

INTRO TO K-NEAREST NEIGHBORS (KNN)

Introduction

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- Popular machine learning algorithm used for classification and regression tasks.
- 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.
- 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.

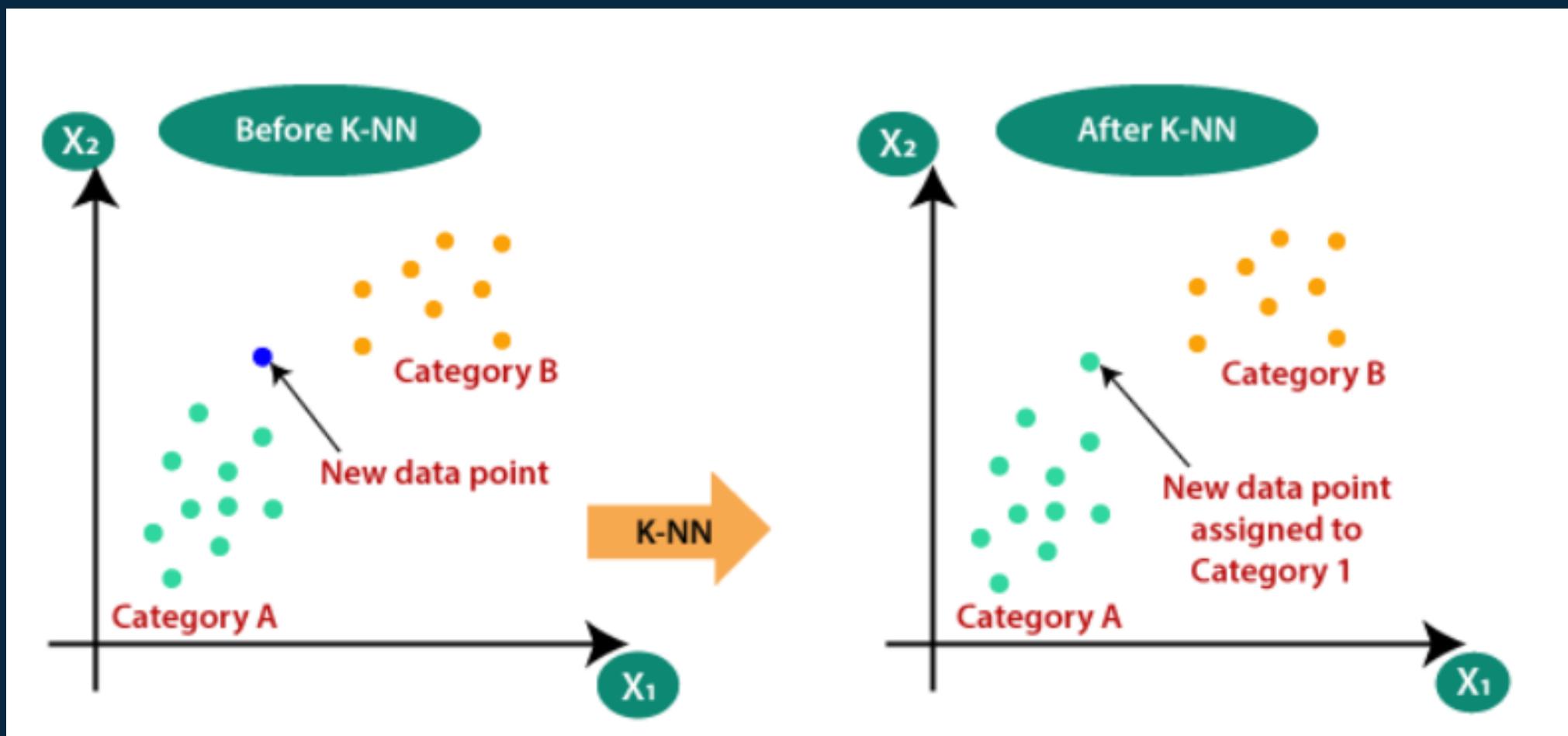
KNN Classifier



Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog.

So for this identification, we can use the KNN algorithm, as it works on a similarity measure

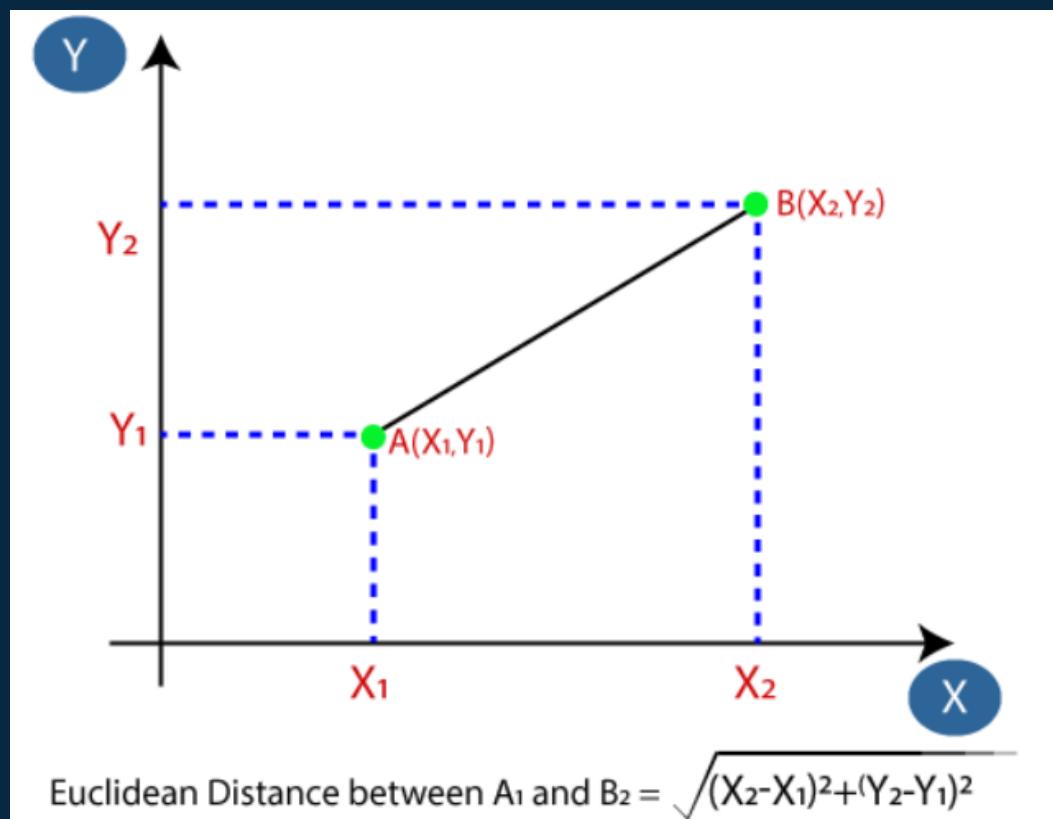
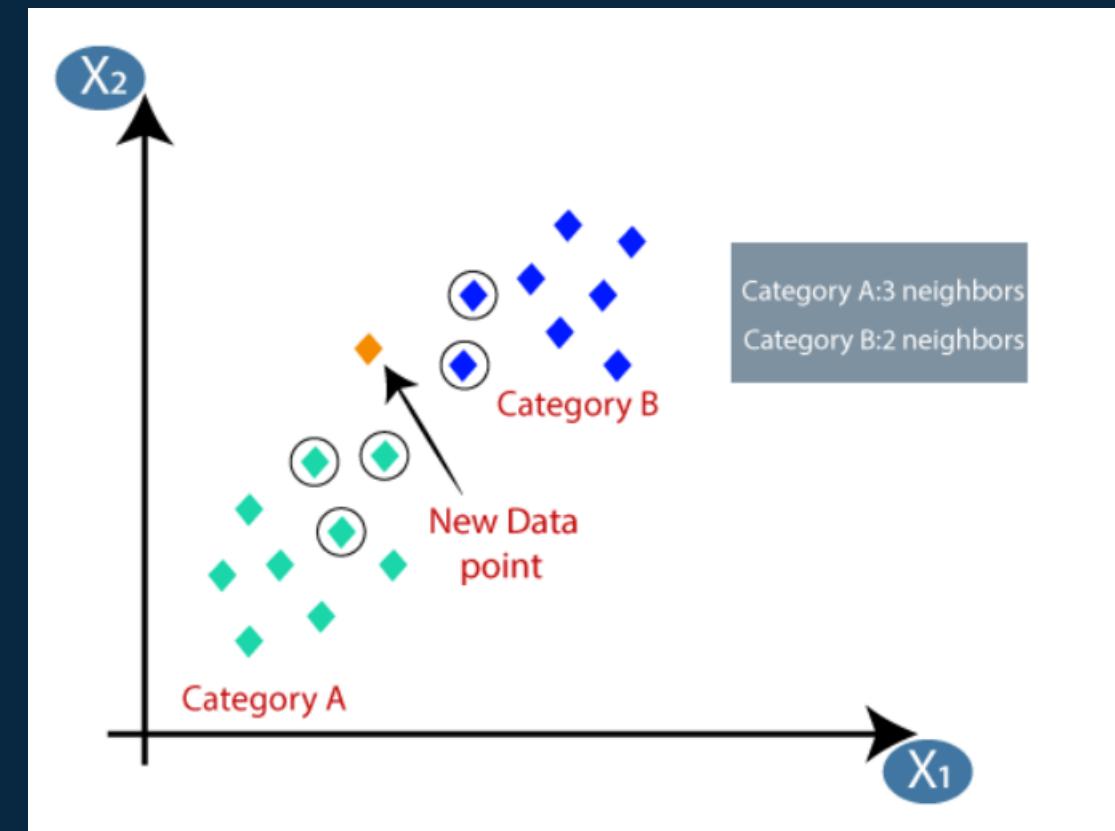
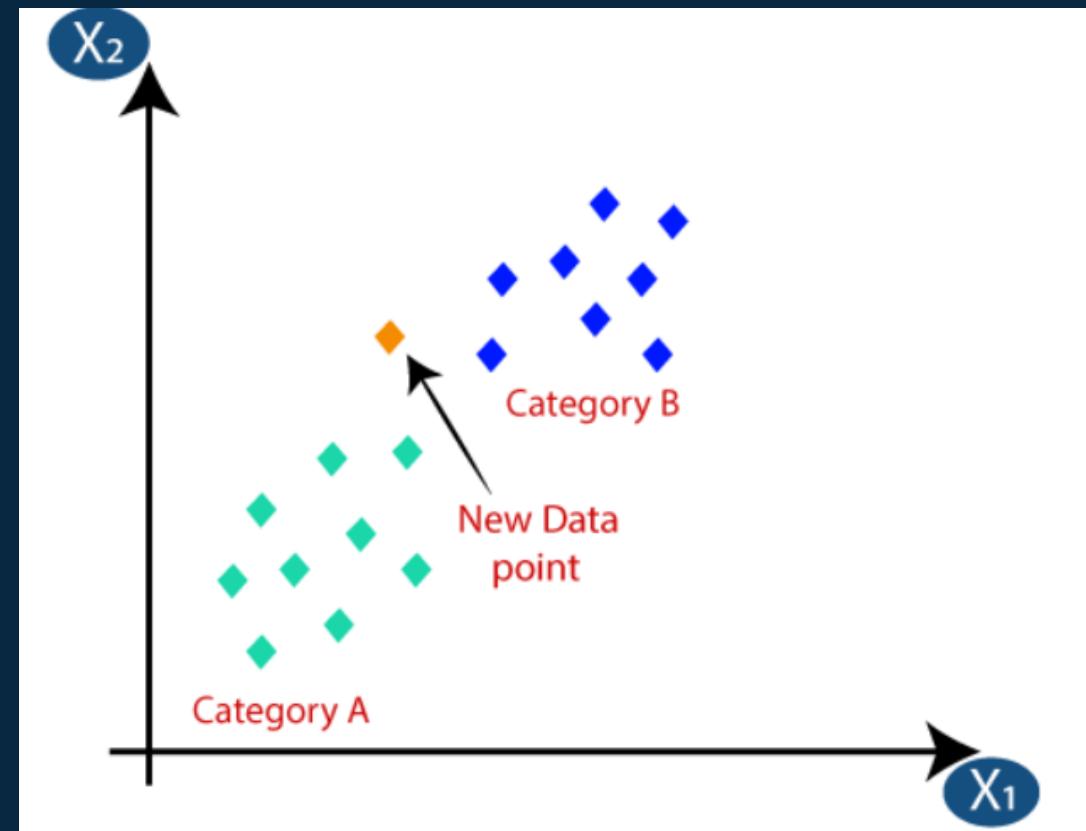
Why do we need KNN



With the help of K-NN, we can easily identify the category or class of a particular dataset.

How does K-NN work?

- **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.



- choose the number of neighbors, so we will choose the $k=5$.
- calculate the Euclidean distance between the data points.
- By calculating the Euclidean distance we got the nearest neighbors

Advantages

- It is simple to implement.
- It is robust to the noisy training data
- It can be more effective if the training data is large.

Disadvantages

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples