K-Nearest Neighbor Algorithm

Exemplar-based Methods

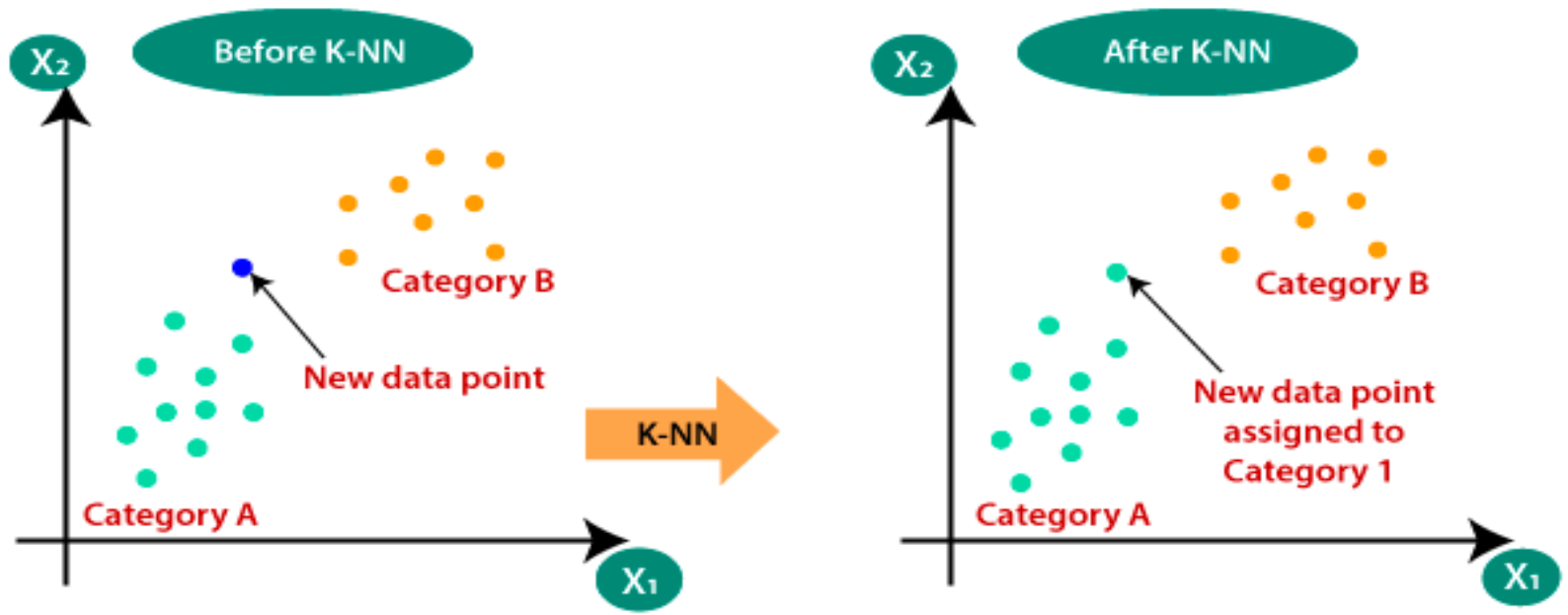
- Parametric models: parameters are estimated from a variable sized datasets. Once the model is fit, the data is thrown away
- KNN is a kind of non parametric models that keep the training data around
- It focuses on similarity between text input x and each of the training inputs x_n
- Since the models keep the training examples around at test time, we call them exemplar-based models.

- Supervised Model: KNN Algorithm
- Unsupervised Model: K Means Algorithm

Steps in KNN Algorithm:

- **Store Training Data**: KNN memorizes the training dataset.
- **Identify the neighbors:** Select the number K of the neighbors
- **Compute the distance:** Calculate the Euclidean distance of **K number of neighbors**
- Take the K nearest neighbors as per the calculated Euclidean distance.
- **Vote for Classification :** Among these k neighbors, count the number of the data points in each category.
- Assign the new data points to that category for which the number of the neighbor is maximum.
- **Compute Average for Regression**:
  - Regression: The output is calculated as the mean/weighted mean of the target values of the k nearest neighbors.

K Nearest Neighbor classifier

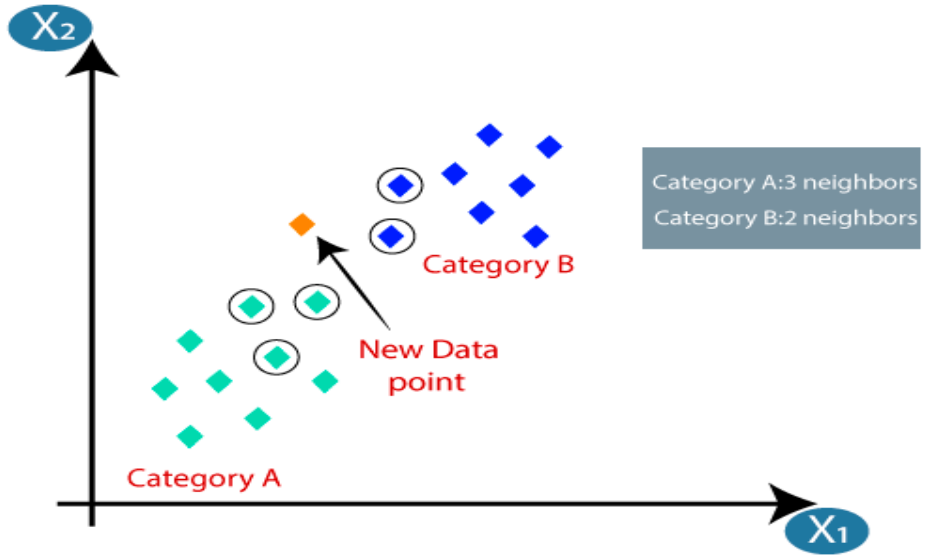


- In the **k-Nearest Neighbours (k-NN)** algorithm **k** is just a number that tells the algorithm how many nearby points (neighbours) to look at when it makes a decision.

Example:

- Imagine you're deciding which fruit it is based on its shape and size. You compare it to fruits you already know.
- If **k = 5**, the algorithm looks at the 5 closest fruits to the new one.
- If 3 of those 5 fruits are apples and 2 are banana, the algorithm says the new fruit is an apple because most of its neighbours are apples.

The value of k in the k-nearest neighbors (k-NN) algorithm should be chosen based on the input data.
If the input data has more outliers or noise, a higher value of k would be better.



Distance Metrics Used in KNN Algorithm

Euclidean Distance

- Euclidean distance is defined as the straight-line distance between two points in a plane or space.

$$\text{distance}(x, X_i) = \sqrt{\sum_{j=1}^{d} (x_j - X_{i_j})^2]}$$

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 street

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

Advantages of KNN Algorithm:

- It is simple to implement.
- It is robust to noisy training data

Disadvantages of KNN Algorithm:

- 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.
- It can be more effective if the training data is large.

The curse of dimensionality:

- The **Curse of Dimensionality** refers to the challenges and phenomena that arise when analyzing and organizing data in high-dimensional spaces.
- As the number of dimensions (features) in the data increases, the volume of the space grows exponentially, leading to various computational and analytical difficulties.

END