LP-III Machine Learning (2022-23)	
Assignment 1: Implement Gradient Descent Algorithm to find the local minima of a function.	
Student Name:	**Roll No. :**
Batch:	**Division :**

Assignment 1: Implement Gradient Descent Algorithm to find the local minima of a function. For example, find the local minima of the function $y=(x+3)^2$ starting from the point x=2.

Title: Implement Gradient Descent Algorithm to find the local minima of a function. For example, find the local minima of the function $y=(x+3)^2$ starting from the point x=2.
Aim: Find local minima for given function.

Prerequisites: Gradient, Cost Function, Differential and Convex Function

Theory:

Gradient descent is an optimization algorithm that's used when training a machine learning model. It's based on a convex function and tweaks its parameters iteratively to minimize a given function to its local minimum.

A gradient simply measures the change in all weights with regard to the change in error. You can also think of a gradient as the slope of a function. The higher the gradient, the steeper the slope and the faster a model can learn. But if the slope is zero, the model stops learning. In mathematical terms, a gradient is a partial derivative with respect to its inputs.

Imagine you have a machine learning problem and want to train your algorithm with gradient descent to minimize your cost-function $J(w, b)$ and reach its local minimum by tweaking its parameters (w and b). The Figure 1 below shows the horizontal axes representing the parameters (w and b), while the cost function $J(w, b)$ is represented on the vertical axes. Gradient descent is a convex function.

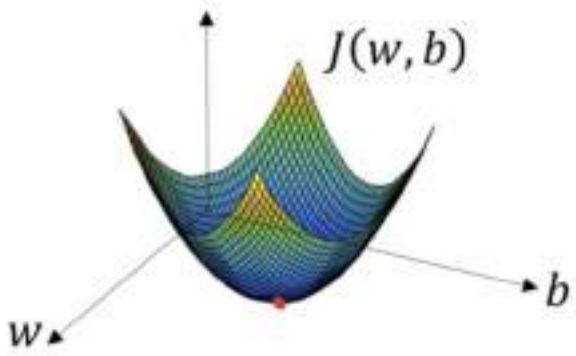

Figure 1: Cost Function J(w, b) mapped against w and b.
(image courtesy: https://builtin.com/data-science/gradient-descent)

We know we want to find the values of w and b that correspond to the minimum of the cost function. To start finding the right values we initialize w and b with some random numbers.

Gradient descent then starts at that point (somewhere around the top of our illustration), and it takes one step after another in the steepest downside direction (i.e., from the top to the bottom of the illustration) until it reaches the point where the cost function is as small as possible.

How big the steps the gradient descent takes into the direction of the local minimum are determined by the learning rate, which figures out how fast or slow we will move towards the optimal weights. For gradient descent to reach the local minimum we must set the learning rate to an appropriate value, which is neither too low nor too high. This is important because if the steps it takes are too big, it may not reach the local minimum because it bounces back and forth between the convex function of gradient descent. If we set the learning rate to a very small value, gradient descent will eventually reach the local minimum but that may take a while.

For given convex function, $(x+3)^2$ with start point as 2.
$(x+3)^2 = x^2 + 6x + 9$
Learning Rate = 0.1 or any other values
Number of Maximum Iteration = 100

Algorithm:

1. Choose a starting point (initialisation)

   start = 2

2. Calculate gradient at this point.

   diff = Learning Rate * Gradient(x)

   $x_{new} = x_{old}$ - diff

3. Make a scaled step in the opposite direction to the gradient (objective: minimise)
4. Repeat points 2 and 3 until one of the criteria is met:
   - maximum number of iterations reached
   - step size is smaller than the tolerance (due to scaling or a small gradient).

Conclusion: Using concept of Gradient Descent Algorithm, we have found local minima for $(x+3)^2$ with start point as 2.

Questions:

1. What is Local minima.

2. What is Gradient.

3. What is the effect of smaller and larger learning rate.

4. What is Stochastic Gradient Descent (SGD).

5. List out different optimizers?

LP-III Machine Learning (2022-23)	
Assignment 2: 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 and Support Vector Machine for classification. Analyse their performance.	
Student Name:	**Roll No. :**
Batch:	**Division :**

Assignment 2:
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 and Support Vector Machine for classification. Analyze their performance.

Title: Implement KNN and SVM classification algorithm to predict Normal and Abnormal emails using dataset available at https://www.kaggle.com/datasets/balaka18/email-spam-classification-dataset-csv.

Aim: Predict and Analyse Results of KNN and SVM algorithm for Classification.

Prerequisites: Binary Classification, KNN, SVM.

Theory:

K-nearest Neighbours (KNN) Algorithm:

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

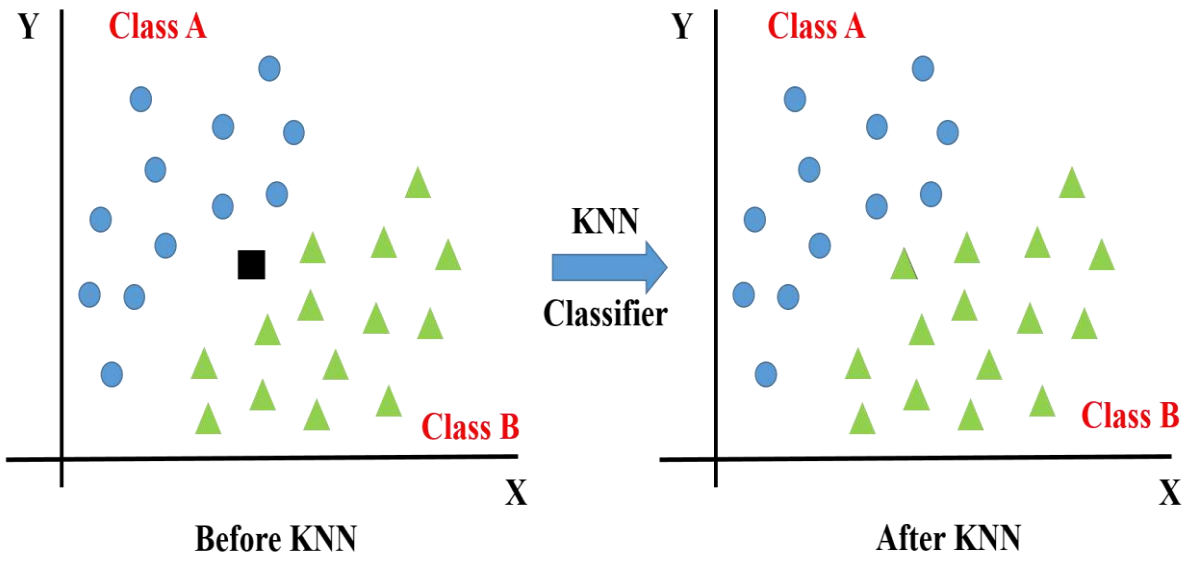

Figure 1: Visualization of KNN.

The impact of selecting a smaller or larger K value on the model

Larger K value: The case of underfitting occurs when the value of k is increased. In this case, the model would be unable to correctly learn on the training data.

Smaller k value: The condition of overfitting occurs when the value of k is smaller. The model will capture all of the training data, including noise. The model will perform poorly for the test data in this scenario.

When the problem statement is of 'classification' type, KNN tends to use the concept of "Majority Voting". Within the given range of K values, the class with the most votes is chosen. When the problem statement is of 'regression' type, KNN employs a mean/average method for predicting the value of new data. Based on the value of K, it would consider all of the nearest neighbours. The algorithm attempts to calculate the mean for all the nearest neighbours' values until it has identified all the nearest neighbours within a certain range of the K value.

Algorithm:

The K-NN working can be explained on the basis of the below 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 as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

Step-6: Our model is ready.

Support Vector Machine (SVM) Algorithm:

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. 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**. SVM 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**.

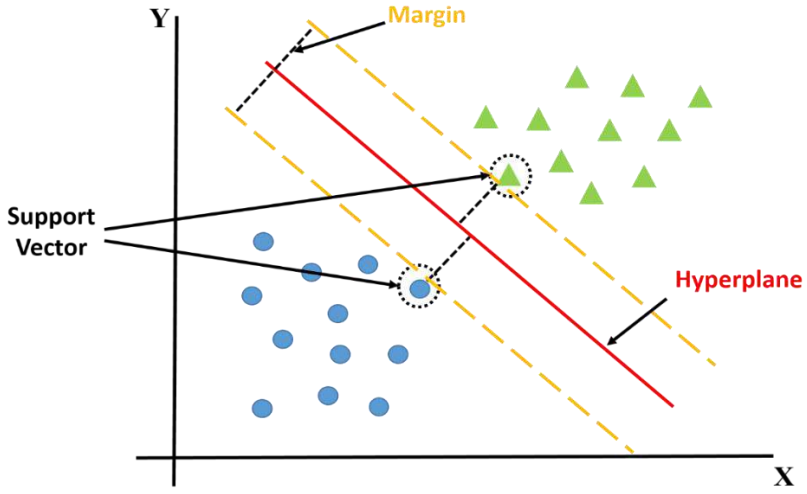

Figure 2: Terminologies in SVM.

There are two types of Support Vector Machines:

1. **Linear SVM or Simple SVM:** Linear SVM is used for linearly separable data. If a dataset can be classified into two classes with a single straight line, then that data is considered to be linearly separable data, and the classifier is referred to as the linear SVM classifier. It is typically used for linear regression and classification problems.

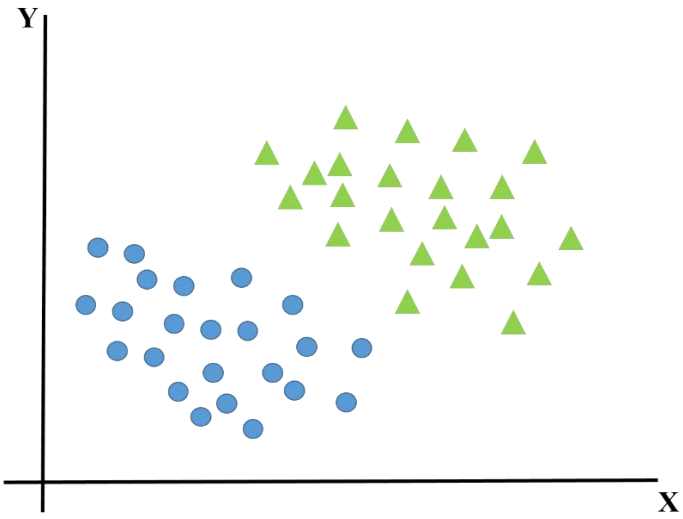

Figure 3: Linearly Separable Dataset.

2. **Nonlinear SVM or Kernel SVM:** Nonlinear SVM is used for nonlinearly separated data, i.e., a dataset that cannot be classified by using a straight line. The classifier used in this case is referred to as a nonlinear SVM classifier. It has more flexibility for nonlinear data because more features can be added to fit a hyperplane instead of a two-dimensional space.

