* EXPERIMENT No: 03

TITLE:- Implement K-nearest neighbour classification using python.

THEORY:-

K-Nearest Neighbours (KNN) is a supervised machine learning techniques that may be used to handle both classification and regression tasks.

KNN is regarded as an algorithm that originates from actual life. People tend to be impacted by the people around them. The KNN classifier in Python is one of the simplest and widely used classification Algorithm, where a new data point is classified based on its similarity to a specific group of neighbours data points.

KNN is a supervised machine learning algorithm. In supervised learning, the algorithm learns from labeled training data, where each data point is associated with a known label or outcome. KNN specificaly requires labeled training data to classify new data points based on their similarity to existing data points with known class labels. It falls under the category of supervised learning algorithm.

* FEATURES

- KNN is a simple yet powerful algorithm/technique in supervised machine learning, applicable not only to classification, but also regression tasks.
- Its distinguishing features likes in its instance-based learning approach, where it memorizes the entire training dataset rather than constructing a model. when tasked with predicting the class or value of a new data point, KNN identifies its k-nearest neighbour based on a chosen distance metric, such as Euclidean or manhatten distance. The predicted class in classification tasks or the average value in regression tasks is then determined by majority voting among these neighbours.

KNN's performance hinges on several key factors. Parameters Selection, particularly the choice of k, significantly impacts its effectiveness. A smaller k value yields more flexible decision boundaries but risks overfitting, while a larger k value may lead to oversmoothing. Aditionally, feature scalling and the selection of an appropriate distance metric are crucial for accurate predictions on test data. While KNN excels in its ability to handle multiclass class^{n} task and regression problem, its. Nonetheless, with proper parameter tunning and preprocessing steps, KNN remains a versatile & effective algorithm.

Working of KNN Classifier in Python.

For a given data point in the set, the algorithms find the distance between this and all other k number of data points in the dataset close to all the initial points & vote for that category that has the most frequency. Usually, Euclidiean distance is taking as a measure of distance. Thus the end resultant model is just the labeled data placed in a space. This algorithm is popularly known for various applications like genetics, forecasting, etc.

The algorithm is best when more feature are present and out shows svm in this case. KNN reduces overfitting use the square root of the number of samples in the dataset as value for k. An optimal value has to be found out since lower value may lead to overfitting and higher value may require high computational complication in distance. So using an error plot may help.

Steps of KNN for classifying a new data point

Step 1
Select the value of k neighbours (say k=5)

Step 2 Find the K(5) nearest data point for our new data point based on euclidean.

Step 3 :

Among these K data points count the data points in each category decision making.

Step 4 :

Assign the new data point to the category that has the most neighbours at the new datapoint.

CONCLUSION :-

Implementing K-nearest neighbour classification using Python offers a versatile and intutive approach to pattern recognition and classification tasks. By harnessing Python's rich ecosystem of libraries and resources, such as scikit learn, developers can seamlessly integrate KNN algorithm into their experiment, thereby empowering them to make informed decision based on nearest neighbour's collective wisdom.

* EXPERIMENT. NO: 04

TITLE:- K-Means Clusterring

THEORY:-

Given the following data, which specify classification for nine combinations of VAR1 & VAR2 predict a classification for a case where VAR1=0.906 and VAR2=0.606, using the result of K-means clusterring with three means (i.e, 3 centroids)

VAR 1	VAR 2	CLASS
1.713	1.586	0
0.180	1.786	1
0.353	1.240	1
0.940	1.566	0
1.486	0.759	1
1.266	1.106	0
1.540	0.419	1
0.459	1.799	1
0.773	0.186	1

We need apply K-means clusterring with 3 means (i.e 3 centroids) predict the class for the var1=0.906 and VAR2=0.606

CONCLUSION:-

K-means clustering is a vital unsupervised learning technique, efficiently partitioning data into distinct clusters by iteratively optimizing centroids. Widely applied accross industries, from market segmentation to image analysis, K-means facilitates data exploration and pattern recognition, driving informed decision-making and innovation.

ASSIGNMENT NO.05

TITLE :- Implement linear Regression using Python

THEORY:-

Linear regression is propbably on of the most Important and widely used regression techniques. It's among the simplest regression methods. one of its main advantage is that the ease of interrpreting results.

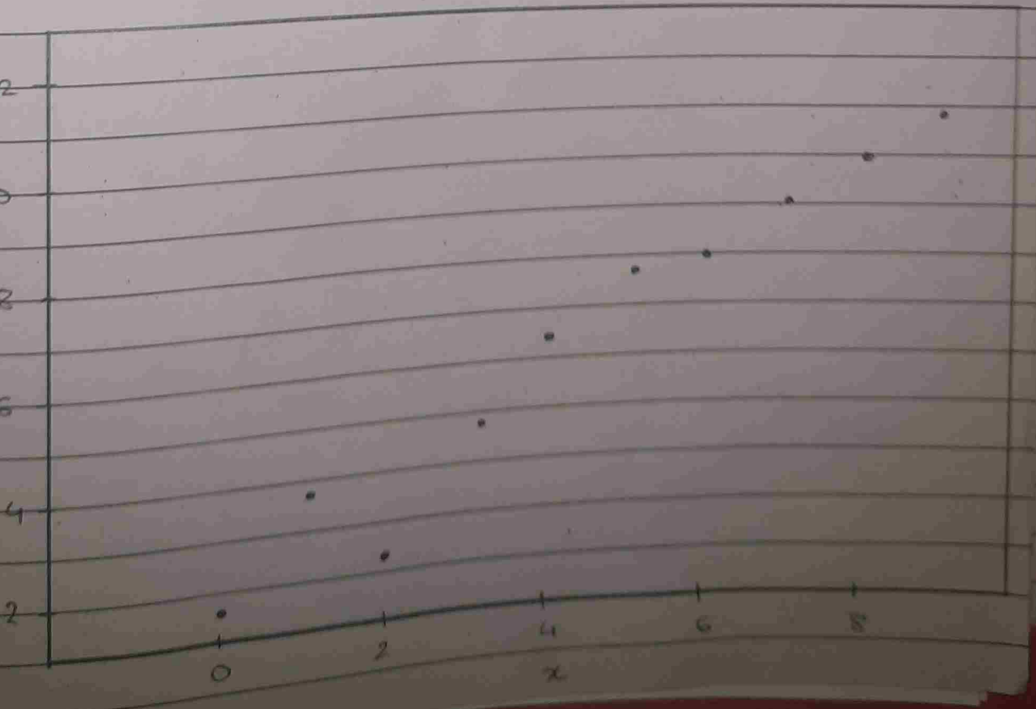
Linear Regression is statistical method that is used to predict a continous dependent variable (target) based on one or more independent variable (predictor) This techniques assomes a relationship between the dependent & independent variables, which implies that the dependent variable changes proportionally with changes in independent voriable

In other words linear regression is used to determine the extent to which one or more variables can predict the value of the dependent variable.

In linear Regression, we assume that the two variables i.e dependent & independent are linearly related. Hence, we try to find a linear function that predicts the response value (y) as accurately as possible as a function of the feature, or independent variable (x). NbN

Let us consider a dataset where we have a value of response y for every feature x:
For generality we define:

x as feature vector, i.e x = [x_1, x_2, ... x_n]
y as response vector, i.e y = [y_1, y_2, ... y_n]
for n observations (in the above example, n=10)
A scatter plot of the above dataset looks like



* EXPERIMENT. NO: 06

# TITLE: Implement Naive Bayes theorem to classify the Eight text.

# THEORY:-

* Naive Bayes Approach is a popular method for classifying text docoments. This method classifies document into predetermined types based on the likelihood of a word occurring, utilizing the concept of the Bayes Theorem.

