**How to classify data set into train and test data**

1st step:

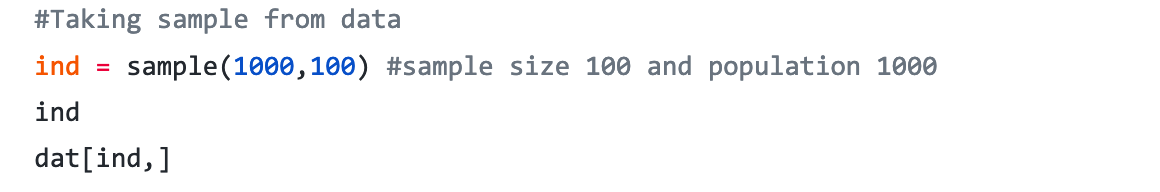
Import data from working directory





2nd step:

Generate random number from data say size of 100 and put this sample on a variable called ***ind.*** And select data by generating random number.

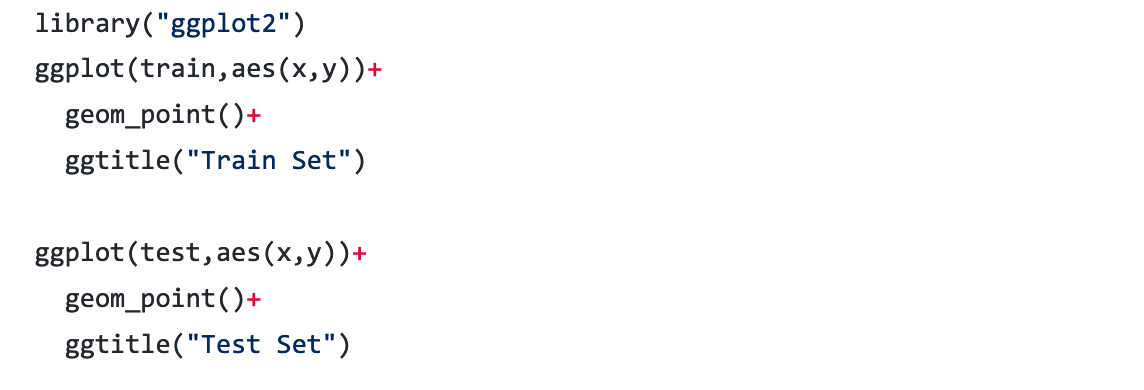


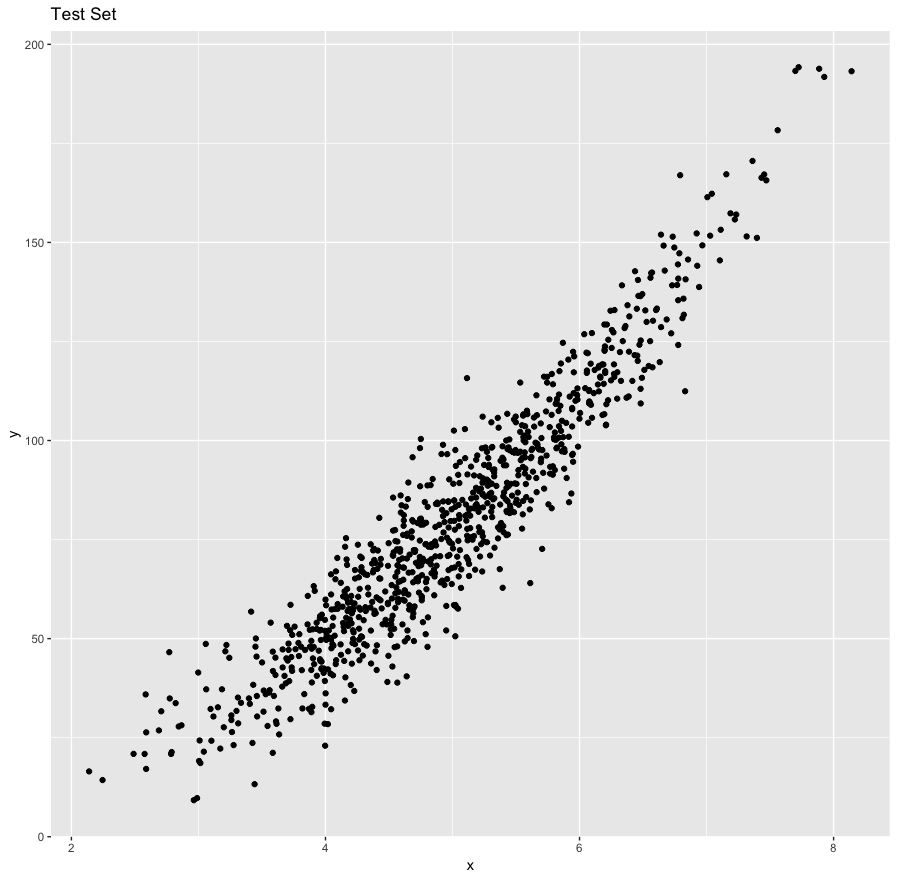
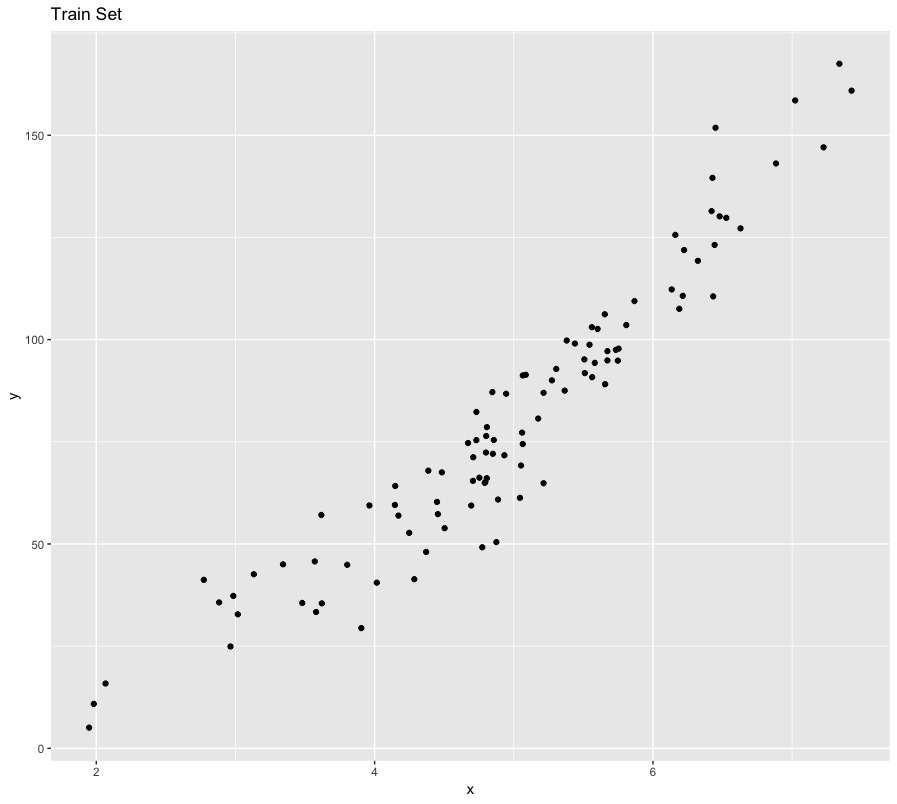
3rd step:

Put generated sample in train variable and rest are in test variable.



