [16]

Advantages of KNN classifier:

- Simple and easy to train and test non-linear datasets.
- No assumption about the underlying data distribution.
- Faster because it does not take a training period, only learns data at the time of making predictions.
- Easy to implement since it only requires k value and a distance metric as parameters.

Disadvantages of KNN classifier:

- Sensitive to noise hence missing values need to be defaulted and outliers need to be removed.
- Performance can degrade with high-dimensional data.
- Requires feature scaling (normalization) before applying KNN algorithm.
- Optimal choice of k is crucial for the model to be accurate.

Advantages of Logistic regression classifier:

- Easy to implement yet provides great training efficiency.
- In a low dimensional dataset, logistic regression is less prone to over-fitting.
- Very fast at classifying unknown records.
- It provides the direction of association between features (positive or negative).

Disadvantages of Logistic regression classifier:

- Logistic regression assumes linearity between independent and dependent variable.
- Logistic regression cannot solve non-linear problems because it has a linear decision surface.

- Logistic regression is sensitive to multicollinearity between independent variables.
- On high dimensional datasets, the model may be over-fit on the training set.

Conclusion: Since the KNN classifier has a better performing model with performance **0.81,** the KNN model would be most appropriate for the titanic dataset.

Question 4

4. Regression – wine quality

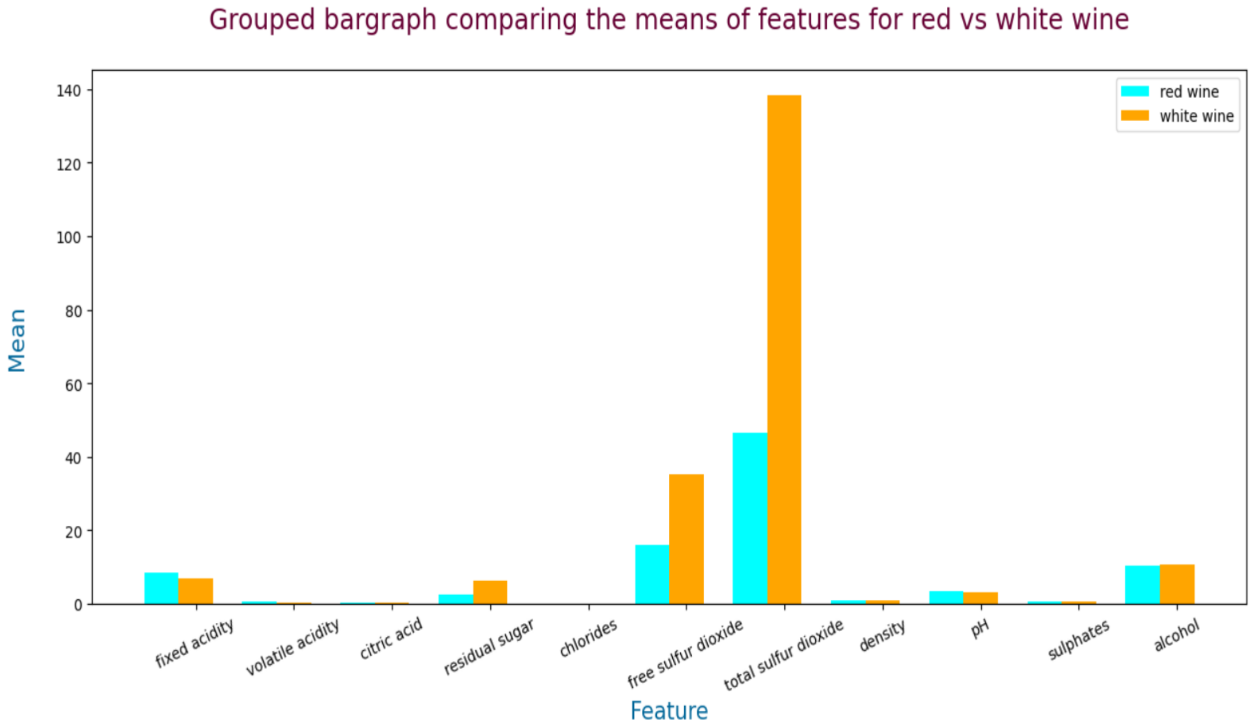
Technologies used

- Pandas : To manipulate csv datasets
- Numpy: to perform mathematical calculations
- Matplotlib: Used its submodule called Pyplot to graph computation results
- Sklearn: To acquire and use various machine learning models

4.1.

I extracted the feature variable for both the red wine and white wine dataset. Then I computed the mean per feature for each dataset and I plotted a grouped bar graph comparing how the means of features compare for red wine vs white wine.

Result obtained:



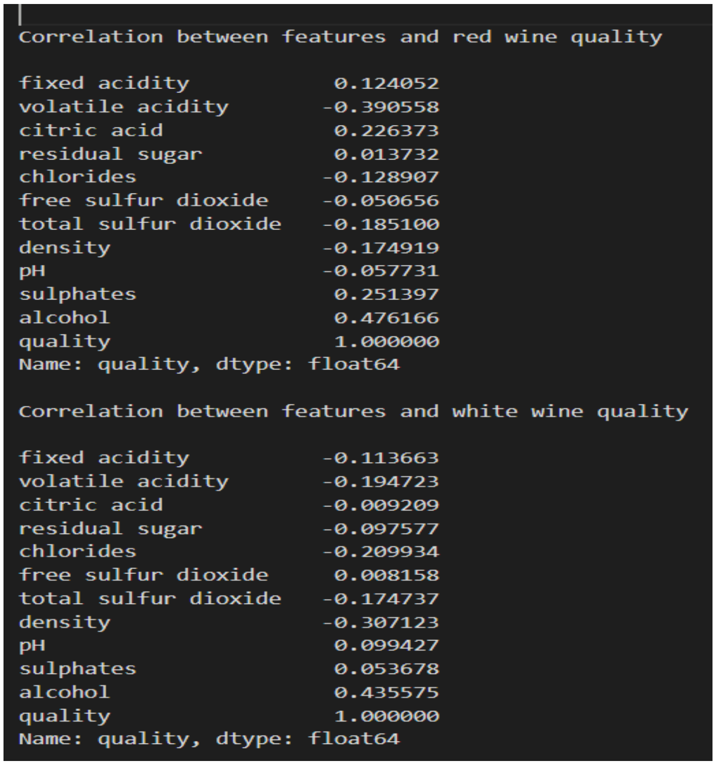
Inference on the observations:

- Sulfur dioxide is the highest component in both wines because it is a critical component of wines to preserve the wine's color and taste. However, white wine has a higher sulfur dioxide quantity than red wine.
- On average, both wines have approximately the same alcohol composition.
- Also, the level of acidity measured by the power of hydrogen (PH) is approximately the same in both wines.

4.2.

I calculated the correlation of each feature to the target variable and the correlation coefficients obtained indicate that **alcohol** is the most relevant feature.

- Correlation coefficient of alcohol with red wine quality = 0.476
- Correlation coefficient of alcohol with white wine quality = 0.436



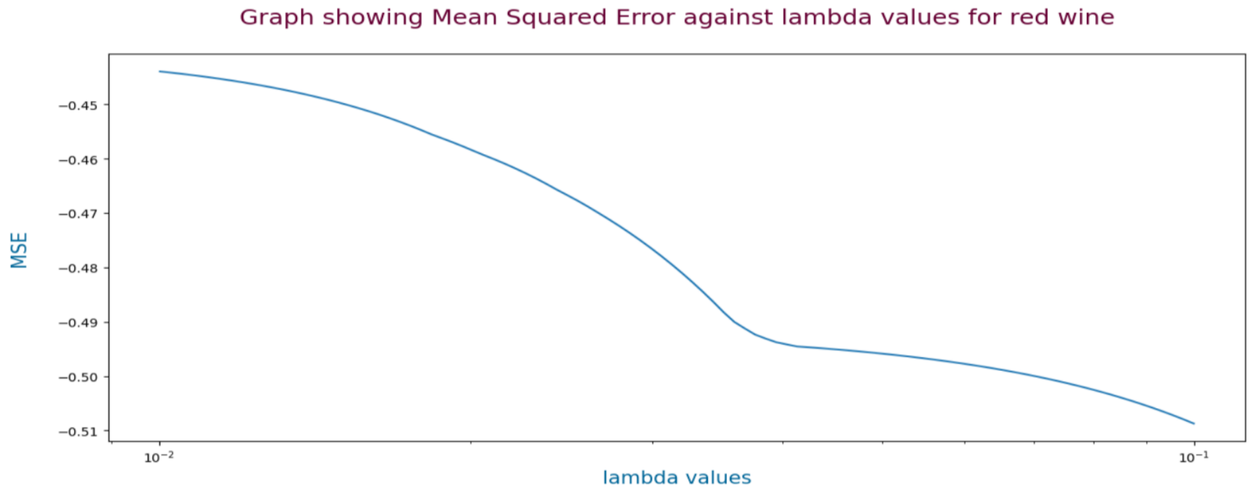
```
Correlation between features and red wine quality

fixed acidity           0.124052
volatile acidity       -0.390558
citric acid             0.226373
residual sugar          0.013732
chlorides              -0.128907
free sulfur dioxide    -0.050656
total sulfur dioxide   -0.185100
density                -0.174919
pH                     -0.057731
sulphates               0.251397
alcohol                 0.476166
quality                 1.000000
Name: quality, dtype: float64

Correlation between features and white wine quality

fixed acidity          -0.113663
volatile acidity       -0.194723
citric acid            -0.009209
residual sugar         -0.097577
chlorides              -0.209934
free sulfur dioxide     0.008158
total sulfur dioxide   -0.174737
density                -0.307123
pH                      0.099427
sulphates               0.053678
alcohol                 0.435575
quality                 1.000000
Name: quality, dtype: float64
```

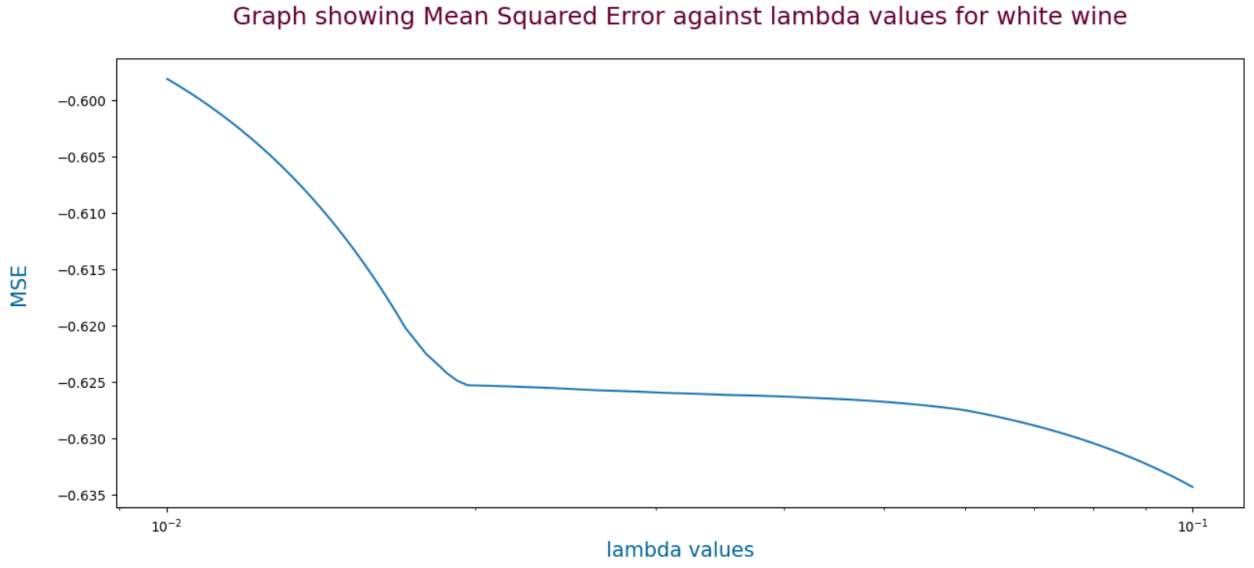
4.3.

Setting the lambda values to be in the range of log(0.001) to log(1), I fitted Lasso() models for the red wine dataset and the white wine dataset separately.

