

# KNN Report

Donal Loitam

July 20, 2022

## Contents

<b>1 Introduction</b>	<b>1</b>
<b>2 Key Points of the Algorithm</b>	<b>2</b>
<b>3 Some Questions</b>	<b>4</b>

## 1 Introduction

- K-NN algorithm is one of the easiest algorithm which can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity with the available datas

### Example:

Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



Figure 1: Example

## 2 Key Points of the Algorithm

Let's take a simple case to understand this algorithm. Suppose there are two categories : Following is a spread of orange circles and green circles . Now, suppose we have a new data point  $x_1$ , so we have to classify this data point in one of the categories based on it's similarity. We can use K-NN to accomplish this classification task :

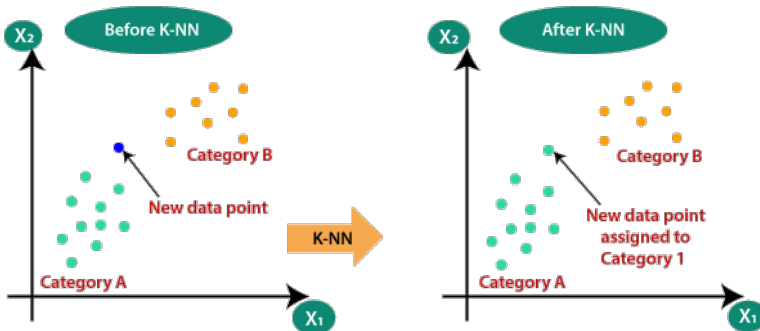


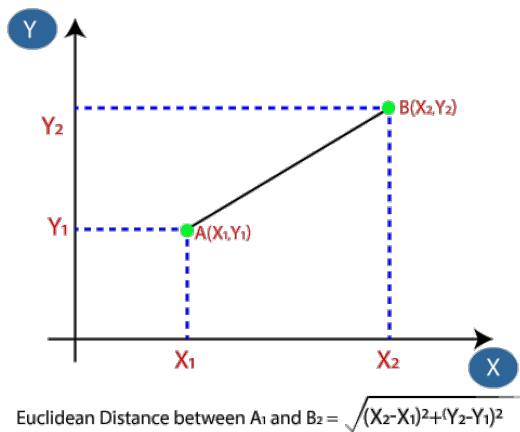
Figure 2: Classification using K-NN

**But How does it work ?**

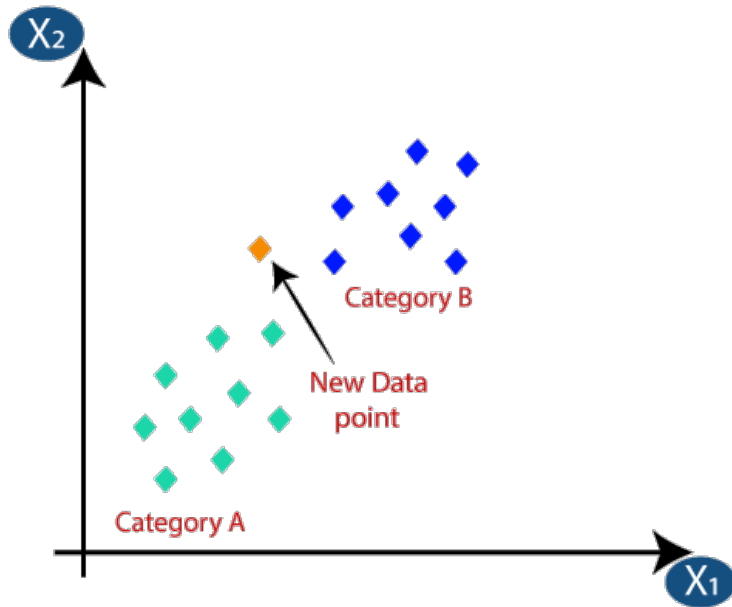
The K-NN working can be explained on the basis of the below algorithm:

- Select the number 'K' of the neighbors.(how to know what to select is discussed later)
- Calculate the Euclidean distance of the new test obseravtion from all the observation of the training dataset
- Take the K nearest neighbors as per the calculated Euclidean distance.
- Among these k neighbors, count the number of the data points in each category.
- Assign the new data points to that category for which the number of the neighbor is maximum.

**Euclidean distance:** The Euclidean distance is the distance between two points calculated as:



Suppose we have a new data point and we need to put it in the required category. Consider the below image:



By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B. Consider the below image:



As we can see the 3 nearest neighbors are from category A compared to 2 from category B, hence this new data point must belong to category A.

### 3 Some Questions

#### 1. How to select the value of K in the K-NN Algorithm?

**Ans** In KNN, finding the value of k is not easy. A small value of k means that noise will have a higher influence on the result and a large value make it computationally expensive

There are no proper methods to find the **K** value but you can try the following :

- Square Root Method: Take the square root of the number of samples in the training dataset and assign it to the K value.
- A tip is to keep the value of N to be odd so that there is no tie between choosing a class. But, then again it points to the fact that training data is highly correlated between classes and using a simple classification algorithm such as k-NN would result in poor classification performance.

#### 2. What are the cons of KNN?

**Ans.** Some of the cons of KNN algorithm are as follows :

- Does not work well with large datasets
- Does not work well with high dimensions : KNN algorithms generally do not work well with high dimensional data since, with the increasing number of dimensions, it becomes difficult to calculate the distance for each dimension.
- Sensitive to Noise and Outliers :KNN is highly sensitive to the noise present in the dataset and requires manual imputation of the missing values along with outliers removal.

#### 3. Why do you need to scale your data for the k-NN algorithm?

**Ans** Imagine a dataset having n number of instances and N number of features. There is one feature having values ranging between 0 and 1. Meanwhile, there is also a feature that varies from -999 to 999. When these values are substituted in the formula of Euclidean Distance, this will affect the performance by giving higher weightage to variables having a higher magnitude

#### 4. Why should we not use the KNN algorithm for large datasets?

**Ans. The Problem in processing the data:**

KNN works well with smaller datasets because it is a lazy learner. It needs to store all the data and then make a decision only at run time. It includes the computation of distances for a given point with all other points. So if the dataset is large, there will be a lot of processing which may adversely impact the performance of the algorithm.

#### **Sensitive to noise:**

Another thing in the context of large datasets is that there is more likely a chance of noise in the dataset

#### 5. How to handle categorical variables in the KNN Algorithm?

**Ans** To handle the categorical variables we have to create dummy variables out of a categorical variable and include them instead of the original categorical variable. Unlike regression, create k dummies instead of (k- 1).

For example, a categorical variable named “Degree” has 5 unique levels or categories. So we will create 5 dummy variables. Each dummy variable has 1 against its degree and else 0.