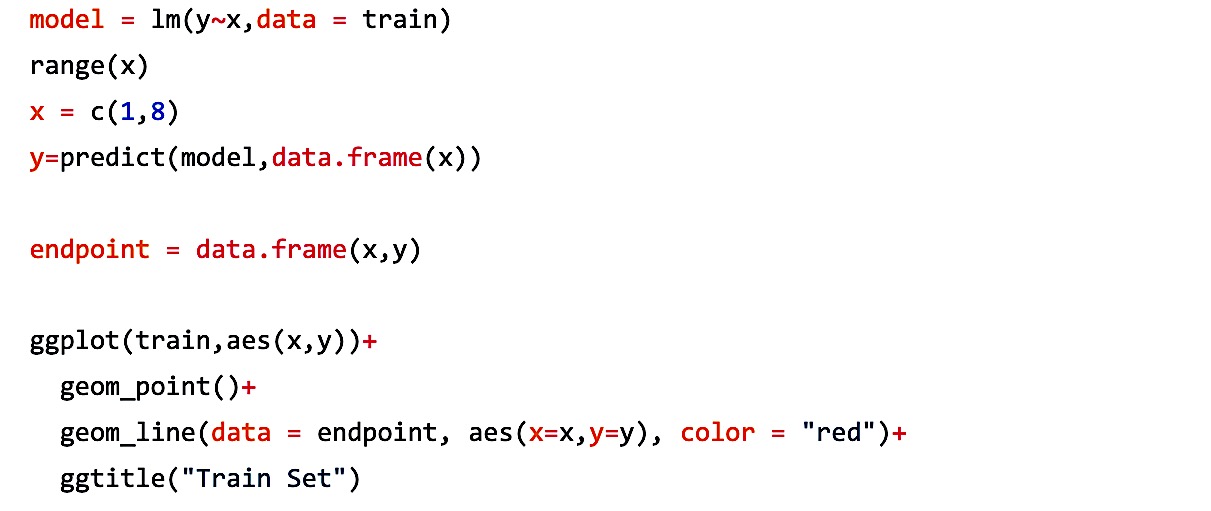
Plotting data:

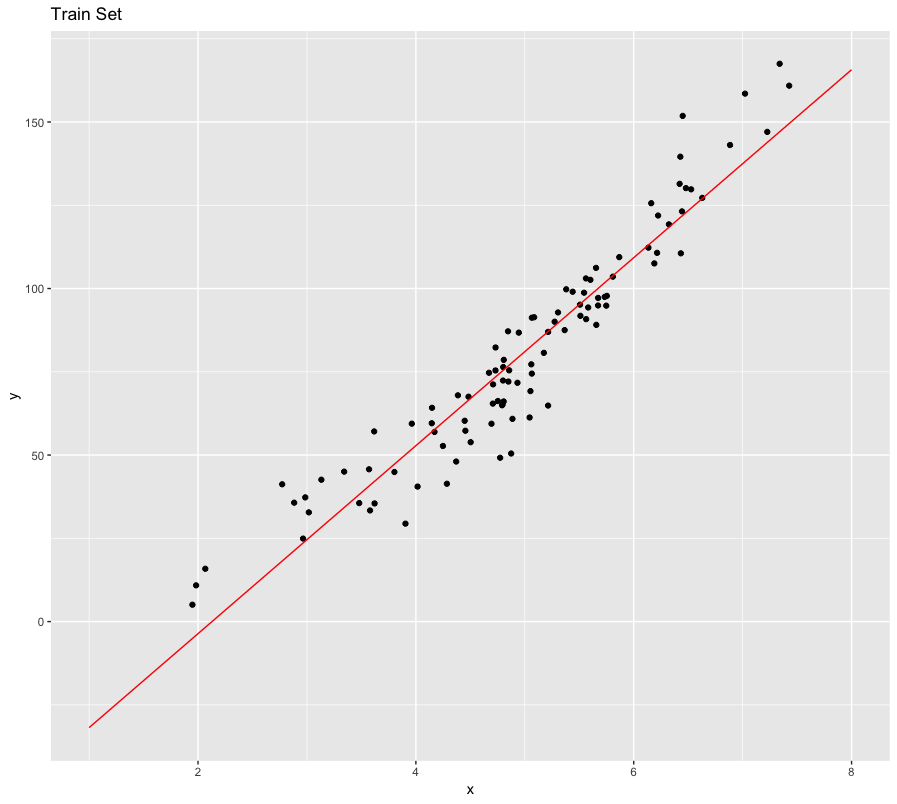


Plotting Least Square Fitted Line

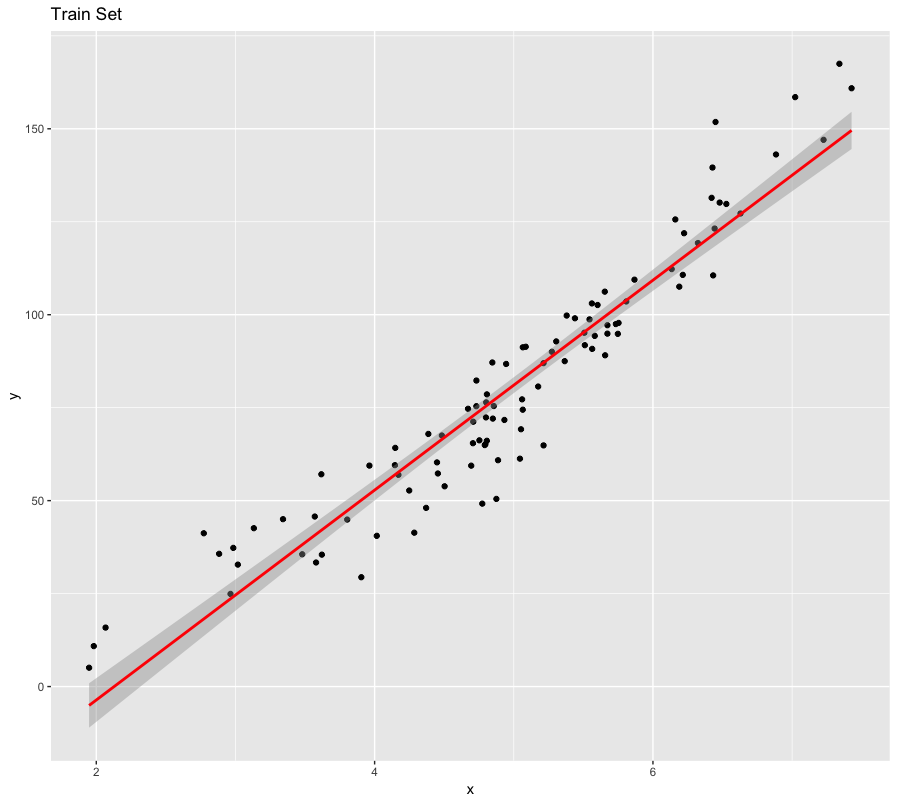
1. We can create model for train data by lm function. Then we have to create a data frame which contains predicted value of 1st point and endpoint of data and plot line graph to these points.





1. We can do this by a simple code only





**kNN Algorithm Implementation using R Programming**

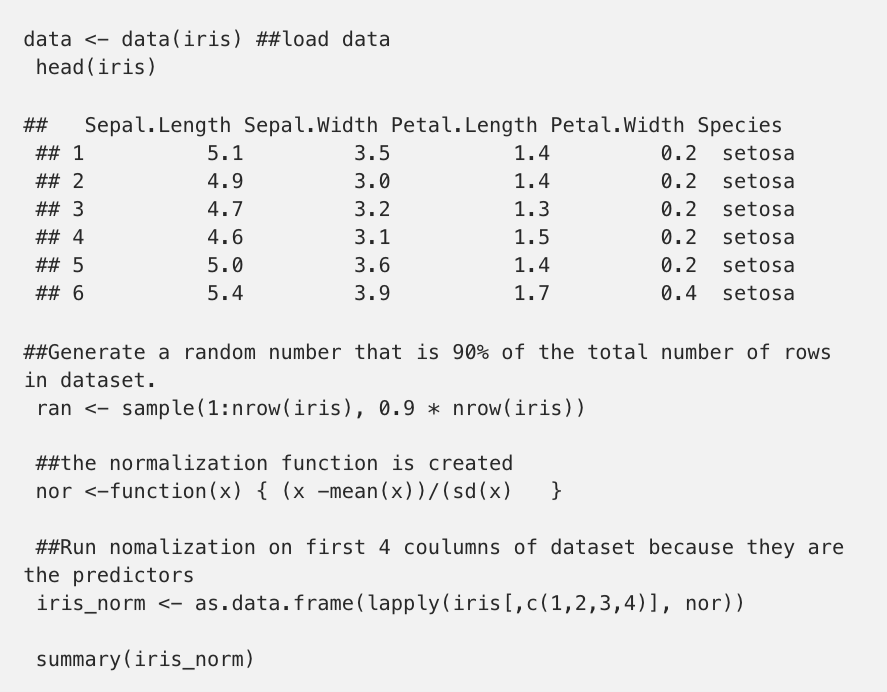
The KNN or k-nearest neighbor’s algorithm is one of the simplest machine learning algorithms and is an example of instance-based learning, where new data are classified based on stored, labeled instances.

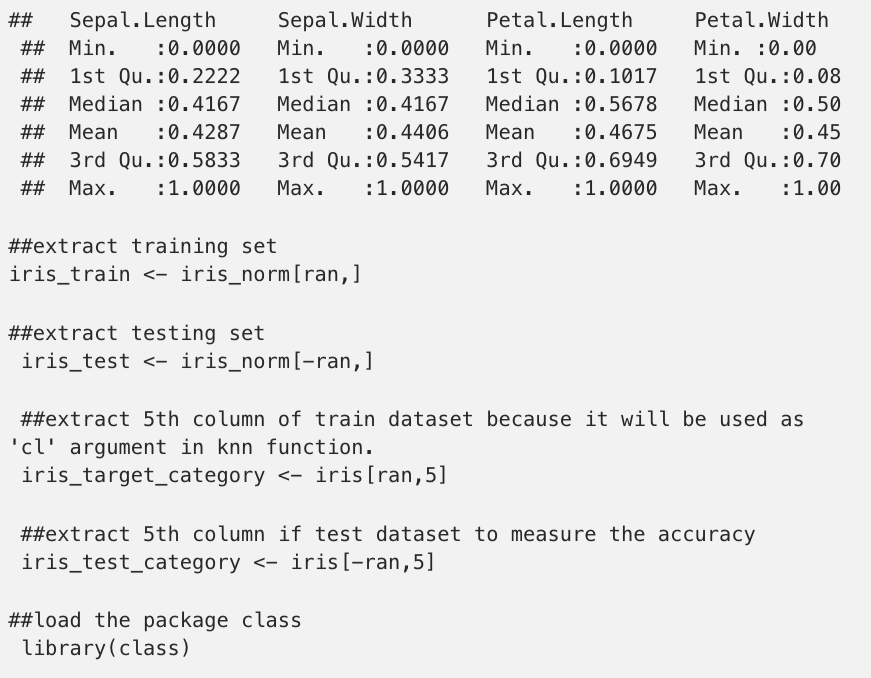
More specifically, the distance between the stored data and the new instance is calculated by means of some kind of a similarity measure. This similarity measure is typically expressed by a distance measure such as the Euclidean distance, cosine similarity or the Manhattan distance.

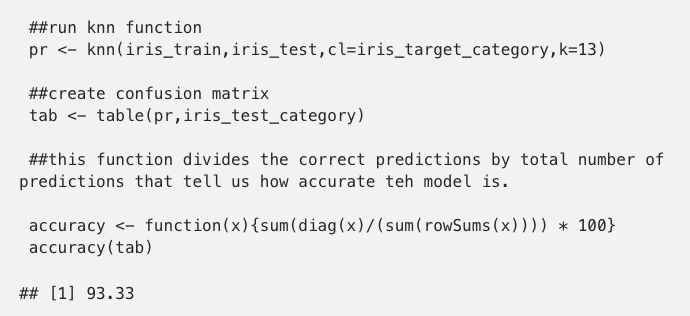
Let’s say we are given certain number of objects and each object has its own unique attribute associated with it, for example, you have 5 chairs 10 beds and 15 tables and each for each one of them we know the length, breadth and height. Now, if someone give us a new object with new attributes and ask us to predict as to which category does that new object belongs to, that means we are given the dimension and asked to predict if it is a chair, bed or table, then we would have to use knn algorithm to determine that.



In the iris dataset that is already available in R, I have run the k-nearest neighbor algorithm that gave me 93% accurate result







In the cancer dataset that is already available in my pc, I have run the k-nearest neighbor algorithm that gave me 84% accurate result

