COLLEGE OF ENGINEERING AND MANAGEMENT,KOLAGHAT

SANTU JANA

SEM:6TH

YEAR:3RD

SUBJECT: PATTERN RECOGNITION

ROLL:CSE/21/L-146

UNIVERSITY ROLL:10700121127

TOPIC: KNN (K-NEAREST NEIGHBOURHOOD) CLASSIFIER WITH EXAMPLE

Introduction

- K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems.
- However, it is mainly used for classification predictive problems in industry.
- There are three categories of learning algorithms:
- 1. Lazy learning algorithm - KNN is a lazy learning algorithm because it does not have a specialized training phase or model and uses all the data for training while classification.
- 2. Non-parametric learning algorithm - KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.
- 3. Eager learning algorithm - Eager learners, when given a set of training tuples, will construct a generalization model before receiving new (e.g., test) tuples to classify.

KNN Algorithm:

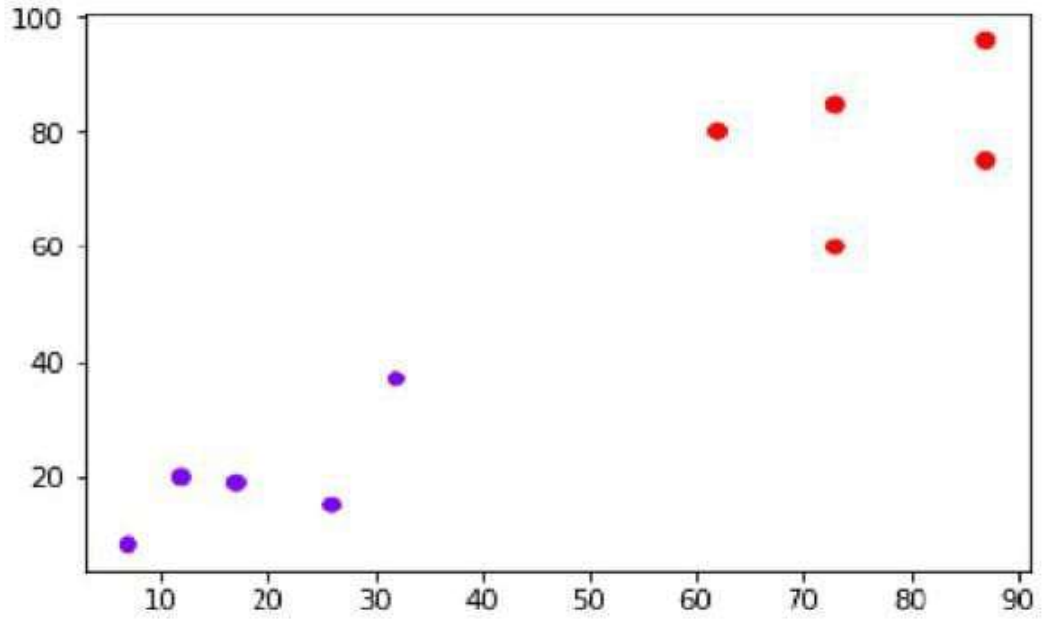
- K-nearest neighbors (KNN) algorithm uses 'feature similarity' to predict the values of new data points which further means that the new data point will be assigned a value based on how closely it matches the points in the training set.
- We can understand its working with the help of following steps –
- Step 1 - For implementing any algorithm, we need dataset. So during the first step of KNN, we must load the training as well as test data.
- Step 2- Next, we need to choose the value of - Ki.e. the nearest data points. K can be any integer.

KNN Algorithm(cont..):

- Step 3 - For each point in the test data do the following –
- 3.1 - Calculate the distance between test data and each row of training data with the help of any of the method namely: Euclidean, Manhattan or Hamming distance. The most commonly used method to calculate distance is Euclidean.
- 3.2 Now, based on the distance value, -- sort them in ascending order.
- 3.3 Next, it will choose the top K rows from the sorted array.
- 3.4 - Now, it will assign a class to the test point based on most frequent class of these rowStep 4 - End

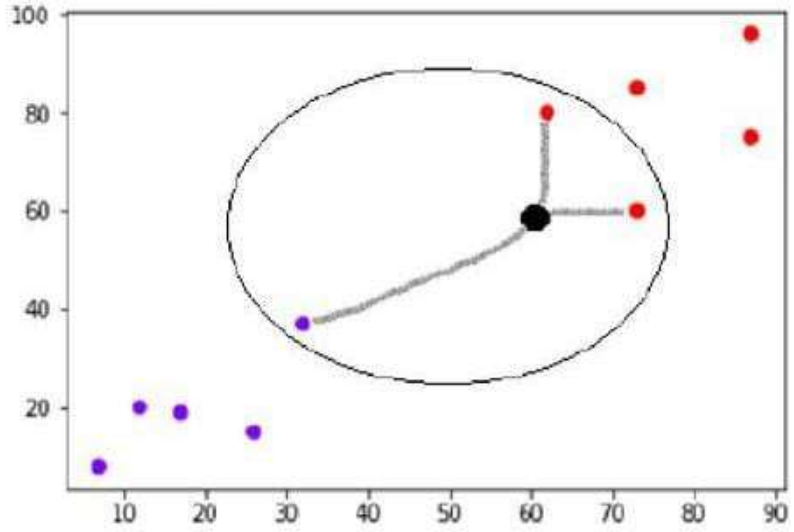
Example-1:

- The following is an example to understand the concept of K and working of KNN algorithmSuppose we have a dataset which can be plotted as follows:



Example-1 (Conti..)

- • Now, we need to classify new data point with black dot (at point 60,60) into blue or red class. We are assuming K = 3 i.e. it would find three nearest data points. It is shown in the following diagram:



We can see in the beside diagram the three nearest black dot. Among those three, two of them lies in Red class assigned in red class.

Advantages

1. No Training Period: KNN is called Lazy Learner (Instance based learning). It does not learn anything in the training period. It does not derive any discriminative function from the training data. It stores the training dataset and learns from it only at the time of making real time predictions. This makes the KNN algorithm much faster than other algorithms that require training e.g. Linear Regression etc.
2. Since the KNN algorithm requires no training before making predictions, new data can be added seamlessly which will not impact the accuracy of the algorithm.
3. KNN is very easy to implement. There are only two parameters required to implement KNN i.e. the value of K and the distance function (e.g. Euclidean or Manhattan etc.)

Dis advantages

- 1. Does not work well with large dataset: In large datasets, the cost of calculating the distance between the new point and each existing points is huge which degrades the performance of the algorithm.
- 2. Does not work well with high dimensions: The KNN algorithm doesn't work well with high dimensional data because with large number of dimensions, it becomes difficult for the algorithm to calculate the distance in each dimension.
- 3. Need feature scaling: We need to do feature scaling (standardization and normalization) before applying KNN algorithm to any dataset. If we don't do so, KNN may generate wrong predictions.
- 4. Sensitive to noisy data, missing values and outliers: KNN is sensitive to noise in the dataset. We need to manually impute missing values and remove outliers.

Conclusion:

- KNN is an effective machine learning algorithm that can be used in credit scoring, prediction of cancer cells, image recognition, and many other applications. The main importance of using KNN is that it's easy to implement and works well with small datasets.