Refer repo: [py/ML/17\_knn\_classification at master · codebasics/py · GitHub](https://github.com/codebasics/py/tree/master/ML/17_knn_classification)

What is the KNN algorithm?

The k-nearest neighbors (KNN) algorithm is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. It is one of the popular and simplest classification and regression classifiers used in machine learning today.

While the KNN algorithm can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

**Distance Metrics Used in KNN Algorithm**

As we know that the KNN algorithm helps us identify the nearest points or the groups for a query point. But to determine the closest groups or the nearest points for a query point we need some metric. For this purpose, we use below distance metrics:

**Euclidean Distance**

This is nothing but the cartesian distance between the two points which are in the plane/hyperplane. Euclidean distance can also be visualized as the length of the straight line that joins the two points which are into consideration. This metric helps us calculate the net displacement done between the two states of an object.

**Manhattan Distance**

Manhattan Distance metric is generally used when we are interested in the total distance traveled by the object instead of the displacement. This metric is calculated by summing the absolute difference between the coordinates of the points in n-dimensions.

**Minkowski Distance**

We can say that the Euclidean, as well as the Manhattan distance, are special cases of the Minkowski distance.

From the formula above we can say that when p = 2 then it is the same as the formula for the Euclidean distance and when p = 1 then we obtain the formula for the Manhattan distance.

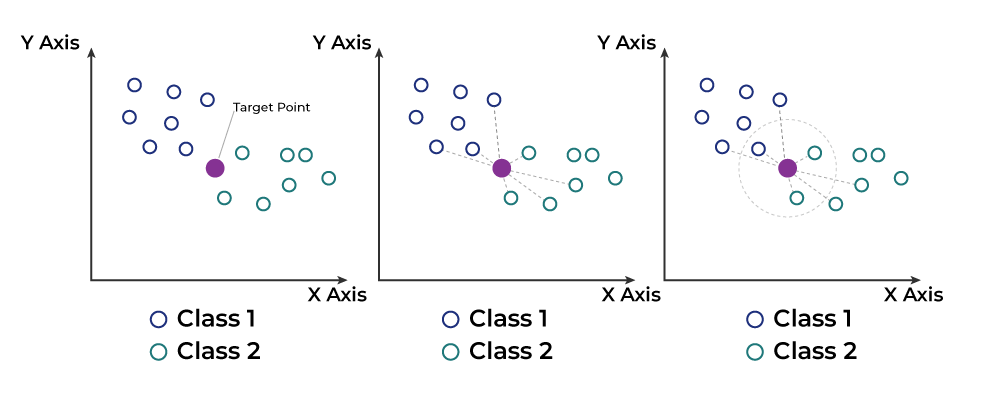
The above-discussed metrics are most common while dealing with a [Machine Learning](https://www.geeksforgeeks.org/machine-learning/) problem but there are other distance metrics as well like [Hamming Distance](https://www.geeksforgeeks.org/hamming-distance-two-strings/) which come in handy while dealing with problems that require overlapping comparisons between two vectors whose contents can be Boolean as well as string values.

**How to choose the value of k for KNN Algorithm?**

The value of k is very crucial in the KNN algorithm to define the number of neighbors in the algorithm. 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. It is recommended to choose an odd value for k to avoid ties in classification. [Cross-validation](https://www.geeksforgeeks.org/cross-validation-machine-learning/) methods can help in selecting the best k value for the given dataset.

**Workings of KNN algorithm**

Thе K-Nearest Neighbors (KNN) algorithm operates on the principle of similarity, where it predicts the label or value of a new data point by considering the labels or values of its K nearest neighbors in the training dataset.



Step-by-Step explanation of how KNN works is discussed below:

**Step 1: Selecting the optimal value of K**

* K represents the number of nearest neighbors that needs to be considered while making prediction.

**Step 2: Calculating distance**

* To measure the similarity between target and training data points, Euclidean distance is used. Distance is calculated between each of the data points in the dataset and target point.

**Step 3: Finding Nearest Neighbors**

* The k data points with the smallest distances to the target point are the nearest neighbors.

**Step 4: Voting for Classification or Taking Average for Regression**

* In the classification problem, the class labels of are determined by performing majority voting. The class with the most occurrences among the neighbors becomes the predicted class for the target data point.
* In the regression problem, the class label is calculated by taking average of the target values of K nearest neighbors. The calculated average value becomes the predicted output for the target data point.