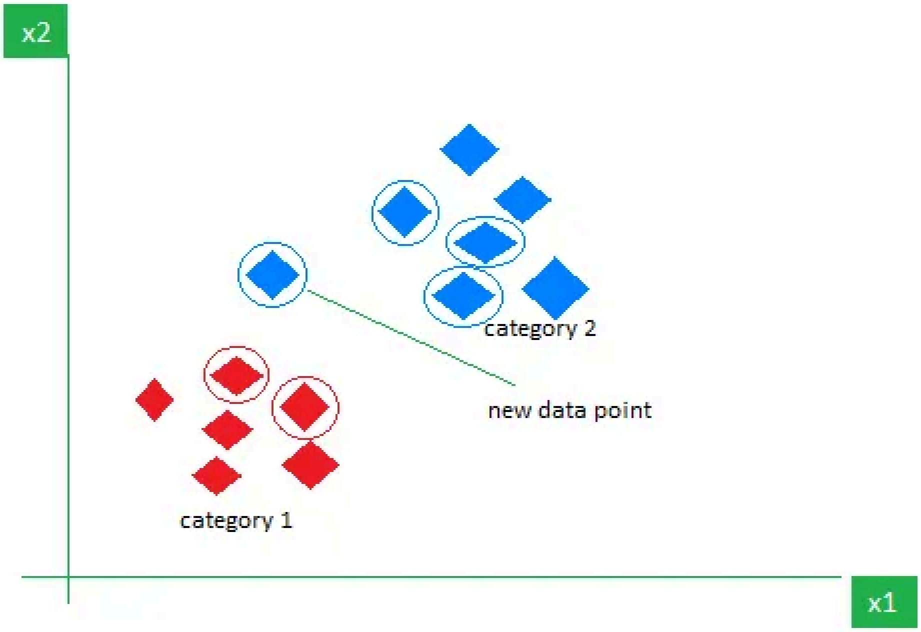
KNN (K-Nearest Neighbor)

Mallinathkhonde
4 min read · Just now

K-Nearest Neighbors (KNN) is a simple way to classify things by looking at what's nearby. Imagine a streaming service wants to predict if a new user is likely to cancel their subscription (churn) based on their age. They checks the ages of its existing users and whether they churned or stayed. If most of the "K" closest users in age of new user canceled their subscription KNN will predict the new user might churn too. The key idea is that users with similar ages tend to have similar behaviors and KNN uses this closeness to make decisions.

K-Nearest Neighbors is also called as 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.

As an example, consider the following table of data points containing two features:



What is ‘K’ in K Nearest Neighbors?

In the **k-Nearest Neighbors (k-NN)** algorithm **k** is just a number that tells the algorithm how many nearby points (neighbors) to look at when it makes a decision.

How to choose the value of k for KNN Algorithm?

The value of k is critical in KNN as it determines the number of neighbors to consider when making predictions. Selecting the optimal value of k depends on the characteristics of the input data. If the dataset has significant outliers or noise a higher k can help smooth out the predictions and reduce the influence of noisy data. However choosing very high value can lead to underfitting where the model becomes too simplistic.

Statistical Methods for Selecting k:

- **Cross-Validation:** A robust method for selecting the best k is to perform k-fold cross-validation. This involves splitting the data into k subsets training the model on some subsets and testing it on the remaining ones and repeating this for each subset. The value of k that results in the highest average validation accuracy is usually the best choice.
- **Elbow Method:** In the **elbow method** we plot the model's error rate or accuracy for different values of k. As we increase k the error usually decreases initially. However after a certain point the error rate starts to decrease more slowly. This point where the curve forms an "elbow" that point is considered as best k.
- **Odd Values for k:** It's also recommended to choose an odd value for k especially in classification tasks to avoid ties when deciding the majority class.

Distance Metrics Used in KNN Algorithm

KNN uses distance metrics to identify nearest neighbor, these neighbors are used for classification and regression task. To identify nearest neighbor we use below distance metrics:

1. Euclidean Distance

Euclidean distance is defined as the straight-line distance between two points in a plane or space. You can think of it like the shortest path you would walk if you were to go directly from one point to another.

Euclidean Distance= (∑*j*=1*d*(*xj*–*Xij*)2])1/2

2. Manhattan Distance

This is the total distance you would travel if you could only move along horizontal and vertical lines (like a grid or city streets). It’s also called “taxicab distance” because a taxi can only drive along the grid-like streets of a city.

Manhattan Distance=j=1∑d∣xj−Xij∣

3. Minkowski Distance

Minkowski distance is like a family of distances, which includes both **Euclidean** and **Manhattan distances** as special cases.

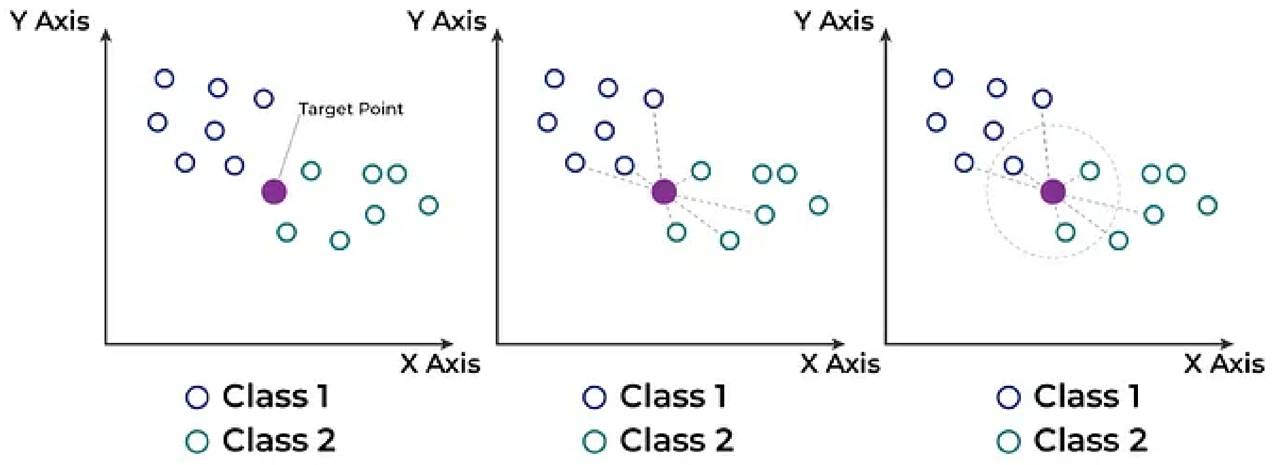
Minkowski Distance=(j=1∑d∣xj−Xij∣p)1/p

From the formula above we can say that when p = 2 then it is the same as the formula for the Euclidean distance and when p = 1 then we obtain the formula for the Manhattan distance.

So, you can think of Makowski as a flexible distance formula that can look like either Manhattan or Euclidean distance depending on the value of p

Working of KNN algorithm

The K-Nearest Neighbors (KNN) algorithm operates on the principle of similarity where it predicts the label or value of a new data point by considering the labels or values of its K nearest neighbors in the training dataset.



Step 1: Selecting the optimal value of K

- K represents the number of nearest neighbors that needs to be considered while making prediction.

Step 2: Calculating distance

- To measure the similarity between target and training data points Euclidean distance is used. Distance is calculated between data points in the dataset and target point.

Step 3: Finding Nearest Neighbors

- The k data points with the smallest distances to the target point are nearest neighbors.

Step 4: Voting for Classification or Taking Average for Regression

- When you want to classify a data point into a category (like spam or not spam), the K-NN algorithm looks at the **K closest points** in the dataset. These closest points are called neighbors. The algorithm then looks at which category the neighbors belong to and picks the one that appears the most. This is called **majority voting.**

- In regression, the algorithm still looks for the **K closest points.** But instead of voting for a class in classification, it takes the **average** of the values of those K neighbors. This average is the predicted value for the new point for the algorithm.

Knn Algorithm

Written by Mallinathkhonde

0 Followers · 1 Following

Edit profile