Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur(Dumbarwadi), Tal. Junnar, Dist. Pune-412409

Assignment No : 1

- **Title :**

Predict the price of the Uber ride from a given pickup point to the agreed drop-off location. perform following tasks :

1) Pre process the dataset.
2) Identify outliers
3) check the correlation
4) Implement linear regression & random forest model.
5) Evaluate the models & compare their respective scores like R2, RMSE, etc.

- **Objective :**

- To predict the fare amount of an Uber ride based on location & time date.
- To perform data preprocessing & outlier detection.
- To implement linear regression & random forest regression models.
- To evaluate model performance using R^2, RMSE & MAE.

- **Problem statement :**

To build a machine learning model that can be accurately predict Uber ride fares using features such as pickup / drop-off coordinates & time. Dataset is taken from kaggle & consists of past Uber ride data.

- Software & Hardware Requirements :
  - Software : python 3.x, Jupyter Notebook, Numpy, Pandas
  - Hardware : standard PC / laptop with minimum 4GB RAM & internet Access.

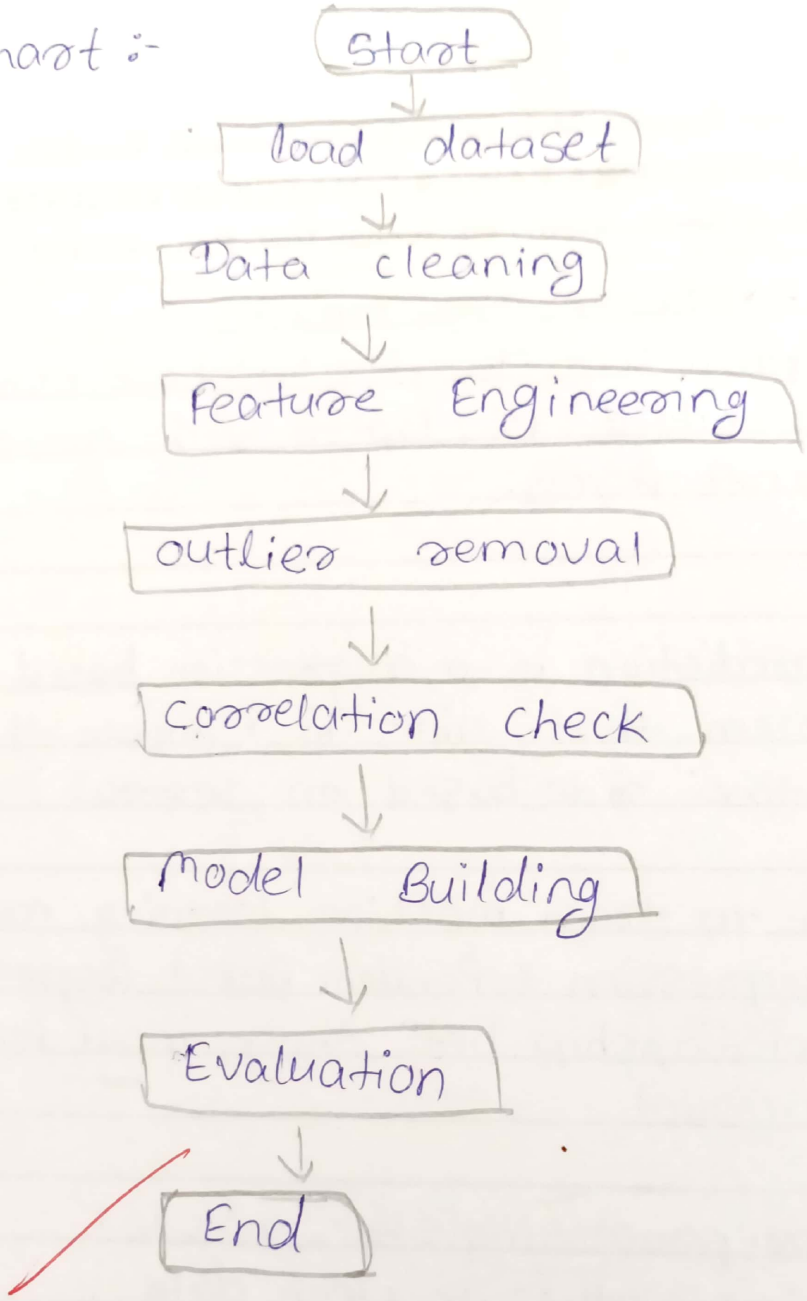
- Theory :
  Uber fare prediction is a regression based machine learning problem that aims to estimate the fare amount of a taxi ride based on several input factors.

- To core idea to train machine learning models such as linear Regression & Random forest Regression to learn the relationship betn these input feature & output fare amount.

- Important preprocessing steps include :
  - Handling missing values to clean data.
  - Datetime feature extraction like hours, day, month
  - calculating the distance betn using Haversine.
  - outlier detection & removal using IQR.
  - correlation Analysis to identify which features influence

- The models are trained & tested using train-test-split & then evaluate using matrix like :
  - R^2, RASE (Root mean squared Error), MAE (mean absolute error) : measures how well prediction.

Flowchart :-



- Algorithm :

1) load dataset
2) clean dataset : handling missing values remove outliers
3) Feature engineering : calculate distance using Haversine
4) check correlation
5) split data into training & testing.
6) Train models : linear Regression & Random forest
7) Evaluate models using : R^2, RMSE, MAE.

- Flowchart :

Start → load dataset → Data cleaning → Feature Engg → outlier removal → correlation check → model Building → Evaluation → End.

- Test Data set :

- Input : Pickup_longitude, pickup_latitude, droopoff_longitude, droopoff_latitude, passenger_count.
- output : fare_amount (predicted).

- Haversine formula :

$$distance = 2.R.\arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1).\cos(\phi_2).\sin^2\left(\frac{\Delta\lambda}{2}\right)}\right)$$

where, ϕ : latitude , λ : longitude , $R = 6371 R = 6371$ km

- Preprocessing Done :

- Remove null, & zero values, negative fare & Passenger Coun
- Add new colum distance & Drop irrelevant columns.

- outlier detection : use IQR method to remove outliers.

- correlation check :
  - High correlation found betⁿ fare_amount & distance
  - passenger count had weak correlation.

- Model Implementation

1) Linear Regression :
   - Assumes linear relationship
   - sensitive to utliers.

2) Random Forest Regressor :
   - ~~enab~~ ensemble of decision trees
   - Handles non linearity & outliers better
   - Higher accuracy in our case.

- Conclusion | Analysis
  - Distance calculated using Haversine formula was the most important feature.
  - Random forest outperformed linear regression in term of accuracy.
  - outlier removal & feature engg. significantly improved results.
  - final model is useful for fare prediction in ride sharing application like Uber.

Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur (Dumbarwadi), Tal. Junnar, Dist. Pune - 412409

Questions :

1) What is the objective of this Uber fare prediction assignment ?

→ The objective is to build a ML model that can be accurately predict the fare amount of an Uber ride based on various feature like pickup & dropoff location date, time, & passenger count.

2) Which is the target column in your model ?

→ The target column is 'fare amount'.

3) What kind of preprocessing did you perform on dataset

→ Handling missing values, remove outliers, extracted feature from datetime & calculate distance.

4) How did you handle missing or null values ?

→ Rows with null values were dropped using dropna().

5) What are outliers ?

→ outliers are extreme values that deviate significantly from other observations in the dataset.

Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur (Dumbarwadi), Tal. Junnar, Dist. Pune - 412409

6) What is correlation?
→ Correlation is a statistical measures that express the extent to which two variables are linearly related.

7) Explain how linear Regression works?
→ It fits a line that minimizes the difference betⁿ actual & predicted values using the least squares method.

8) What is Random forest Regressor?
→ An ensemble model that uses multiple decisions trees & averages their predictions for better accuracy.

9) Which model performed better in your results & why?
→ Random forest performed better due to its ability to model non linear relationships & handle variance

10) What metrics did you use to evaluate your models?
→ R^2 score, RMSE (Root Mean Squared Error), & MAE (Mean Absolute Error).

Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur (Dumbarwadi), Tal. Junnar, Dist. Pune - 412409

Assignment No : 2

* Title : K-nearest Neighbors & support vector machine.

* Problem statement : classify the email using the binary classification method. email spam detection has two states : a) normal state - Not spam. b) Abnormal state - Spam. Use K-Nearest Neighbors & Support vector machine for classification. Analyze their performance. Dataset link : The emails.csv dataset on the kaggle.

* ~~Prerequ~~ prerequisities : understand distance metrics like Euclidean distance for measuring similarity betn data points.

Theory :

The K nearest neighbors algorithm, also known as KNN, is a non-parametric, supervised learning classified which uses proximity to make classification or predictions about the grouping of an individual data point.

- while it can be used for either regression or classification problem it is typically used as a

classification algorithm, working off the assumption that similar points can be found near on another.

- KNN algorithm assumes the similarity betⁿ the new cases / data & available cases & puts the new case into the category that is most similar to available categories.

- KNN algorithm stores all the available data & classifies a new data point based on the similarity. KNN is non parametric algorithm which means does not make any assumption on underlying data.

KNN algorithm :

1) Get the value of k.
2) Consider all points & the new points in a n-dimensional space.
3) calculate the distance of new points from all points.
4) Sort the distance of all point.
5) Sort the distance of all point & select k point which smallest distance.
6) estimate the value of test point by weighted

average of its neighbor

9) Is the error of test points satisfying then end, if not then start from step 1.

Support vector machine:

- Support vector machine (svm) is supervised learning algorithm which is used for classification as well as Regression problems.

- The goal of 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 point in the correct category in the future. This is best decision boundary is called a hyperplane.

- Svm choose the extreme points / vectors that help in creating the hyperplane. These extreme cases are called support vectors, & hence the algorithm is termed as support vector machine.

Dataset Description:

This is a file containing related information 5172 randomly picked email files & their respective labels for spam, not spam classification. There are 3002 colums. last colum has the labels for

Prediction : 1 for spam, 0 for not spam.

Algorithm :

1) Importing libraries
2) Importing datasets
3) finding missing data
4) Encoding categorical Data
5) Spliting Dataset into training & test set.
6) Applying classifiers KNN & SVM for classification of spam & not spam.
7) Evaluating performance of classifier in terms of metrics such as RMSE, R^2 score etc.

Conclusion :

we have understand how to apply KNN & SVM algorithm on the dataset.

Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur (Dumbarwadi), Tal. Junnar, Dist. Pune - 412409

Assignment No : 3

Title : Prediction of Bank customer churn using Neural Network classifier.

Problem Statement : The ~~ob~~ Given 'a bank customer build a neural network based classifier that can determine whether they will leave or not in the next 6 months:

perform the following steps:

1) Read the dataset
2) Distinguish the feature & target set & divide the data set into training & test sets.
3) Normalize the train & test data.
4) Initialize & build the model. Identify the points of improvement & implement the same
5) Print the accuracy score & confusion matrix.

Prerequisites : before implementing the model, the following knowledge & tools are required:

- python programming
- Libraries : NumPy, Pandas, Matploblib, Scikit learn Tensorflow / Keras.
- Understanding the concepts like :
  - Data preprocessing
  - Feature Scaling / Normalization.

Theory :

Dataset Description :

Dataset contains 10000 records & 14 Features. such as : customer ID, Surname : unique identifiers. credit Score, Age, Tenure, Balance, Num of Products, HasCrCard, IsActiveMember, Estimated Salary : Numerical features.
Geography, Gender : categorical features
Excited : Target variable (1 → customer left the bank, 0 → customer stayed).

Data preprocessing steps :

a) Reading the Dataset : The dataset is imported using pandas, & unnecessary columns are removed.

b) Feature & Target separation :
- Feature set (x) : All independent variables that describe customer behavior.
- Target set (y) : The dependent variable excited.

c) Train-Test split : The dataset is split into training (80%) & testing (20%) sets using train_test_split() from scikit-learn.

d) Normalization (Feature Scaling) : Neural networks requires data on a similar scale.

Neural Network Model Building:

Model initialization: A sequential model from keras is used to build a feed forward Artificial Neural Network (ANN).

Model architecture:

Input layer: Takes the number of input features.

Hidden layers: one or more fully connected layers with activation funcns.

output layer: A single neuron with sigmoid activation for binary classification.

Training: The model is using training data for several epochs & batch size.

Model evaluation

After training, predictions are made on the test data. model performance is evaluated using: Accuracy score, confusion matrix.

Conclusion:

In this experiment, a neural network classifier was successfully developed to predict bank customer churn. The model achieved good accuracy after proper preprocessing, normalization & tuning.

Shri Gajanan Maharaj Shikshan Prasarak Mandal's
SHARADCHANDRA PAWAR COLLEGE OF ENGINEERING
Otur (Dumbarwadi), Tal. Junnar, Dist. Pune - 412409

Assignment No : 4

Title : K - Neigbest Neighbors Algorithm.

Problem Statement : Implement KNN algorithm on diabetes.csv dataset. Compute confusion matrix, accuracy, error rate, precision & recall on the given dataset.

Prerequisites :

- Understanding the fundamental concepts of Supervised learning.
- Understanding of distance metrics to measure the similarity betn data points. Distance metrics like Euclidean distance & manhattan distance which are commonly used in KNN
- concept of confusion matrix.

Theory :

KNN is supervised learning technique. KNN algo. assumes the similarity betn the new case (data) & available cases & put the new case into the category that is most similar to the available categories. KNN is a non parametric algo. It is also called a lazy learner algorithm.

KNN Algorithm

Step 1: Select the number K of the neighbors.
Step 2: Calculate the Euclidean distance of K number of neighbors.
Step 3: Take the K nearest neighbors, count the numbers of the data points in each category.
Step 4: Assign the new data points to that category for which the number of the number of the neighbor is maximum.
Step 5: our model is ready.

Confusion matrix:

A confusion matrix is a performance measurement tool used in machine learning to evaluate the quality of a classification model. It provides a summary of a model's prediction & the actual outcomes in a tabular format, allowing you to assess how well the model performed for different classes.

		predicted class: Positive	predicted class: negative
actual class	Positive	TP	FN
actual class	negative	FP	TN

fig. Confusion matrix.

- True Positive (TP): The model correctly predicted instance of positive class.
- True Negative (TN): the model correctly predicted instance of negative class.
- False positive (FP): The model incorrectly predicted instance as the positive class when they were actually the negative class.
- False Negative (FN): The model incorrectly instence as the negative class when they were actually positive class.

Dataset Description

This dataset is originally from the National institute of Diabetes & Digestive & Kidney Disease. The objective is to predict based on diagnotis measurements whether a patient has diabetes. Several constraints were placed on the selection of these instances from a larger database.

Conclusion :

Thus we have learned KNN is often used for simple classification problems & serves as a baseline model. A confusion matrix provides a more detailed view of a classification model's performance compared to a single accuracy score.

Assignment No: 5

Title : K-Means clustering algorithm

Problem statement : Implement K means clustering / hierarchical clustering on sales-data sample. CSV dataset. Determine the number of clusters using the elbow method.

Prerequisites : understanding the basic of Kmeans clustering & Hierarchical clustering.

Theory :

- K-means clustering is a widely used unsupervised machine learning algorithm that divided a dataset into K, distinct, non-overlapping clusters. It aims to group similar data points together & separate dissimilar data points.

- Kmeans Clustering is an unsupervised learning algorithm that is used to solve the clustering problem in machine learning or data science. In this topic we will learn what K means clustering algorithm is how the algorithm works, along with the python implementation of K-means clustering.

- It allows us to cluster the data into different groups & a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training. It is centroid based algo. where each cluster in associated with a centroid The main aim of this algo. is to minimize the sum of distances beth the data point & their corresponding clusters.

- The K means clustering algo mainly performs two tasks:
  - Determines the best value for K center points or centroids by an iterative process.
  - Assigns each data point to its closed K-center. Those data points which are near to the particular K-center, create a cluster.

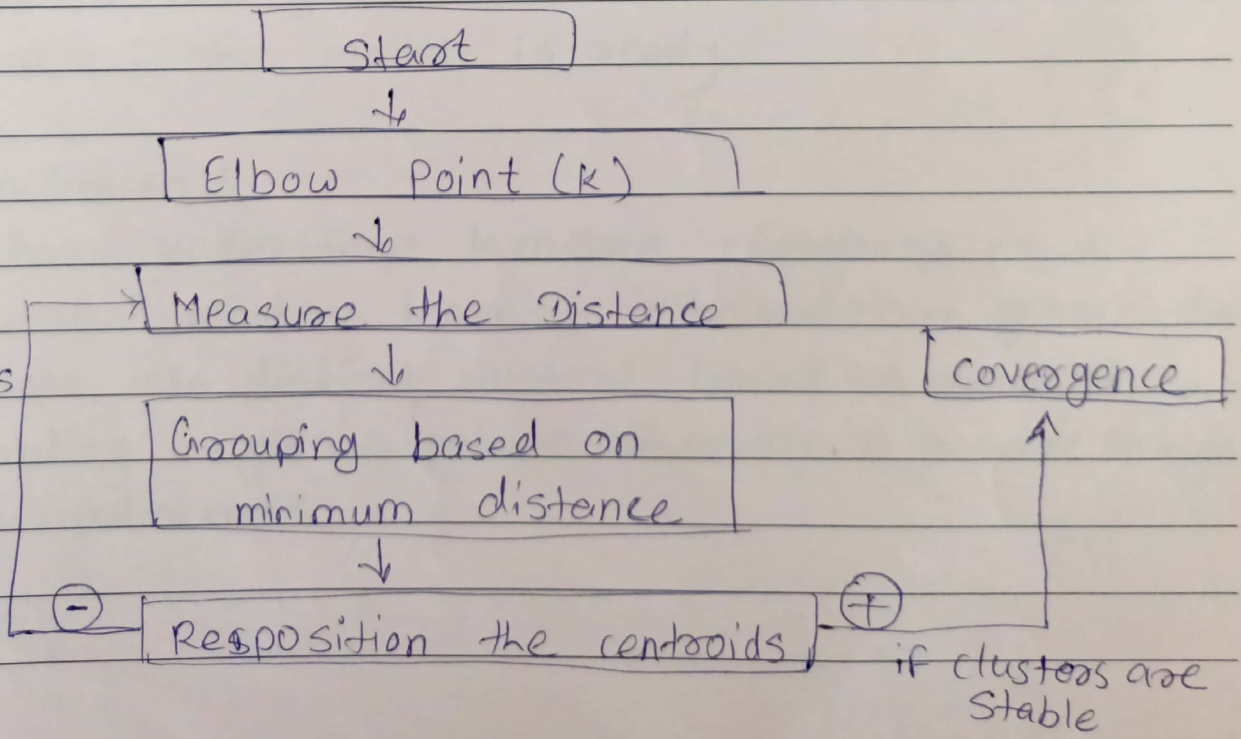


Fig. K Means clustering working.

Algorithm of k means clustering :

- Step 1 : select the number k to decide the number of sete clusters.

$$WSS = \sum_{i=1}^{m} (x_i - c_i)^2$$

- Step 2 : select random k points or centroids.
- Step 3 : Assign each data point to their closedt centroid, which will form the predefined K clusters.
- Step 4 : Calculate the variance & place a new centroid of each cluster.
- Step 5 : Repeat the third steps, which means resign each datapoint to the new closet centroid of each cluster.
- Step 6 : If any reassignment occurs then go to step 4 else go to FINISH.
- Step 7 : The model is ready

conclusion :

we have understand k means clustering is a powerfull machine learning technique that groups data points into distinct clusters based on similarity, enabling insights & pattern discovery in a wide range of applications.