TV PROGRAM POPULARITY PREDICTION USING K-MEDOID

[1]Adya Ananya Das, [2]Ajit Kumar Pasayat
[1]M.Tech Scholar, [2]Asst.Prof
[1]Computer Science,
[1]Centurion University, Jatani, India

Abstract: **Random Forest is a supervised learning algorithm. Like we can already see from it's name, it creates a dense forest and makes it random. The "forest" it builds, is an decision of Decision Trees, most of the time trained with the different methods. In my paper I m using random forest method and K-Medoid method for prediction.**

***Index Terms* - popularity prediction, random forests regression, K-Medoid method, Prediction.**

I. INTRODUCTION

In general, to broadcast information is to transmit it to many receivers. For example, a radio station broadcasts a signal to many listeners, and digital TV subscribers receive a signal that is broadcast by their TV provider. In computer networking, broadcasting is the process of sending data packets to multiple recipients all at once. For instance, a local area network can be configured so that any device on the network can broadcast a message to all the others.[2] Every classifier evaluation using ROCR starts with creating a prediction object. This function is used to transform the input data (which can be in vector, matrix, data frame, or list form) into a standardized format. 'predictions' and 'labels' can simply be vectors of the same length. However, in the case of cross-validation data, different cross-validation runs can be provided as the *columns* of a matrix or data frame, or as the entries of a list. In the case of a matrix or data frame, all cross-validation runs must have the same length, whereas in the case of a list, the lengths can vary across the cross-validation runs. Internally, as described in section 'Value', all of these input formats are converted to list representation. Since scoring classifiers give relative tendencies towards a negative (low scores) or positive (high scores) class, it has to be declared which class label denotes the negative, and which the positive class. Ideally, labels should be supplied as ordered factor(s), the lower level corresponding to the negative class, the upper level to the positive class.[3]

II. PROBLEM FORMULATION

Using the prediction method we can predict the future. But for predicting we can use various types of methods.
For prediction we can use various types of methods. In this paper I m using 3 steps to perform prediction having better accuracy.
Stewart (2000) states the problem succinctly: Every prediction contains an element of irreducible uncertainty … actions that are based on predictions lead to two kinds of errors. One is when an event that is predicted does not occur, i.e., a false alarm. The second is when an event occurs but is not predicted, i.e., a surprise. There is an inevitable tradeoff between the two kinds of errors; steps taken to reduce one will increase the other. This article examines the effects of false alarms and proposes two classifications of false alarms. Further, some rudimentary estimates of the costs of false alarms are presented. Two Types of False Alarms To use tornado warnings as an example, if one were to turn the clock back fifty years, tornado warnings were in their infancy.[4]
For all these process I have used K- Medoid method using R language.
K mean method can also be used in this case, but mean is completely different from medoid.
Mean means sum of all the data divided by no. of data present where as Medoid means the minimal distance from all the data sets.

III. ALGORITHM

K-Medoids Based on the DTW Algorithm (KMDTW(*D*;*C*))[2]

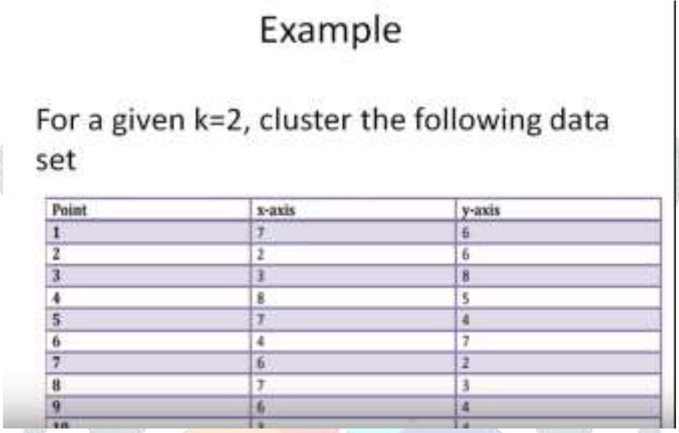
```
1. $D$: the data set containing program popularity time series
2. $C$: the number of trends
3. $K$: the set of trend centers
4. $M$: the set of popularity sequences in each trend
5. initialize $C$ as trend centers of $K$
6. **do**
```

```
for i D 1:size(D)
    for k D 1:K
        DistDi;Ck D DTW(Di;Ck )
    end for
    if(DistDi;Ck is min)
        assign Di into Mk
    end if
end for
while(the cluster membership changes)
return K;M
```

K-Medoid Algorithm Explanation:

Let's take a table

Figure 1: - K-Medoid example [5]



Example

For a given k=2, cluster the following data set

Point	x-axis	y-axis
1	7	6
2	2	6
3	3	8
4	8	5
5	7	4
6	4	7
7	6	2
8	7	3
9	6	4
10	3	4

Randomly we have to choose the medoids.

Let's assume (3,4) & (7,4) are the medoids.

Now we have to measure distance of all points from the choosen medoid points.and we have to find the minimal distance and put them into same cluster.

1) from (7,6) to (7,4) = (7-7+6-4) = 2
2) from (7,6) to (3,4) = (7-3+6-4) = 4+2 = 6

So (7,6) goes under (7,4) .

Like this we have to find all the clusters.

So now after the clustering the clusters formed are : {(3,4),(2,6),(3,8),(4,7)}and {(7,4),(6,2),(6,4),(7,3),(8,5),(7,6)}

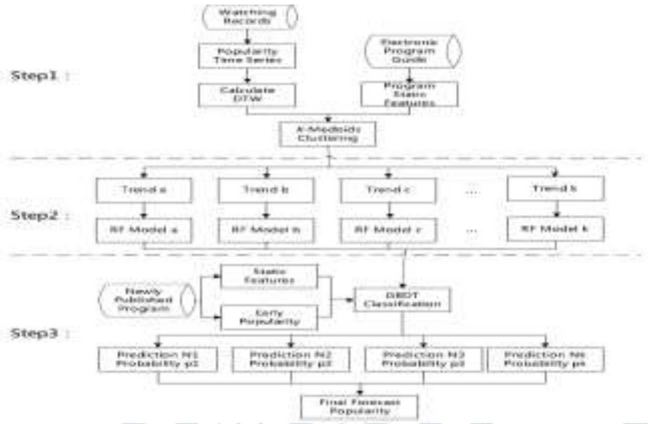
Now calculating cost :

Total cost=cost((3,4),(2,6))+cost((3,4),(3,8))+ cost((3,4),(4,7))+cost((7,4),(6,2))+cost((7,4),(6,4))+ cost((7,4),(7,3))+ cost((7,4),(8,5))+ cost((7,4),(7,6)) = 3+4+4+3+1+1+2+2=20

3.1 RELATED WORK

By using K-Medoid Method we can predict about the TV broadcasting. In R when the number of clusters varies then the experiment results varies. For Example when the number of clusters are 3 , then the graph is different than the number of clusters is 4 or 5.

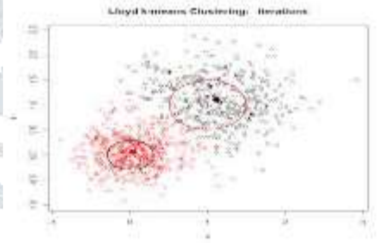
Figure 2: - *Overview of the broadcast TV program popularity prediction method.* [6]



We can also use BER method which is also called as Bit Error Rate method for prediction but it gives the result bit wise which is comparably slower and less correct than the result produced by using the method K- Medoid. In short we can say the accuracy of K-Medoid is better than other some methods.This is very confusing sometime. when number of signals increases then the graph also becomes very confusing. One major fact is in-case of K- Medoid it works in clusters where as bit error rate method works in bit. Only for an example to compare let’s have a look at graph of K-Medoid using clusters.

Using 2 components graphs are -

Figure 3: K- Mean iteration graph [7]



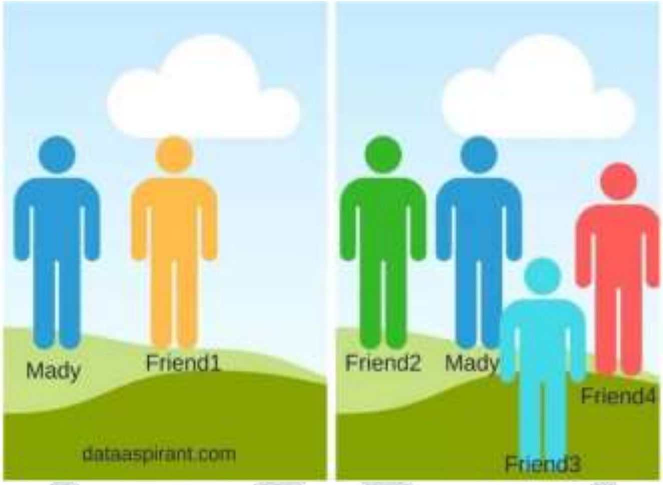
3.2 RANDOM FOREST

Random forest algorithm is a supervised classification algorithm. As the name suggest, this algorithm creates the forest with a number of trees.[8]

As we know the density of the forest depends upon the number of trees in it. In the same way the random forest classifier accuracy also depends upon the number of trees it has. Higher the number of trees in the forest more is the accuracy.

Random forest tree can be better understood by taking the cases happening in our real life as explained below.

Figure 4: Random forest real life example[8]



In summer vacation Mady wants to go to abroad. But he can't decide the exact place where to go. So he firstly decided to ask his best friend (friend 1, shown in above figure) among all friends. After listening to Mady's question his best friend asked him "where have you been previously?" .Based on Mady's reply he suggested him a place.

After his best friend , Mady asked his other friends for their suggestions. All of them gave different types of suggestions according to their point of views by asking several types of questions to Mady.

Finally Mady took his own decision by looking into the majority votes of his friends.

Here technically two algorithms are performed.

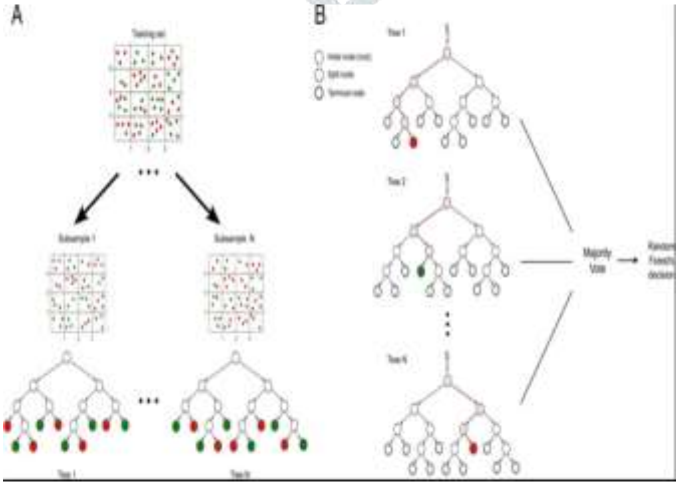
Decission tree
Random forest tree

Choosing Mady's best friend 1st for asking for suggestion among all of the friends is DECISSION TREE ALGORITHM & finally selecting where to go by looking into majority votes of all his friends is RANDOM FOREST ALGORITHM.

3.2.1 USAGE OF RANDOM FOREST

After looking into the graph below ,we can explain why using random forest is easier and wise.

