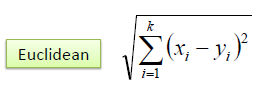
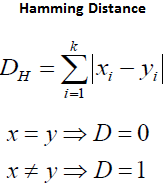
**K Nearest Neighbors**

K Nearest Neighbors (KNN) is a simple algorithm that stores all available samples and classifies new samples or predict numerical targets based on a similarity measurement (e.g., distance functions). It is a nonparametric algorithm, meaning it does not make any underlying assumptions about the distribution of data. KNN is widely used in applications such as credit ratings , bank loan assignment, handwriting detection, image recognition and even video recognition.

KNN can be summarized as follows:  
**- 1) Store the training samples in an array of data points  
- 2) Specify a positive integer K**  
In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Historically, the optimal K for most datasets has been between 3-10.   
**- 3) Calculate the distance of the new sample to all training samples, select the k entries which are closest to the new sample**  
For continuous variables, Eulidean distance defined as follows is used:

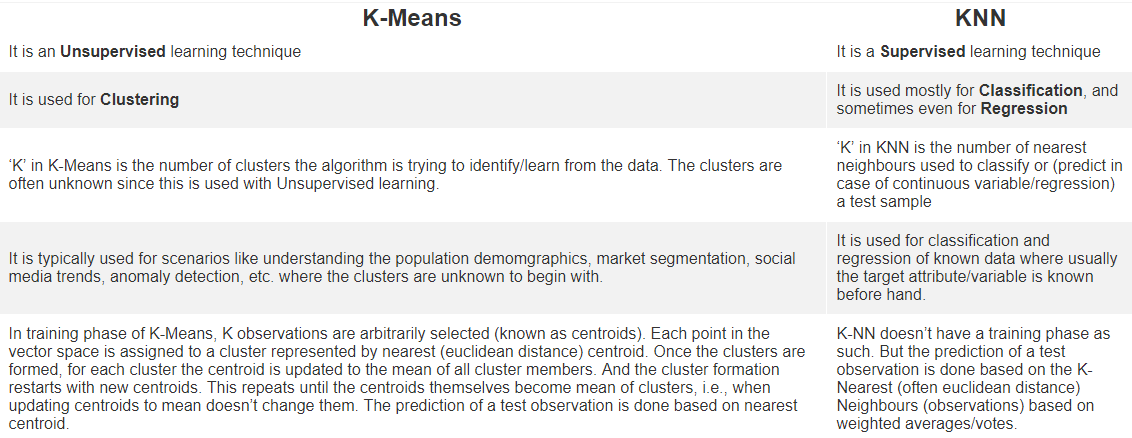


For categorical variables, the Hamming distance must be used:

When variables have different measurement scales or there is a mixture of numerical and categorical variables, the distance can be poorly defined. One solution is to standardize the training set. However, results from the standardized training set may be different from the initial dataset.   
**4) Return the majority labels of the K entries as the label of the new sample**

KNN have a few other features:

1. It can be used for both classification and regression.
2. It stores the entire training dataset which it uses as its representation, therefore it requires high memory and is computationally expensive.
3. It does not learn any model.
4. It is sensitive to irrelevant features and the scale of the data.
5. It is insensitive to outliers.
6. It makes no assumption about data.

KNN is often confused with K-means algorithm. These two algorithms are compared in the following table:

References:

1, https://www.geeksforgeeks.org/k-nearest-neighbours/

2, http://abhijitannaldas.com/kmeans-vs-knn-in-machine-learning.html