

- K-nearest neighbour algorithm
- K-Nearest Neighbors (KNN) is a simple and intuitive machine learning algorithm used for classification and regression tasks. It is a type of instance-based learning, also known as lazy learning, where the algorithm memorizes the training dataset and makes predictions based on the similarity of new instances to existing examples.

Here is step to explain KNN algorithm

1. Data Collection

- Gather a dataset that contains features and corresponding labels. The features are the attributes that describe each data point, and the labels are the categories or values you want to predict.

2. Choose the number of K

- Decide on the number of neighbors (K) to consider when making predictions. K is a hyperparameter that you need to choose before applying the algorithm. A small value of K (e.g., 3 or 5) might make the algorithm sensitive to noise, while a large value of K may smooth out patterns in the data.

3. Calculate distance

- For a given data point that you want to classify or predict, calculate the distance between that point and all other points in the training dataset. The most common distance metrics are Euclidean distance, Manhattan distance, and Minkowski distance.

4. Identify nearest neighbour

→ Identify the K nearest neighbors to the data point based on the calculated distances. These are the K data points with the smallest distances to the input data.

5. Majority Vote (Classification) or Weighted Average (Regression)

→ For regression tasks, calculate the average (or weighted average) of the target values of the K nearest neighbors. This average becomes the predicted value for the input data point.

6. Make predictions

→ Apply the decision rules from the previous step to make predictions for all data points in the testing dataset.

7. Evaluate the model

→ Assess the performance of the model using appropriate evaluation metrics. Common metrics include accuracy for classification and mean squared error for regression.

- Advantages and Disadvantages of KNN algorithm

- Advantages of KNN algorithm

- 1. Simple to Implement

- 2. No Training Phase

- 3. Versatility

- Disadvantages of KNN algorithm

- 1. Computationally Expensive

- 2. Choosing the Right Value of K

- 3. High Memory Requirement

4. Not Suitable for High-Dimensional Data