Figure 5: random forest graph[9]



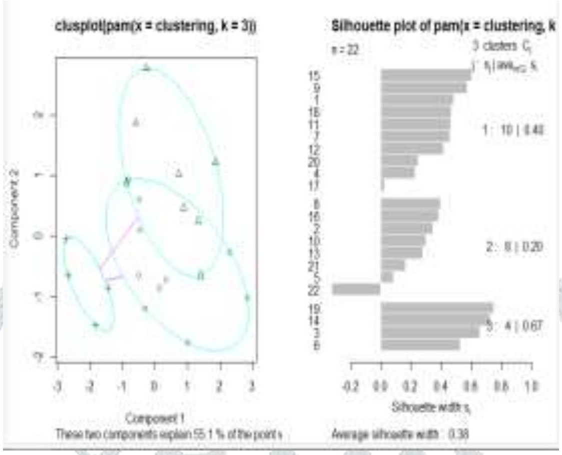
Random forest used to build multiple numbers of decision trees first. In the next step it merges all of them together to get a result having more and better accuracy by which we can perform a stable prediction.

IV. RESULTS AND DISCUSSIONS

Experiments changes according to numbers of clusters. First let's take 3 numbers of clusters and the output is given below.

```
//pam.res <- pam(clustering, 3)
table(pam.res$clustering.clustering$Company)
layout(matrix(c(1,2),1,2))  #2 graphs for pageplot(pam.res)
```

O/P :-



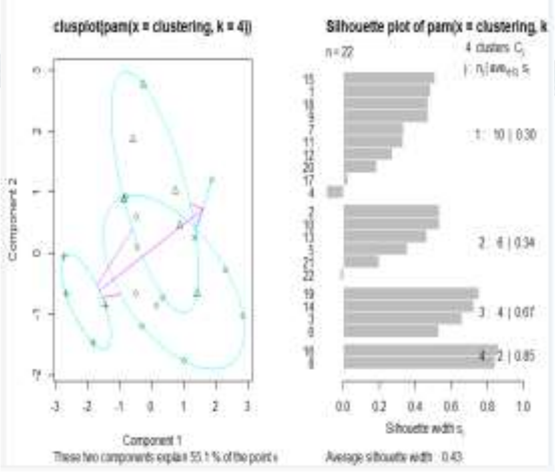
clusplot(pam(x = clustering, k = 3))

Silhouette plot of pam(x = clustering, k

Let the number of clusters be 4

```
pam.res <- pam(clustering, 4)
table(pam.res$clustering.clustering$Company)
layout(matrix(c(1,2),1,2))  #2 graphs for page
plot(pam.res)
```

O/P :-



clusplot(pam(x = clustering, k = 4))

Silhouette plot of pam(x = clustering, k

V. ACKNOWLEDGEMENT

Here we have used K-Medoid Method for the prediction of the TV broadcasting. In this method mostly clusters are used. First of all the data are divided into clusters and the prediction is completely dependent upon the results given by the clusters. Output or result changes according to the number of clusters. In future this work can be extended by using other machine learning approaches and focusing on other criteria like actor based, actress based, director based, production house based etc. **References**

[1]Niklas Donges , SAP Machine Learning Foundation Working Student CODE University of Applied Sciences, Berlin https://machinelearning-blog.com
Feb 22,online: https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd

[2]Computer Hope, Free Computer Helps Since 1998,Updated: 08/08/2017 by Computer Hope

[3]R Documentation ,online: https://www.rdocumentation.org/packages/ROCR/versions/1.0-7/topics/prediction

[4]The moral problem (1994), by Michael R. Smith , Weatherdata Incorporated

[5]Red Apple Tutorials, Simplest Example of K Medoid clustering algorithm, Published on Oct 6, 2017

[6]SPECIAL SECTION ON CONVERGENCE OF SENSOR NETWORKS, CLOUD COMPUTING, AND BIG DATA IN INDUSTRIAL INTERNET OF THING, Received September 27, 2017, accepted October 20, 2017, date of publication October 27, 2017, date of current version November 28, 2017.

[7] Revolutions, Daily news about using open source R for big data analysis , predictive modeling, data science & visualization since 2008.

[8] Saimadhu Polamuri,how the random forest algorithm works in machine learning, published on May22,2017.

[9] sandeep agrawal, exploring survival on the titanic in machine learning, published on 27th sep,2016.