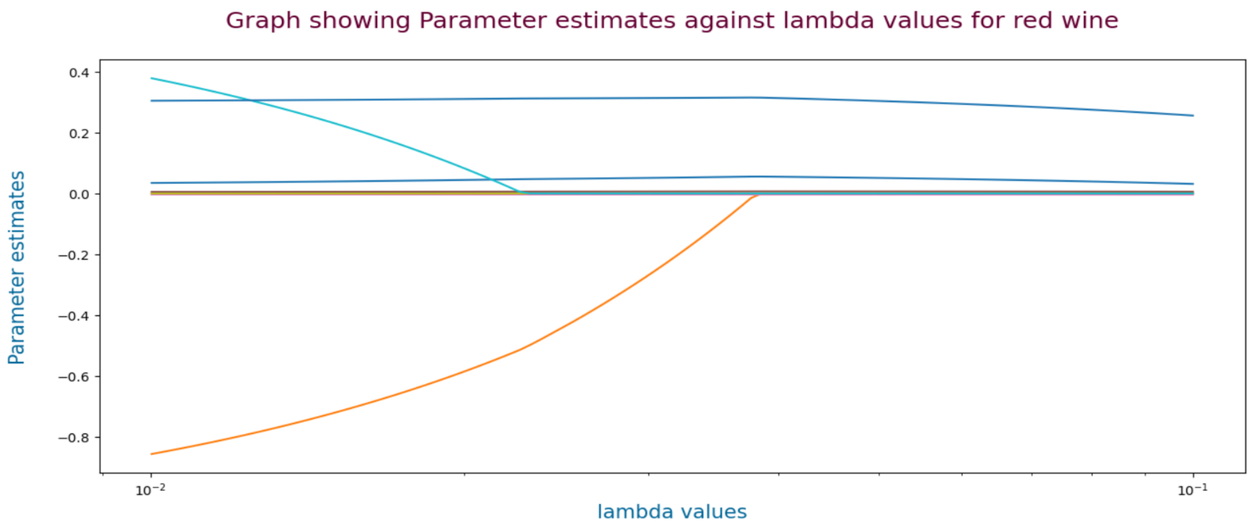
The I used cross validation to determine the Mean Squared Error(MSE), and parameter estimates for each model. Lastly, I plotted MSE against lambda values and parameter estimates against lambda values for each wine type.

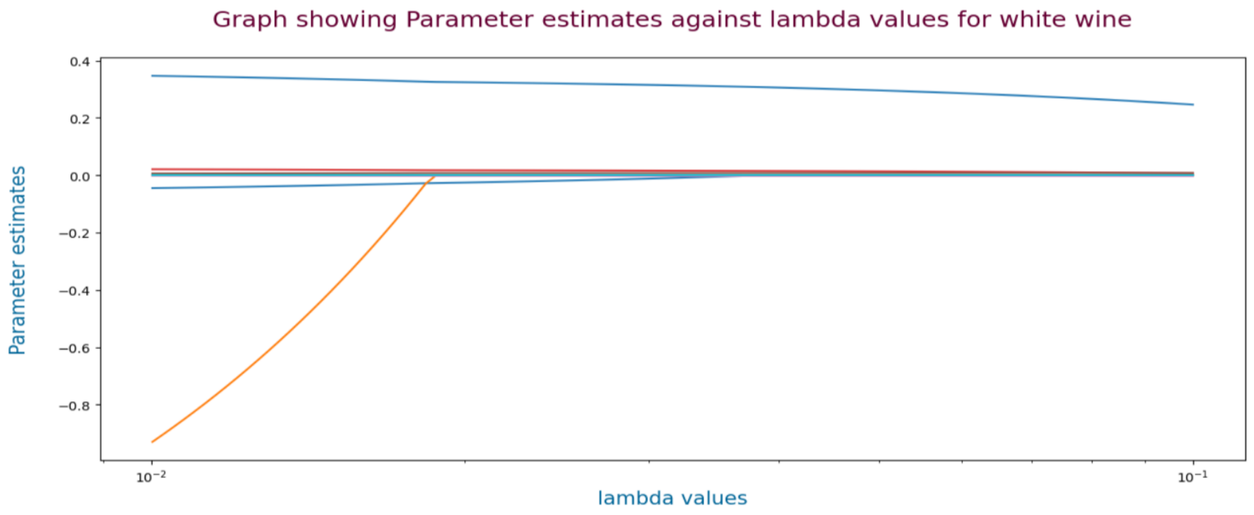
Results obtained:


Graph showing Mean Squared Error against lambda values for red wine


Graph showing Mean Squared Error against lambda values for white wine

- The graphs present the relationship between the regularization parameter (lambda) and the model's Mean Squared Error.
- Lambda controls the strength of regularization, and it is applied to prevent overfitting by penalizing the magnitude of the coefficients.
- The x-axis displays a range of lambda values in a log-scale to cover a broad range of regularization strengths.
- The y-axis represents the Mean Squared Error of the Lasso regression model.
- As lambda increases, the regularization strength becomes stronger and MSE decreases. This is because the models are becoming more constrained, and overfitting is reduced.


Graph showing Parameter estimates against lambda values for red wine


Graph showing Parameter estimates against lambda values for white wine

- As lambda increases, coefficients are penalized, and some are driven toward zero.
- The graphs present the relationship between the regularization parameter (lambda) and the model's feature correlation coefficients.
- Each point on the graph represents the average coefficients across different folds.

4.3.2.

To select features using Lasso, I used the LassoCV method to find the optimum lambda value and performed tuning of the lasso regression model. Following are steps I took:

- I used LassoCV when doing Lasso Regresyonde Tuning.
- I set the alpha parameters to range from log(0.001) to log(1).
- I set the number of Cross-Validation as 10.

- I found the alpha value of the Lasso model established with Cross-Validation and setup the Corrected Lasso model with this optimum alpha value.
- At the end, the coefficients that are not used in Lasso Regression and do not matter are equalize to zero.

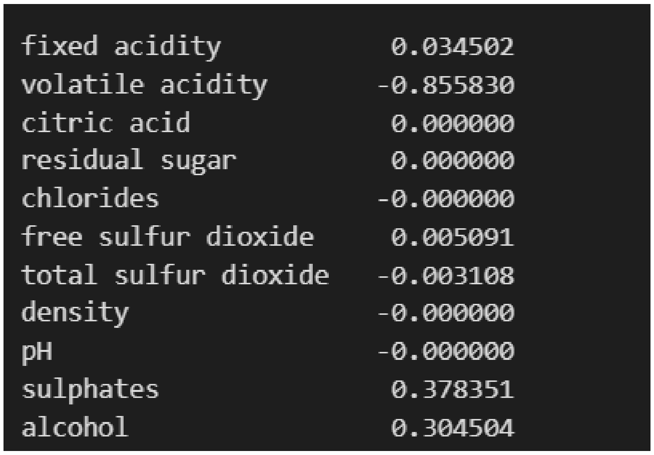
This method of feature selection is better than setting a threshold on the absolute correlation coefficient because :

1. Lasso can handle collinearity among predictors better than correlation coefficient.
2. Correlation coefficient may not detect some relevant features that have non-linear or non-monotonic relationship with the target variable.

[17] [18]

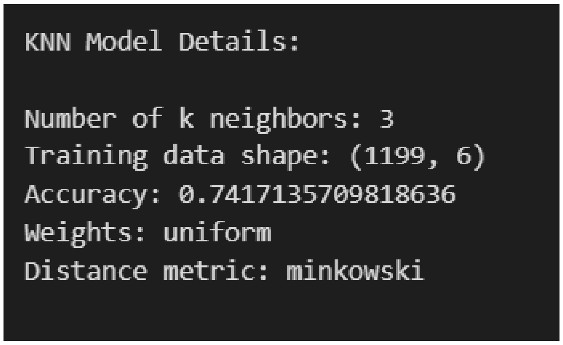
4.4.