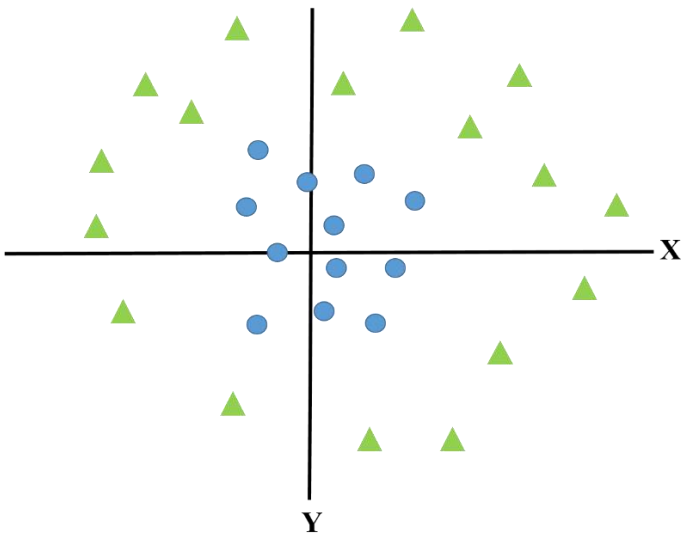

Figure 4: Linearly Non-Separable Dataset.

Conclusion: Using concept of KNN and SVM classification algorithms, we have classified emails into two class normal (non-spam) and abnormal (spam) and compared both, KNN and SVM, using evaluation metrics.

Questions:

1. What is decision boundary?
2. Explain train_test_split() function in detail.
3. What is the significance of MSE and MAE.
4. Explain parameters used for SVC.
5. List out Applications of KNN and SVM?

LP-III Machine Learning (2022-23) **Assignment 3:** Implement K-Nearest Neighbors algorithm on diabetes.csv dataset. Compute confusion matrix, accuracy, error rate, precision and recall on the given dataset.	
Student Name:	**Roll No. :**
Batch:	**Division :**

Assignment 2:
Implement K-Nearest Neighbors algorithm on diabetes.csv dataset. Compute confusion matrix, accuracy, error rate, precision and recall on the given dataset.

Title: Implement KNN classification algorithm to predict diabetes person using dataset available at https://www.kaggle.com/datasets/abdallamahgoub/diabetes.

Aim: Predict and Analyse Results of KNN algorithm for Classification.

Prerequisites: KNN.

Theory:

K-nearest Neighbours (KNN) Algorithm:

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

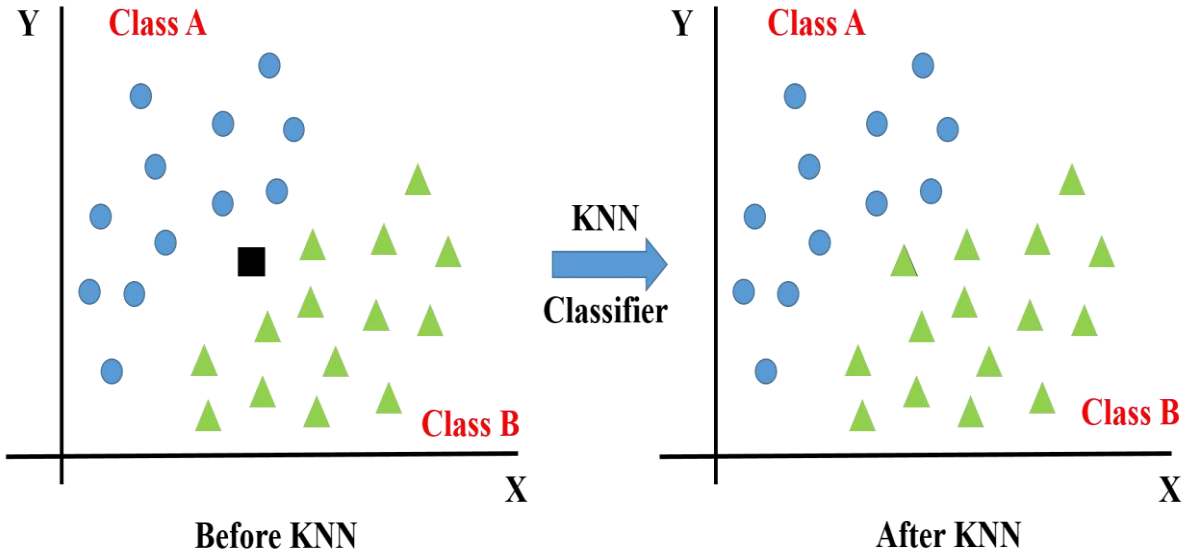

Figure 1: Visualization of KNN.

The impact of selecting a smaller or larger K value on the model

Larger K value: The case of underfitting occurs when the value of k is increased. In this case, the model would be unable to correctly learn on the training data.

Smaller k value: The condition of overfitting occurs when the value of k is smaller. The model will capture all of the training data, including noise. The model will perform poorly for the test data in this scenario.

When the problem statement is of 'classification' type, KNN tends to use the concept of "Majority Voting". Within the given range of K values, the class with the most votes is chosen. When the problem statement is of 'regression' type, KNN employs a mean/average method for predicting the value of new data. Based on the value of K, it would consider all of the nearest neighbours. The algorithm attempts to calculate the mean for all the nearest neighbours' values until it has identified all the nearest neighbours within a certain range of the K value.

Algorithm:

The K-NN working can be explained on the basis of the below 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 as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

Step-6: Our model is ready.

Conclusion: Using concept of KNN classification algorithms, we have classified diabetes person into two classes and evaluated KNN algorithm using evaluation metrics.

Questions:

1. What is confusion matrix?

2. Explain accuracy and error rate?

3. What is the significance of precision?

4. Explain Recall and F-1 Score?

5. Explain: 1. head() 2. shape 3. isnull() 4. drop()?

LP-III Machine Learning (2022-23) **Assignment 4:** Implement K-Means clustering/ hierarchical clustering on sales_data_sample.csv dataset. Determine the number of clusters using the elbow method.	
Student Name:	**Roll No. :**
Batch:	**Division :**

Assignment 4:
Implement K-Means clustering/ hierarchical clustering on sales_data_sample.csv dataset. Determine the number of clusters using the elbow method.

Title: Implement K-Means clustering/ hierarchical clustering on sales_data_sample.csv dataset using dataset available at https://www.kaggle.com/datasets/kyanyoga/sample-sales-data.

Aim: Implement K-Means clustering/ hierarchical clustering.

Prerequisites: Linear Regression, Random Forest, Decision Tree.

Theory:

Clustering:

Problem involves assigning the input into two or more clusters based on feature similarity. Similar groups based on their interests, age, geography, etc can be done by using Unsupervised Learning algorithms. A way of grouping the data points into different clusters, consisting of similar data points. It does it by finding some similar patterns in the dataset such as shape, size, color, behavior, etc., and divides them as per the presence and absence of those similar patterns. It is an unsupervised learning method, hence no supervision is provided to the algorithm, and it deals with the unlabeled dataset. After applying this clustering technique, each cluster or group is provided with a cluster-ID. ML system can use this id to simplify the processing of large and complex datasets.

