K-Nearest Neighbors: Classification and Regression.

Initially KNN is a non-parametric classification method developed by Evelyn Fix and Joseph Hodges in 1951, and later expanded by Thomas Cover. It is used for classification and regression. In both the cases input consists of the K closest training examples in data set. The output depends on whether K-NN is used for classification or regression:

- In classification, the output is a class membership(discrete value). An object is classified by the majority vote that is to the class which most of its neighbors belongs.
- In regression, the output is a property value (continuous value). This value is the average of the values of K nearest neighbors.
- The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

If the features are in different scales, then normalizing the training data can improve its accuracy.

KNN Classification Algorithm :

Given a training set X_train with labels y_train, and given a new instance x-test to be **classified**:

- Find the most similar instances (let's call them X_NN(Nearest Neighbors)) to the x_test that are in X-train.
- Get the labels y_NN for the instances in X_NN.
- Predict the labels for x_test by combining the labels y_NN e.g. simple majority vote.

k-NN regression Algorithm :

In k-NN regression, the k-NN algorithm is used for estimating continuous variables. One such algorithm uses a weighted average of the k nearest neighbors, weighted by the inverse of their distance. This algorithm works as follows:

- Compute the Euclidean distance from the query example(x-test) to the labeled examples(X_train).
- Order the labeled examples by increasing distance.
- Find a heuristically optimal number k of nearest neighbors, based on RMSE. This is done using cross validation.
- Calculate an inverse distance weighted average with the k-nearest multivariate neighbors.

Distance Metric used:

- For continuous variable → Euclidean distance.
- For discrete variables → overlap or hamming distance, Large Margin Nearest Neighbor.

Cons:

- A peculiarity of the k-NN algorithm is that it is sensitive to the local structure of the data.
- When the class distribution is skewed, that is examples of more frequent class tend to dominate the prediction.

  Solution for skewed distribution is to weight the classification, considering the distance(d) from test point to each of the k nearest neighbors. The class(value in regression) for each of the KNN is multiplied by the weight proportional to the inverse of the distance(1/d) from that point to test point (x_test), another way is by abstraction in data representation(self-organizing map).

Note :

In binary (two class) classification problems, it is helpful to choose k to be an odd number as this avoids tied votes. One popular way of choosing the empirically optimal k in this setting is via bootstrap method

Resource link : https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm