KNN

—

By Jayani Nathvani

What is KNN?

K Nearest Neighbor algorithm falls under the Supervised Learning category and is used for classification (most commonly) and regression. It is a versatile algorithm also used for imputing missing values and resampling datasets.As the name (K Nearest Neighbor) suggests it considers K Nearest Neighbors (Data points) to predict the class or continuous value for the new Datapoint.

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on the Supervised Learning technique.

How does KNN work?

- Initialize K to your chosen number of neighbors — Select the number K of the neighbors
- For each example in the data- Calculate the **Euclidean distance** of **K number of neighbors**
- Calculate the distance between the query example and the current example from the data.
- Add the distance and the index of the example to an ordered collection
- Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances
- Pick the first K entries from the sorted collection
- Get the labels of the selected K entries
- If regression, return the mean of the K labels
- If classification, return the mode of the K labels — Assign the new data points to that category for which the number of the neighbor is maximum.

How to select the value of K in the K-NN Algorithm?

- There is no particular way to determine the best value for “K”, so we need to try some values to find the best out of them. The most preferred value for K is 5.
- A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.
- The impact of selecting a smaller or larger K value on the model
- **Larger K value:** The case of underfitting occurs when the value of k is increased. In this case, the model would be unable to correctly learn on the training data.
- **Smaller k value:** The condition of overfitting occurs when the value of k is smaller. The model will capture all of the training data, including noise. The model will perform poorly for the test data in this scenario.

Advantages of KNN Algorithms

- 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 of KNN Algorithm:

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

KNN Hyperparameters

K (number of neighbors)
Distance metric
Weight function (uniform or distance-based)

Real World Use Cases

- Recommendation systems
- Image recognition
- Handwriting detection
- Medical diagnosis
- Fraud detection