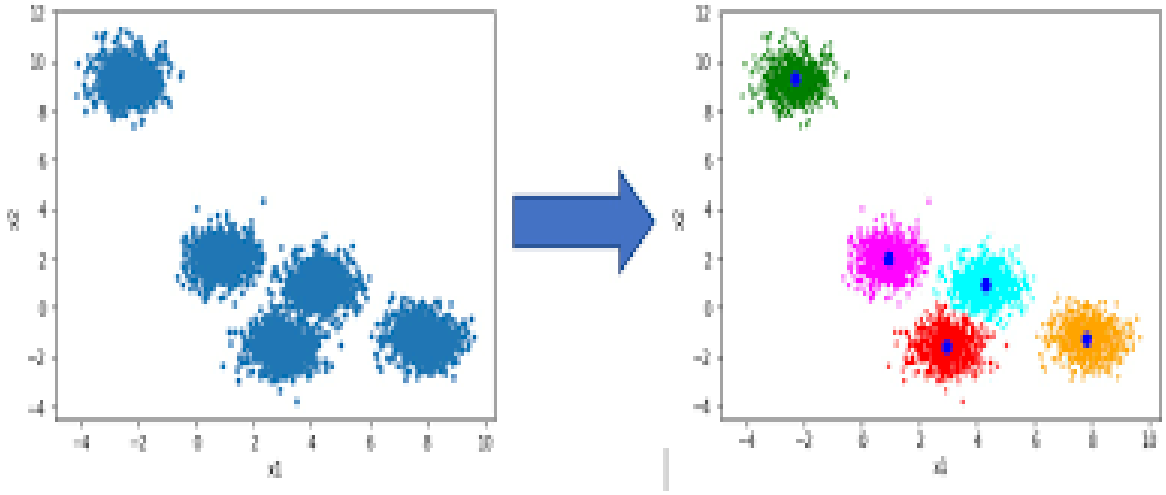

Figure 1: Clustering Algorithm

The clustering technique can be divided into following types:

1. Partitioning Clustering (Centroid-based Clustering)
2. Density-based Clustering
3. Hierarchical Clustering
4. Distributed Model-based Clustering
5. Fuzzy Clustering

K-means Clustering:

K-Means Clustering is an Unsupervised Learning Algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on. It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

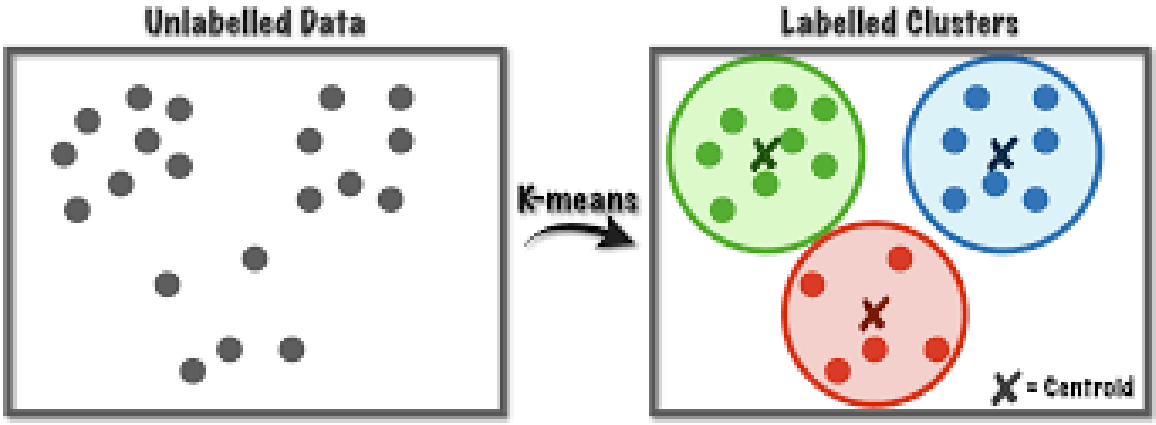

Figure 2: K-means Clustering Algorithm

Algorithm:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each data point to the new closest 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 implemented K-means clustering algorithms on sales data and determined number of clusters using Elbow method.

Questions:

1. What is Clustering?

2. Explain any one distance metric?

3. What is Elbow method?

4. Explain significance of k in k-means clustering?

5. List out types of clustering algorithms?

LP-III Machine Learning (2022-23) **Assignment 5:** Predict the price of the Uber ride from a given pickup point to the agreed drop-off location.	
Student Name:	**Roll No. :**
Batch:	**Division :**

Assignment 4:
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 and random forest regression models.
5. Evaluate the models and compare their respective scores like R2, RMSE, etc.

Title: Implement Linear Regression and Random Forest Regression algorithm to predict the price of the Uber ride from a given pickup point to the agreed drop-off location using dataset available at https://www.kaggle.com/datasets/yasserh/uber-fares-dataset.

Aim: Predict and Analyse Results of Linear Regression and Random Forest for Regression.

Prerequisites: Linear Regression, Random Forest, Decision Tree.

Theory:

Linear Regression Algorithm:

Regression Analysis is the process of estimating the relationship between a dependent variable and independent variables.