Below are the correlation coefficients of the Lasso regression model that led to selecting **fixed acidity, volatile acidity, free sulfur dioxide, total sulfur dioxide, sulphates,** and **alcohol** as relevant variables:



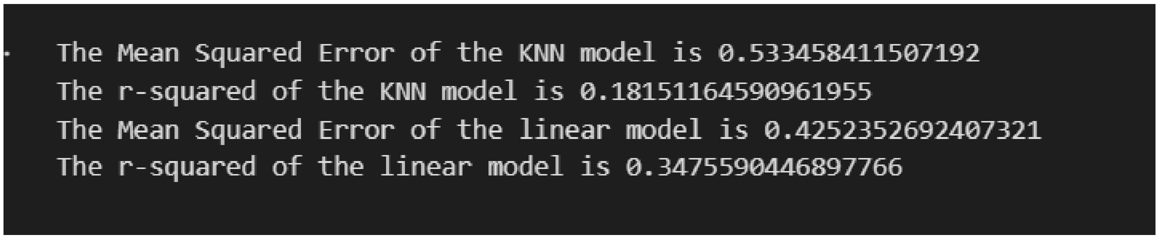
```
fixed acidity           0.034502
volatile acidity       -0.855830
citric acid             0.000000
residual sugar          0.000000
chlorides              -0.000000
free sulfur dioxide     0.005091
total sulfur dioxide   -0.003108
density                -0.000000
pH                     -0.000000
sulphates               0.378351
alcohol                 0.304504
```

Below is the KNN regression model built:



```
KNN Model Details:

Number of k neighbors: 3
Training data shape: (1199, 6)
Accuracy: 0.7417135709818636
Weights: uniform
Distance metric: minkowski
```

4.5.



```
The Mean Squared Error of the KNN model is 0.533458411507192
The r-squared of the KNN model is 0.18151164590961955
The Mean Squared Error of the linear model is 0.4252352692407321
The r-squared of the linear model is 0.3475590446897766
```

Given the performance of the two models based on Mean Squared Error and the Coefficient of determination, the Linear regression model perform better than the KNN model.

Advantages of linear regression

- Implementing Linear Regression is straightforward and interpreting the output coefficients is simpler.
- It is the preferred algorithm when there is a clear linear relationship between the independent and dependent variables, owing to its lower complexity compared to alternative algorithms.
- While Linear Regression is prone to over-fitting, this can be mitigated through the application of dimensionality reduction techniques, regularization methods (such as L1 and L2), and cross-validation.

Disadvantages of linear regression model

- In the linear regression method, outliers substantial influence on the regression, and the boundaries are linear.
- Linear regression assumes a linear association between dependent and independent variables, implying a straight-line relationship.
- Linear regression also examines the connection between the mean of dependent variables and independent variables, which is not a complete description of a single variable

Advantages of KNN:

- Training-Free Approach: KNN, known as a Lazy Learner or Instance-based learning, operates without a training period. Unlike algorithms such as SVM and Linear Regression, it swiftly processes real-time predictions without learning discriminative functions during training.
- Seamless Integration of New Data: With KNN, the addition of new data is seamless and does not compromise algorithm accuracy, thanks to its training-free nature.
- Simplicity in Implementation: Implementing KNN is uncomplicated, requiring only two parameters: the choice of K and the selection of a distance function (e.g., Euclidean or Manhattan).

Disadvantages of KNN:

- Challenges with Large Datasets: KNN's efficiency diminishes in large datasets due to the substantial computational cost associated with calculating distances for each new point.
- Struggles in High Dimensions: KNN encounters difficulties with high-dimensional data, making distance calculations across numerous dimensions a complex task.
- Necessity for Feature Scaling: Prior to applying KNN to a dataset, feature scaling (standardization and normalization) is essential to ensure accurate predictions.
- Vulnerability to Noise: KNN is sensitive to dataset noise, requiring manual handling of missing values and the elimination of outliers.

[19] [20]

[1] ‘What Is Nonlinear? Definition, Vs. Linear, and Analysis’, Investopedia. Accessed: Nov. 13, 2023. [Online]. Available: https://www.investopedia.com/terms/n/nonlinearity.asp

[2] ‘CMUdiaml9.pdf: Data, Inference, and Applied Machine Learning - DIAML’. Accessed: Nov. 13, 2023. [Online]. Available: https://canvas.cmu.edu/courses/36029/files/10293443?module_item_id=5635264

[3] ‘Quadratic Equations’. Accessed: Nov. 13, 2023. [Online]. Available: https://www.mathsisfun.com/algebra/quadratic-equation.html

[4] Stephanie, ‘Nonlinear Regression: Simple Definition & Examples’, Statistics How To. Accessed: Nov. 13, 2023. [Online]. Available: https://www.statisticshowto.com/nonlinear-regression/

[5] ‘Approximate entropy’, *Wikipedia*. Sep. 05, 2023. Accessed: Nov. 14, 2023. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Approximate_entropy&oldid=1173919176

[6] ‘Decision Tree’, GeeksforGeeks. Accessed: Nov. 14, 2023. [Online]. Available: https://www.geeksforgeeks.org/decision-tree/

[7] F. Albrecht, ‘Decision Trees for Classification — Complete Example’, Medium. Accessed: Nov. 14, 2023. [Online]. Available: https://towardsdatascience.com/decision-trees-for-classification-complete-example-d0bc17fcf1c2

[8] S. Ray, ‘8 Ways to Improve Accuracy of Machine Learning Models (Updated 2023)’, Analytics Vidhya. Accessed: Nov. 14, 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2015/12/improve-machine-learning-results/

[9] Y. Verma, ‘How to improve the accuracy of a classification model?’, Analytics India Magazine. Accessed: Nov. 14, 2023. [Online]. Available: https://analyticsindiamag.com/how-to-improve-the-accuracy-of-a-classification-model/

[10] S. Kumar, ‘A Simple Introduction to Validating and Testing a Model- Part 1’, Analytics Vidhya. Accessed: Nov. 14, 2023. [Online]. Available: https://medium.com/analytics-vidhya/a-simple-introduction-to-validating-and-testing-a-model-part-1-2a0765deb198

[11] ‘Decision Tree Advantages and Disadvantages | Decision Tree Regressor’, EDUCBA. Accessed: Nov. 20, 2023. [Online]. Available: https://www.educba.com/decision-tree-advantages-and-disadvantages/

[12] ‘CMUdiaml10.pdf: Data, Inference, and Applied Machine Learning - DIAML’. Accessed: Nov. 17, 2023. [Online]. Available: https://canvas.cmu.edu/courses/36029/files/10351830?module_item_id=5641608

[13] ‘K-Nearest Neighbors (KNN) Classification with scikit-learn’. Accessed: Nov. 17, 2023. [Online]. Available: https://www.datacamp.com/tutorial/k-nearest-neighbor-classification-scikit-learn#An%20Overview%20of%20K-Nearest%20Neighbors

[14] P. Sharma, ‘Understanding Distance Metrics Used in Machine Learning’, Analytics Vidhya. Accessed: Nov. 17, 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/02/4-types-of-distance-metrics-in-machine-learning/

[15] ‘What are the Advantages and Disadvantages of KNN Classifier?’, i2tutorials. Accessed: Nov. 17, 2023. [Online]. Available: https://www.i2tutorials.com/advantages-and-disadvantages-of-knn-classifier/

[16] ‘Advantages and Disadvantages of Logistic Regression’, GeeksforGeeks. Accessed: Nov. 17, 2023. [Online]. Available: https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/

[17] K. Kargın, ‘Lasso Regression Fundamentals and Modeling in Python’, Analytics Vidhya. Accessed: Nov. 19, 2023. [Online]. Available: https://medium.com/analytics-vidhya/lasso-regression-fundamentals-and-modeling-in-python-ad8251a636cd

[18] A. Gupta, ‘Feature Selection Techniques in Machine Learning (Updated 2023)’, Analytics Vidhya. Accessed: Nov. 19, 2023. [Online]. Available:

https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/

[19] ‘What are the Advantages and Disadvantages of KNN Classifier?’, i2tutorials. Accessed: Nov. 19, 2023. [Online]. Available: https://www.i2tutorials.com/advantages-and-disadvantages-of-knn-classifier/

[20] ‘ML - Advantages and Disadvantages of Linear Regression’, GeeksforGeeks. Accessed: Nov. 19, 2023. [Online]. Available: https://www.geeksforgeeks.org/ml-advantages-and-disadvantages-of-linear-regression/