ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 12, December 2017

Comparision of K-Means and Knn Algorithm in Data Mining

[1] S.R.Kalaiselvi, [2] C. Karpagam
[1] Assistant Professor Dept. of Computer Science,
[2] Dr.N.G.P College of Arts and Science, Coimbatore.

Abstract: **- Data mining is the procedure for analyzing data from different perspective and shortening it into helpful information. It can be used to increase income, minimize the costs. Data mining software is one of the analytical tool for analyzing data. It allows users to examine data from many different proportions, classify it, and review the relationships identified. Theoretically, data mining is the process of ruling correlations or patterns among dozens of fields in huge relational databases. Nowadays, organizations are accumulating vast and growing amounts of data in different formats and different databases. A data mining algorithm is a set of calculations which creates a data mining model from data. To build a model, the algorithm first analyzes the data and it look for particular type of pattern. These algorithms use the outcome of the analyzed data to define the most favorable parameters for creating the mining model. K-means is the unsupervised learning algorithm and it is an incremental approach to clustering data dynamically adds one cluster center at a time through a deterministic global search procedure. It is a simple and easy way to classify a given data set through a certain number of clusters. The k-Nearest Neighbors algorithm (or k-NN for short) is a non parametric method used for classification and regression. In k-NN algorithm neighbors are taken from a set of objects for which the class (for k-NN classification) otherwise the object property value (for k-NN regression) is identified. This can be consideration of as the training set for the algorithm, though no explicit training step is necessary.**

Keywords: - Data - Information – Relationships – Analytical tools – Clustering Algorithm – groups - centroid – Data Frame – distance metrics - Prediction – Euclidean Distance – Neighbour-Dimenions–GetAccuracyfunction.

I. INTRODUCTION

K-means clustering algorithm mechanism top once the input data is mostly numeric. We can take a model of clustering, consider an analysis of super market shopping actions based on reliability certificate data. Basically take each customer and create a pasture for the total amount purchased in various departments in the shop over the course of some period of time. This data is all numeric, so work with it quite easily.

HOW K-MEANS CLUSTER DETECTION WORKS

K-means clustering is a type of unendorsed learning, which is used when you have unlabeled data (i.e., data without defined categories). The aim of this algorithm is to find out groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are:

1. The centroids of the K clusters

2.Labels for the training data The algorithm then iterates between two steps:

1. Data task step:

Each centroid defines one of the clusters. In this step, each data point is assign to its nearest centroid, based on the squared Euclidean distance.

2. Centroid update step:

In this step, the centroids are recomputed.

HOW TO IMPLEMENT K-NEAREST NEIGHBORS IN PYTHON

STEPS

1. Grip Data

The first item we want near do is set our data file. The data is in CSV plan lacking a heading line or any citation script. We protect open the file with the open function and read the data lines using the reader function in the CSV module. Next we have to split the data into a training dataset that kNN can use to make prediction and a test dataset that we can use to evaluate the precision of the model. We first need to convert the flower procedures that were loaded as strings into numbers that we can work with. Next we require to split the data set randomly into arrange and datasets.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 12, December 2017

2. Comparison
In order to make prediction we require calculating the connection between any two given data instances. This is needed so that we can put the k most related data instances in the training dataset for a given member of the test dataset and in turn make a prediction. Given that all four flower dimensions are numeric and have the same unit, we can straight use the Euclidean space measure. This is defined as the square root of the sum of the squared differences between the two arrays of numbers (read that again a few times and let it sink in).Additionally, we want to control which fields to include in the distance calculation. In particular, we only desire to include the first 4 attributes. One advance is to limit the Euclidean distance to a fixed length, ignore the final dimension.

3. Neighbors
Now that we have a comparison measure, we can use it collect the k most similar instance for a given unseen instance. These are a directly forward process of manipulative the distance for all instances and select a subset with the minimum distance values.

4. Reaction
Once we have placed the most similar neighbors for a test instance, the next task is to devise a predict reply based on those neighbors. We can do this by allowing each neighbor to vote for their class quality, and take the majority vote as the estimate. Below provides a function for getting the majority selected reply from a number of neighbors. It assumes the class is the last aspect for each neighbor. This approach returns one response in the case of a draw, but you could feel such cases in a specific way, such as frequent no response or select an balanced arbitrary response.

5. Accuracy
We have all of the pieces of the kNN algorithm in place. An significant left over concern is how to evaluate the accuracy of predictions. An easy way to assess the accuracy of the model is to calculate a ratio of the total correct predictions out of all predictions made, called the classification accuracy. Below is the getAccuracy function that sums the total correct predictions and returns the accuracy as a percentage of correct classifications.

6. Core
We now have all the elements of the algorithm and we can tie them together with a main function.

II. CONCLUSION

In short, the algorithms are annoying to complete special goals. K-nearest neighbor is a split of supervised learning classification algorithms. It is supervise because you are difficult to classify a point based on the known classification of other points. In contrast, K-means is a subset of unsupervised learning clustering algorithms. It is unsupervised because the points have no external classification. The in each case mean different things. In K-NN, the represents the number of neighbors who have a vote in determining a new player's position. The in K-means, determine the number of clusters we want to end up.

REFERENCES

1. Tan, Steinbasc & Kumar. Introduction to Data Mining. 2006.

2. Zaki & Meira. Data Mining and Analysis Fundamental concepts and Algorithms. 2014

3. Cluster Detection. Retrieved from tuoitre.mobi/tu-khoa/k-means-clustering-example-631198.html

4. K Nearest Neighbors. Retrieved from https://machinelearningmastery.com/tutorial-to-implement-k-nearest-neighbors-in-python-from-scratch/