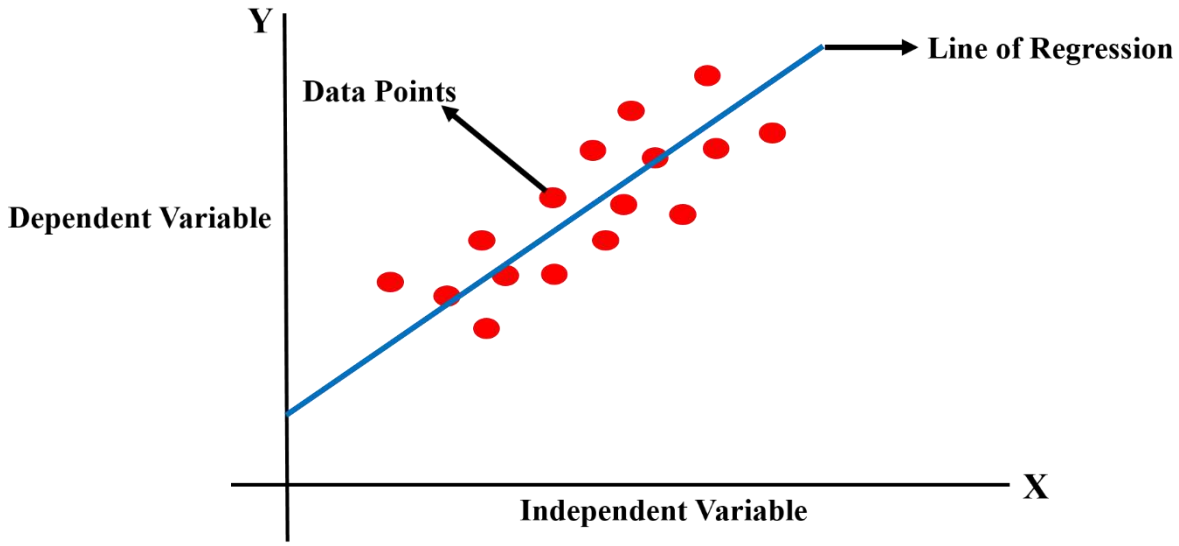

Figure 1: Linear Regression

Simple Linear Regression:

The number of **independent variables** is **one** and there is a **linear relationship** between the **independent(x)** and **dependent(y)** variable.

$$y = \propto_0 + \propto_1 (x) + \varepsilon$$

y = dependent variable

x = independent variable

$\propto_0$ and $\propto_1$ = Regression Coefficients

$\varepsilon =$ Residual Error

Multiple Linear Regression:

The number of independent variables is more then one and there is a linear relationship between the independent(x) and dependent(y) variable.

$$y = \propto_0 + \propto_1 x_1 + \propto_2 x_2 + \propto_3 x_3 + \ldots + \propto_n x_n + \varepsilon$$

y = dependent variable

$x_1, x_2, \ldots, x_n$= independent variable

$\propto_0, \propto_1, \propto_2, \ldots, \propto_n$ = Regression Coefficients

$\varepsilon =$ Residual Error

Random Forest:

Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model. Random Forest works in 2 phases:

1. First is to create the random forest by combining N number of decision trees
2. Second is to make predictions for each tree created in the first phase.

Algorithm:

Step-1: Select random K data points from the training set using Row Selection with Replacement and Feature Selection with Replacement.

Step-2: Build the decision trees associated with the selected data points (Bootstrap Samples).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Conclusion: Using concept of Linear Regression and Random Forest Regression algorithms, we have predicted price of the Uber ride from a given pickup point to the agreed drop-off location, compared and evaluated Linear Regression and Random Forest algorithm using evaluation metrics.

Questions:

1. What are Outliers?

2. Explain Intercept and Regression Coefficients?

3. What is Ensemble Technique?

4. Explain Bagging Technique?

5. Explain: 1. head() 2. shape 3. isnull() 4. drop()?