

K Nearest Neighbors Learning



Dr. Singara Singh Kasana

Associate Professor

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology

Patiala, Punjab

Email: singara@thapar.edu

KNN

- K-Nearest Neighbour is a Supervised Learning technique
- This technique assumes the similarities between the new data and the available data and places the new data in the category that is most similar to the available data.
- It can be used for both Regression and Classification but is often used for the Classification problems.

KNN

- K-NN is a **non-parametric technique**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner** because it does not learn from the training set immediately instead it stores the dataset and at the time of prediction, it performs an action on the dataset.
- KNN technique at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

K-Nearest Neighbors (KNN) is one of the simplest algorithms used in Machine Learning for regression and classification problem.

It uses complete data as training data and predict value of output feature(dependent variable) of test data based on similarity measures (e.g. distance between test dataset and training data set)

Classification is done by a majority vote to its neighbors. The test data is assigned to the class of its K nearest neighbors. As you increase the value of K (i.e. number of nearest neighbors), accuracy might increase.

In case of regression, we generally take the mean of the output feature of K nearest neighbors of the test data.

Distances Used in KNN

➤ Euclidean Distance: $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

➤ Manhattan Distance: $|x_1 - x_2| + |y_1 - y_2|$

➤ Chebyshev Distance: $\max(|x_1 - x_2|, |y_1 - y_2|)$

Working of KNN

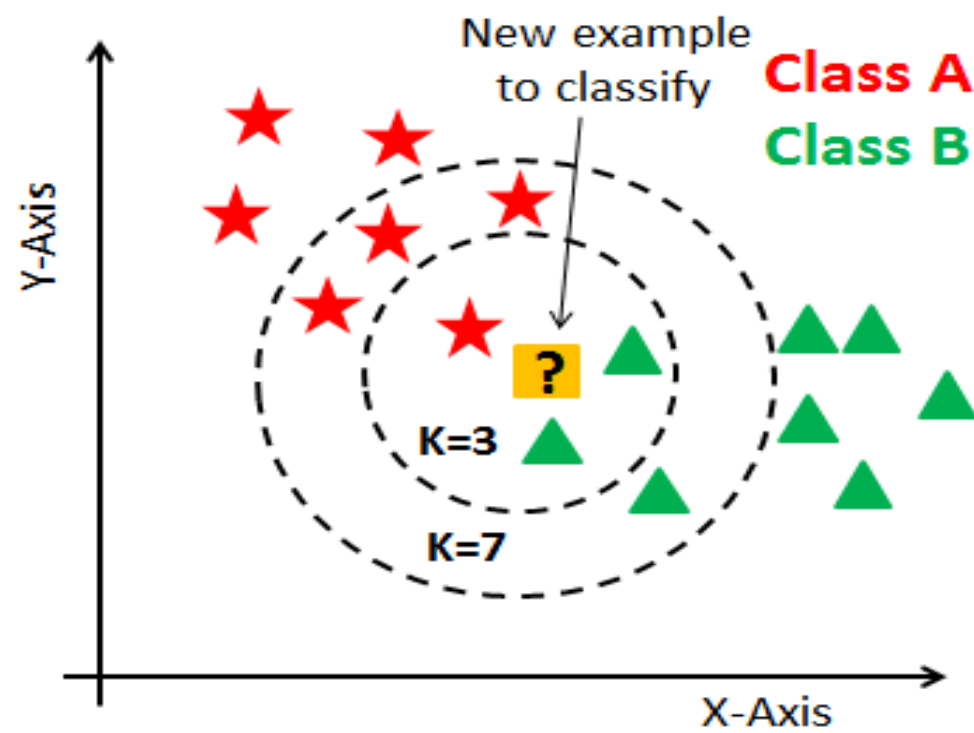
Step-1: Choose the number K.

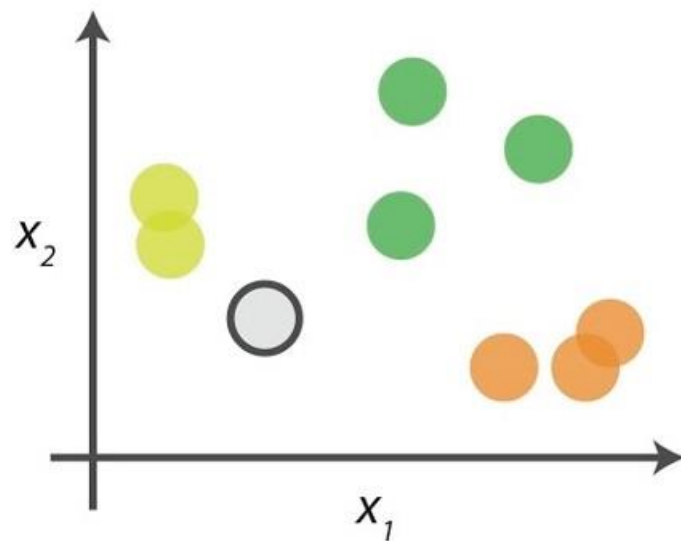
Step-2: Calculate the Euclidean distance of the neighbors from the new data.

Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.

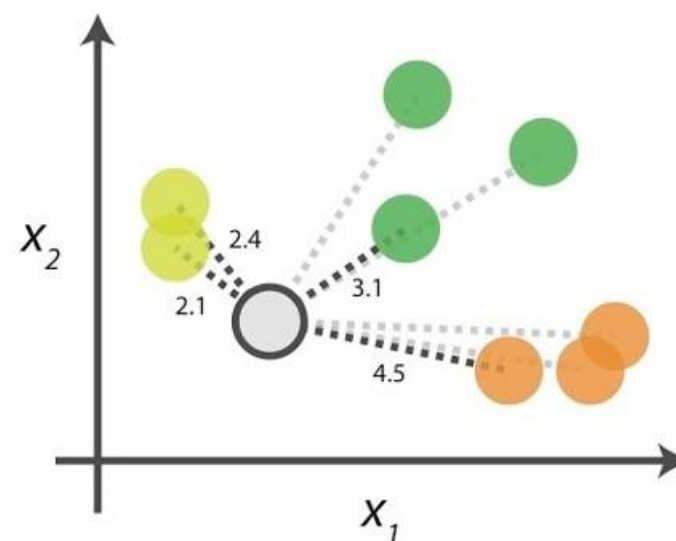
Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.













Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.



Start by calculating the distances between the grey point and all other points.

<i>Point</i>		<i>Distance</i>	
		2.1	→ 1st NN
		2.4	→ 2nd NN
		3.1	→ 3rd NN
		4.5	→ 4th NN

Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

<i>Class</i>	<i># of votes</i>	
	2	→
	1	
	1	
		Class  wins the vote!
		Point  is therefore predicted to be of class  .

Thanks