* Text Classification using Naive Bias:

A probabilistic classification technique, the Naive Bayes algorithm is predicted on Robust, if Naive, independence assumption in its probability model The Naive Bayes algorithm uses Thomas Bayes theorem, which forms the basis for probability model creation. The model can be trained using these probability models in supervised learning.

The Naive Bayes Algorithm is a probabilistic classification method that bases its prediction on the Bayes theorem.

Based on observable data, the Bayes Theorem determines a hypothesis probability when using Naive Bayes, an instances features serves as the evidence, while the class to which the instance belongs serves as the hypothesis.

There are several instances in which Naive Bayes can be applied with great effectiveness.

* **Text Classification:-**

Naive Bayes in text based such as spam filtering sentiment analysis, and document categorization due to its simplicity and efficiency with high-dimensional data.

* **Limited Training Data:**

Naive Bayes can perform well with limited training data, making it valuable when dealing with small datasets or situations wheare collecting extensive labeled data is challenging.

Simple & Quick Prototyping:

When a quick and simple solution is needed for prototyping or baseline performance, Naive Bayes is suitable choice due to its ease of implementation.

CONCLUSION :-

Naive Bayes theorem is a valuable tool for classifying English text, offering efficiency and accuracy through probabilistic principles. Its simplicity and effectiveness make it widely adopted for tasks like spam detection and sentiment analysis, advancing natural language processing applications significantly.

EXPERIMENT NO: 07

TITLE:- Unsupervised Learning: Implement K-Means Clustering & Hierarchial clustering on proper data set of your choice. Compare their convergence.

THEORY:-

K-Means is method of Cluster analysis using a pre-specified no. of clusters. It requires advanced knowledge of "K".
Hierarchial Clustering also known as hierarchial cluster analysis (HCA) is also a method of cluster analysis which seeks to build a hierarchy of cluster without having fixed number of Cluster.

Unlike Hierarchial Clustering, K-means clustering seeks to partition the original data points into 'K' groups or clusters where the user specifies the 'K' in advance. The general idea is to look for clusters that minimizes the squared Euclidean distance of all the points from centre over all attributes (variables or features) and merge those individuals in an iterative manner.

-means Clustering in Python with Scikit learn tutorial will help you understand the inner workings of K-means clustering with an interesting case study.

enifits: It is computationally efficient compared to hierarchial clustering and can be used to analyze large data set.

-means is easier to understand & Implement.

rawbacks :- It is less flexible than hierarchial clustering because it constraints the user to specify the number of cluster beforehead, which may not be obvious in some situations.

-result is not stable & changes from one iteration another for the same dataset. It is more sensitive outliers because the use of outliers in the data mpacts the mean of the cluster.

th K-means and hierarchial clustering are incapable handling categorical data directly and may not rks well with data that is not continous or has y large variance.

output image is clearly showing the five different clusters with different colors. The clusters are formed between two parameters of the dataset, Anual income of customer & Spending. we can change the colors & labels as per the requirement or choice. we can also observe some points from the above patterns, which are given below:

- Cluster1 shows the customers with average salary & average spending so we can categorize these customer as.

Cluster 2 shows the customers has a high income but low spending, so we can categorize them as careful.

Cluster 3 shows the low income & also low spending so they can be categorized as sensible.

cluster 4 shows the customer with low income with very high spending so they can be categorized as careless.

Cluster 5 shows the customer with high income & high spending so they can be categorized as target, and these customers can be the most profitable customers for the mall owner.

CONCLUSION: K-means typically converges faster than hierarchial clustering on the mall Customer Segmentation dataset. The decision between them rests on balancing speed & interpretability based on analysis requirements.

Conclusion :- In this experiment, we studied the concept of Linear Regression ,also the types of LR and its applications. we implemented a python progoram using Linear Regression.

4/1/24