**IDS Project**

****

**Data Analysis on User Knowledge Level Modeling Dataset**

Submitted By: Team-Alpha

**Group Members: - Course Instructor:**

Aman Agrawal - 20ucs017 Dr. Sakthi Balan Muthiah

Manan Badaya - 20ucs109 Dr. Aloke Dutta

Nimit Sethi - 20ucs129 Dr. Subrat Kumar Dash

Nipun - 20ucs131

# OBJECTIVE

To classify UNS i.e. Knowledge level of user, according to the attribute information in five different groups(STG i.e. the degree of study time for goal object materials, SCG i.e. the degree of repetition number of user for goal object materials, STR i.e. the degree of study time of user for related objects with goal object, LPR i.e. the exam performance of user for related objects with goal object, PEG i.e. the exam performance of user for goal objects) using different analysis methods and ML models on dataset and determine the best approach based on accuracy of each method.

**Problem Statement**

We have collected our dataset from the given website and the following steps have been performed on it.

Link to the Dataset: [User Knowledge Modeling](https://archive.ics.uci.edu/ml/datasets/User+Knowledge+Modeling)

1. Data preprocessing and its visualization
2. Explain all the inferences we got from the data
3. Explain what ML classification algorithms are being used and why
4. Implement those algorithms
5. Predict the accuracy of these algorithms.

# Introduction to the dataset

The user knowledge level modeling dataset was collected by Dr. Hamdi Tolga Kahraman consists of namely five attributes (STG, SCG, STR, LPR, PEG) which are used to classify the user’s knowledge level in four classes viz. Very low, Low, Middle, High.

The data set contains 4 classes with Very Low having 50 instances, Low with 129 instances, Middle with 122 instances, High with 130 instances. This gives a total of 403 data entries.

Rows - 403

Columns - 6

Attribute Information (Features):

1. STG (The degree of study time for goal object materials),

2. SCG (The degree of repetition number of user for goal object materials)

3. STR (The degree of study time of user for related objects with goal object)

4. LPR (The exam performance of user for related objects with goal object)

5. PEG (The exam performance of user for goal objects)

6. class:

– Very Low

– Low

– Middle

– High

E.g. (0.08,0.08,0.1,0.24,0.9) => High.

# **PREPROCESSING**

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always the case that we come across clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put it in a formatted way. So for this, we use data preprocessing tasks.

**Need for Preprocessing:**

Real-world data often contains noise, missing values, and is in an unsuitable format that cannot be used directly for machine learning models. Data preprocessing is a necessary task for cleaning data and preparing it for a machine learning model, which boosts the accuracy and efficiency of a machine learning model.

It includes beneath steps:

## Get the Dataset

The first thing we needed to develop a machine learning model was a dataset, because a machine learning model is entirely dependent on data. The dataset is the collection of data for a certain problem in the right format.

**In our code we have used a xlsx file.**

****

## Importing Libraries

Python predefined libraries must be imported to do data preparation using Python.

1. Pandas : Used for reading and framing the data.
2. Matplotlib : used for plotting the graph.
3. Train\_test\_split : used for split the data into two sets: training set and test set.
4. Seaborn : used for plotting heatmap , boxplots and pairplots.
5. Sklearn : scikit learn is probably the most useful library for machine learning in python . This library contains prediction algorithms like logistic regression , linear discriminant analysis , support vector macine etc.

## **Importing the Datasets**

Now we must import the datasets that we have collected for our machine learning research.

**Some Useful Information about Datasets**

1. **Shape parameter** to know the shape of the database i.e it will give number of columns and rows in the dataset.

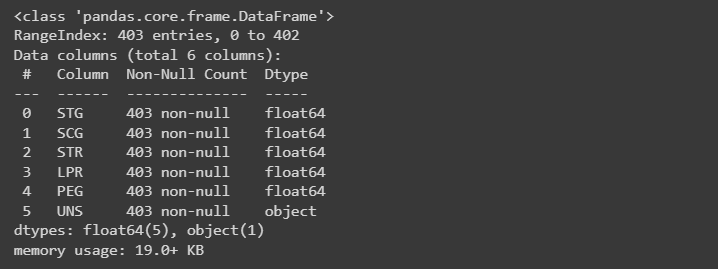


Output : 

means, our Dataset has 6 Columns and 403 rows

* **Info()** method : Used to get the information about our dataset i.e. how many attributes are there , the datatype of each attribute , and whether is there any null values to it or not. 

Output :



* **describe()** method: Pandas describe() is used to display some basic statistical information about a data frame or a collection of numerical values, such as percentile, mean, std, etc.



Output :



Each column's details are shown, including its mean value, standard deviation , aaa minimum and maximum values.

* **isnull()** method: This method is used to check whether our data has missing values or not. Missing values are common when no information is provided for one or more items.



Output:

Shape

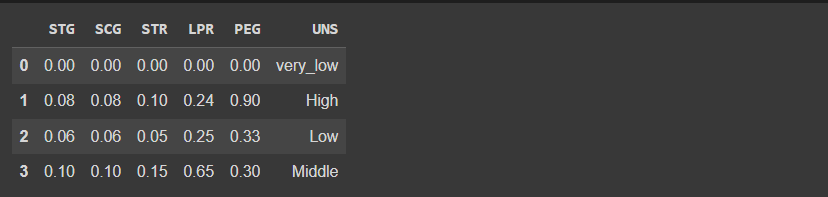
Description automatically generated with medium confidence

In our Dataset there are no null values

* **drop\_duplicates()** method: In Python, the Pandas drop duplicates() function assists in deleting duplicates from the Pandas Dataframe.



Output:



There are only four Unique knowledge level of users.

**Inference:**

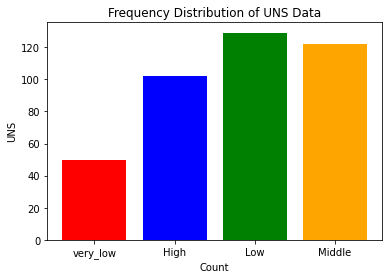
In our dataset, we can observe the following things:

1. All our data is numeric.
2. No missing, null, or duplicate values are present in the dataset.

Preliminary Analysis

1. Visualizing the target column

Target Column: **UNS (The knowledge level of user)**



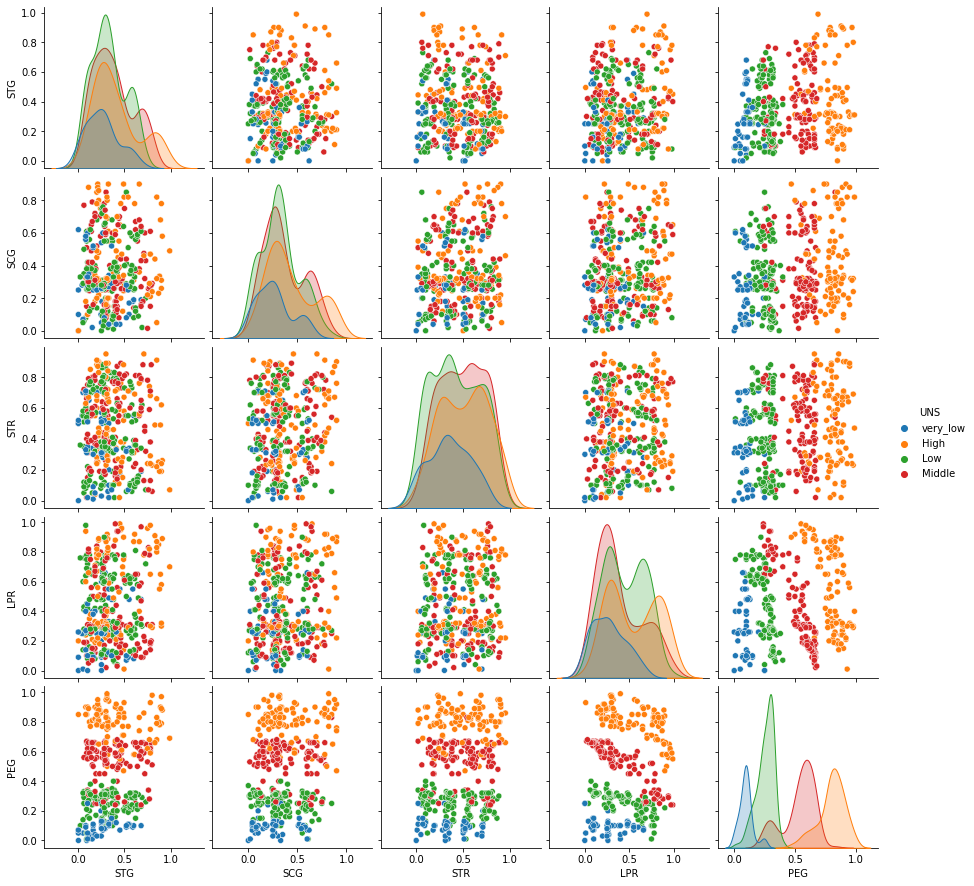
Let’s Visualize the target column using **Count plot** -

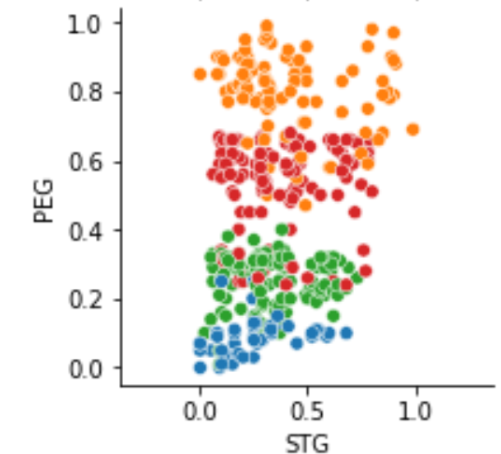
We have count on x-axis and UNS on y-axis. The count plot is used to display the frequency of all the classes. Using a count plot, imbalance in classes can be easily recognized.

1. **Pairplots**: Pairplot allows us to plot pairwise relationships between variables within a dataset. This creates a nice visualization and helps us understand the data by summarizing a large amount of data in a single figure.

Relation between Columns (**STG, SCG, STR, LPR, PEG**) using **pairplot.**



The Pairplot allows us to plot pairwise relationships between variables within a dataset. This creates a nice visualization and helps us understand the data by summarizing a large amount of data in a single figure.

**** ****

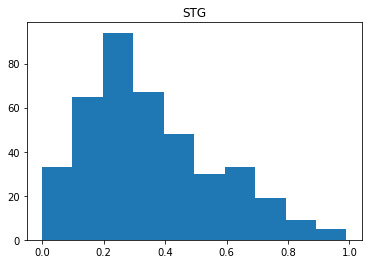
**Inferences:**

From the above plot, we can infer that – When classifying on basis of PEG and STG, high PEG implies High UNS and low PEG implies low UNS. They share a linear relationship.

Similarly, many more inferences can be made using other pair plots.

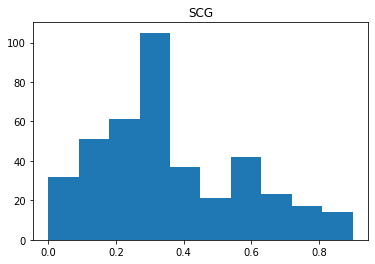
1. Histograms

A histogram, which divides a dataset into discrete, equal-sized intervals called bins, is the finest tool for visualizing the frequency distribution of a dataset. It also provides the location, spread and skewness of the data , and to visualize whether the distribution is symmetric or skewed left or right .



Inferences:

* This histogram implies that data is right skewed, mode is lies between 0.2 and 0.3 and data is not symmetric.



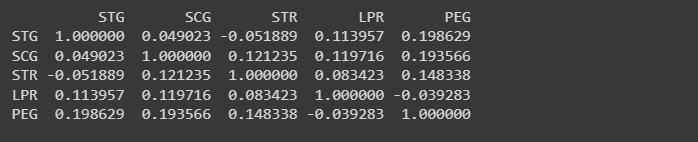


Similarly, from the above plot, we can see that –

1. Highest frequency of STG is between 80 and 100 which is between 0.2 and 0.3
2. Highest frequency of SCG is above 100 which is between 0.2 and 0.4
3. Highest frequency of STR is between 60 and 70 which is between 0.3 and 0.4
4. Highest frequency of LPR is between 80 and 90 which is between 0.2 and 0.3
5. Highest frequency of PEG is between 70 and 80 which is between 0.2 and 0.3
6. **Correlation Handling**

Correlation analysis is a statistical method used to measure the strength of the linear relationship between two variables. It calculates the level of change in one variable due to the change in the other. Researchers use correlation analysis to analyze quantitative data. As the correlation coefficient value goes towards 0, the relationship between the two variables will be weaker. The coefficient sign indicates the direction of the relationship; a + sign indicates a positive relationship, and a - sign indicates a negative relationship.

Pandas dataframe.corr() is used to find the pairwise correlation of all columns in the dataframe. Any NA values are automatically excluded. For any non-numeric data type columns in the dataframe it is ignored.



1. Heatmaps

Text

Description automatically generated

Chart, treemap chart

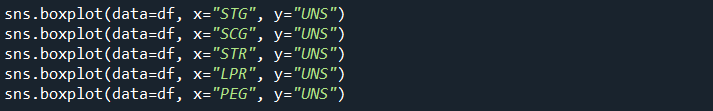
Description automatically generated

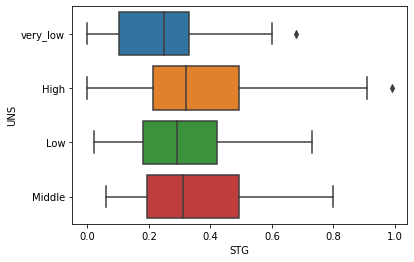
**Inferences:**

1. It shows that PEG and STG are highly correlated. In other words, there is a linear relationship between these two variables.
2. It shows the strength of relationship between both variables and is expressed numerically by the correlation coefficient. It ranges between +1 and -1.
3. **Box Plots**

Box plots are used to depict the distributions of numeric data values, particularly when comparing them across various groups. They are designed to convey high-level information at a glance, providing broad information about the symmetry, skew, variance, outliers etc. Detecting and dealing with outliers is one of the most important phases of data cleaning because these values differ from most of the data.

To show how the category value is distributed in relation to other numerical values, we may use boxplots.





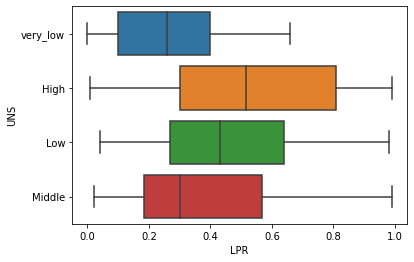
**Inferences:**

It shows the that very\_low and high class have outliers which differs from most of the data.

Chart, box and whisker chart

Description automatically generatedChart, box and whisker chart

Description automatically generated



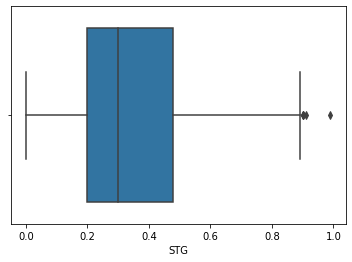
A picture containing text, clock

Description automatically generated

**Handling Outliers: -**

An outlier is an object that deviates greatly from the rest of the objects. They might be the result of a measurement or execution mistake. Outlier analysis or outlier mining is the process of analyzing outlier data.

For example Let's see boxplot of STG :

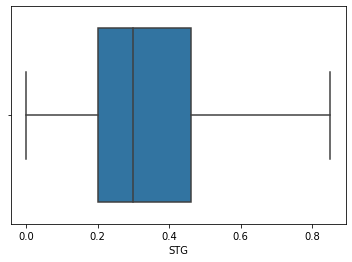
**We can see that the values above 0.85 are outliers**

Removing outliers: -

We will detect the outliers using IQR method and then remove them. In this method we remove all points that lie outside the range defined by the quartiles +/- 1.5\*IQR.

Text

Description automatically generated

Boxplot of STG after removing the outliers.

Chart

Description automatically generated Chart

Description automatically generated

Chart

Description automatically generated Chart

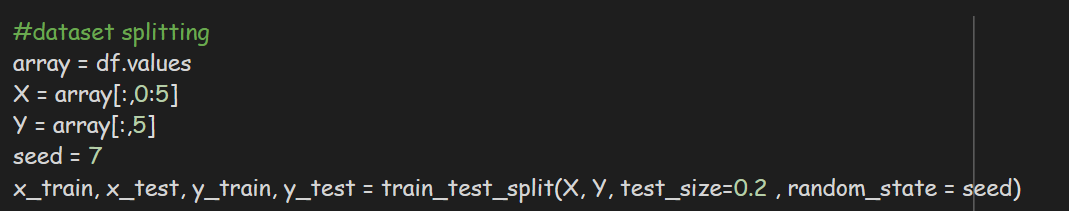
Description automatically generated

All other attributes show no outliers.

**Splitting of the data set in Training and Validation sets**

When preparing data for machine learning, we separate our dataset into a training set and a test set. This is a vital stage in the preparation of the data since it allows us to improve the functionality of our machine learning model.

Allocating 80% data to training dataset and 20% for testing purpose.



* **x\_train**: features for the training data
* **x\_test:** features for testing data
* **y\_train**: Dependent variables for training data
* **y\_test**: Independent variable for testing data
* **test\_size:**  for specifying the size of the test set
* **random\_state:** used to set a seed for a random generator so

that you always get the same result .

**ML classification**

Now, we will apply different ML algorithms of classification to find out which gives the best accuracy:

1. Logistic Regression: -

Logistic regression aims to solve classification problems. It does this by predicting categorical outcomes, unlike linear regression that predicts a continuous outcome. In our dataset we treat UNS as dependent variable and all other variables as independent variables and UNS is depending on all these variables. It transforms complex calculations around probability into a straightforward arithmetic problem.

**Assumptions:**

1. The variable must be independent of each other.
2. It also requires a significant sample size. This can be as small as 10 examples of each variable in a model. But this requirement goes up as the probability of each outcome drops.
3. Another assumption with logistic regression is that each variable can be represented using binary categories.

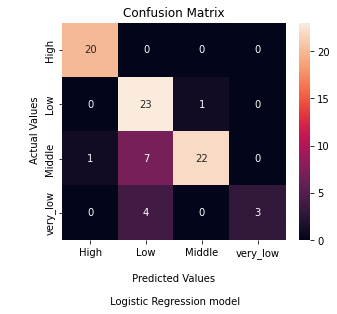
**Note: -**

Logistic regression, by default, is limited to two-class classification problems. But Some extensions like one-vs-rest can allow logistic regression to be used for multi-class classification problems, although they require that the classification problem first be transformed into multiple binary classification problems.









Graphical user interface, text

Description automatically generated

1. Decision Tree Classifier: -

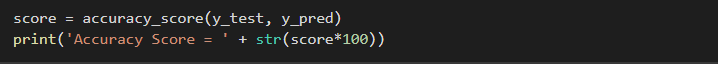
It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

Note: A decision tree can contain categorical data (YES/NO) as well as numeric data.

Diagram

Description automatically generated







Graphical user interface, application

Description automatically generated



1. **Gaussian Naïve Bayes Classifier**

A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It’s specifically used when the features have continuous values. It’s also assumed that all the features are following a gaussian distribution i.e, normal distribution.







Graphical user interface, application

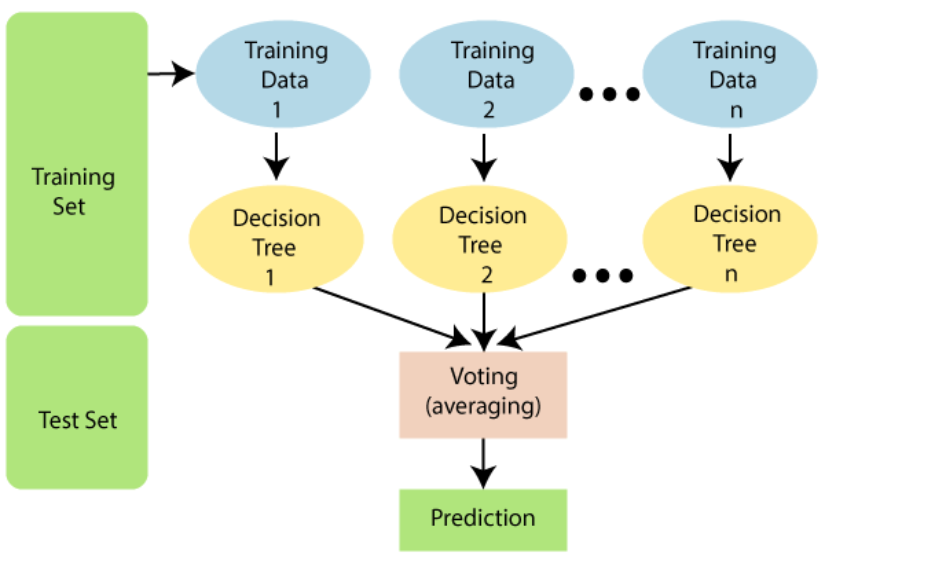
Description automatically generated



1. **Random Forest Classifier**

This classifier is based on the concept of **ensemble learning,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. ***Random* Forest is a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.**

The below diagram explains the working of the Random Forest algorithm:

****

****

****

****

**Graphical user interface, application

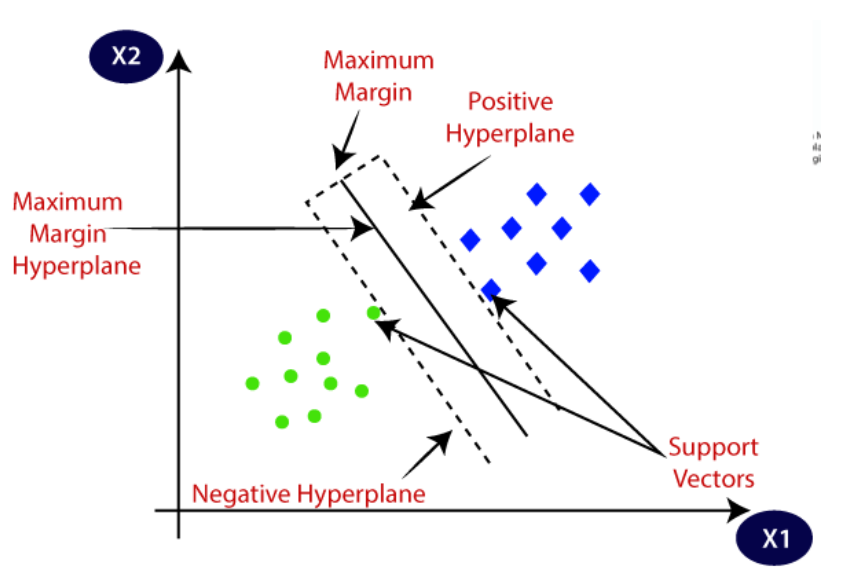
Description automatically generated**

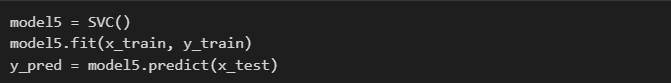
****

1. **Support Vector Machine:**

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. VM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane.

Note : This classifier is also a binary classifier but can be extended to multi class classifier.









Graphical user interface, application

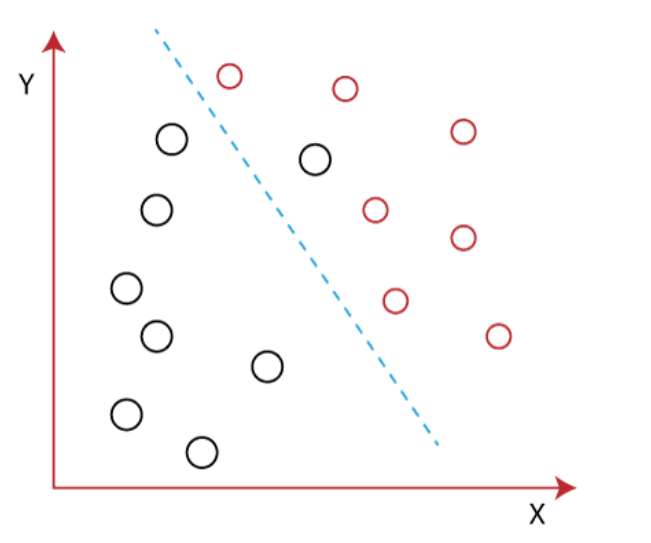
Description automatically generated



1. **Linear Discriminant Analysis**

Although the logistic regression algorithm is limited to only two-class, linear Discriminant analysis is applicable for more than two classes of classification problems.It is one of the most popular dimensionality reduction techniques used for supervised classification problems in machine learning*.*

Example:



It is impossible to draw a straight line in a 2-d plane that can separate these data points efficiently but using linear Discriminant analysis; we can dimensionally reduce the 2-D plane into the 1-D plane. Using this technique, we can also maximize the separability between multiple classes.







Graphical user interface, application

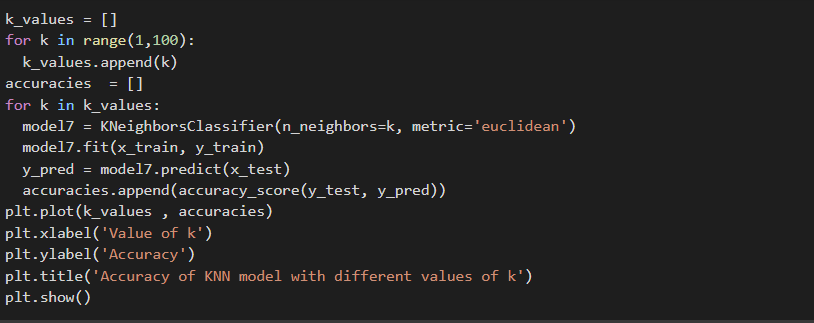
Description automatically generated



1. **KNN (K nearest Neighbor)**

K-nearest neighbours (KNN) algorithm uses ‘feature similarity’ to predict the values of new datapoints which further means that the new data point will be assigned a value based on how closely it matches the points in the training set. It is very useful for nonlinear data because there is no assumption about data in this algorithm. But It is computationally a bit expensive algorithm because it stores all the training data.

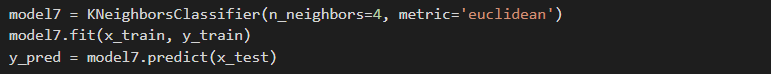
Finding the Best K value for our dataset:



Chart, line chart, scatter chart

Description automatically generated

Best value of k for our model is k = 4







Graphical user interface, application

Description automatically generated

Graphical user interface, text

Description automatically generated

**Evaluating different training models:**

1. Comparing Accuracies:

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| 1. Logistic Regression | 83.95% |
| 1. Decision Tree Classifier | 95.06% |
| 1. Gaussian Naïve Bayes | 83.95% |
| 1. Random Forest classifier | 91.35% |
| 1. Support Vector Machine | 93.82% |
| 1. Linear Discriminant analysis | 93.82% |
| 1. KNN classifier | 92.59% |

**Chart, bar chart

Description automatically generated**

1. **Comparing F1 Score:**

|  |  |
| --- | --- |
| **Model** | **F1 Score** |
| 1. Logistic Regression | 0.8395 |
| 1. Decision Tree Classifier | 0.9259 |
| 1. Gaussian Naïve Bayes | 0.8395 |
| 1. Random Forest classifier | 0.9382 |
| 1. Support Vector Machine | 0.9382 |
| 1. Linear Discriminant analysis | 0.9506 |
| 1. KNN classifier | 0.9259 |

**Chart, bar chart

Description automatically generated**

**Conclusion:**

Here Decision Tree Classifier have high accuracy and Linear Discriminant Classifier have high F Score.