Article can be accessed online at http://www.publishingindia.com

Clustering Trend Predictions using Evolutionary k-means Algorithm for Automated Clustering

Jyoti Lakhani*, Dharmesh Harwani**

Abstract

The paper proposed a method of hybridization of k-means algorithm and evolutionary programming. The blend of the two generates k number of clusters C = (c1, ..., ck) in the data space D = {x1, ..., xn}. These clusters will evolve in such a way that prediction of the upcoming trends of clusters in the application is possible. The proposed hybrid is named as evolutionary k-means clustering algorithm which is useful in generating and predicting clustering trends in an automated system.

Keywords: Clustering, Data Mining, Evolutionary Programming, K-Means

1. Introduction

Clustering is a data mining technique of data analysis. Clustering is used to group objects on the basis of similarity; resulting similar type of objects in one cluster. The k-means algorithm is a clustering algorithm uses minimum distance to find out the statistical clustering patterns. In this paper, the k-means algorithm is hybridized with the evolutionary programming to automate the process of finding clusters in a given data set. This is useful for detecting the up-coming probabilistic clusters for automated systems.

2. The k-means Algorithm

The k-means algorithm is a clustering algorithm which defines clusters on the basis of minimum distance. From a given data points for k number of points which denote the centres of k clusters are selected randomly. By considering these points as centres of clusters, other points are also selected by following the minimum distance criteria (Gaussian distance). This process will be repeated until all clusters are created having all the data points with minimum distances. (MacQueen, 1967)

3. Evolutionary Programming

Evolution is a continuous process of changing environment by selecting and fittest individuals from the population. Only the fittest will get the chance to reproduce and generate its successors. This process is called natural selection (Darwin, 1859). Based on the concept of natural selection and evolution, evolutionary programming is introduced by. Fogel in 1966 (Fogel, 1966). The classical evolutionary programming proposed by Fogel uses finite state automata for machine learning tasks such as prediction. To achieve the adaptive behaviour by a machine, prediction of the environment is must. The predictive behaviour is achieved by recombination and mutation operator. In a typical evolutionary programming, it is required to generate an initial population by random selection of the genetic content. Genetic content resides on the chromosomes and defines the behaviour of the individuals. A value of fitness is assigned to the genetic content. In order to produce the next population the chromosome with highest fitness value of genetic content is given a chance to reproduce offspring. The reproduction is implemented by one of the three possibilities copy, crossover and mutation. This process will follow that

* Department of Computer Science, Maharaja Ganga Singh University, Bikaner, Rajasthan, India.
E-mail: jyotilakhanimgsu@gmail.com
** Department of Microbiology, Maharaja Ganga Singh University, Bikaner, Rajasthan, India.
E-mail: dharmesh_harwani@hotmail.com

fittest individual has higher probability of getting selected for reproduction. If a newly generated population contains the fittest individual, it switches from or terminated. If not then the process will continue until the fittest is met.

In the automated clustering process, proposed in the current communication, we need to have certain parameters for prediction. Once clusters are generated using specific data points, clusters would evolve to be more adaptive to the space. Evolutionary programming can be used to access the dynamic population of the clusters which are more stable in space.

The Proposed Algorithm

Algorithm Name: Evolutionary k-means algorithm for automated clustering

Input: Assuming that *n* spatial data points are available to implement the evolutionary clustering D = {*x1, x2, ..., xn*}. The task of algorithm is to find out k number of clusters C = {*c1, c2, ... , ck*}. Let *r(k)* be the k number of random data points selected from the data set D, representing the centres/centroids of the initial cluster set.

Output: The output of the present algorithm is a set of k number of clusters where each cluster is fulfilling the fitness criteria.

Starting Criteria: Algorithm will start automatically when

- There is a change in any cluster or its data point, which may cause a data point to coupele loosely with its parent cluster.
- When there is an indication of un-match fitness criterion.
- When there is a change in fitness criteria itself.
- When one wants to increase data points in a given population automatically for the purpose of predictions.
- When one needs to moderate clusters automatically for future prediction analysis

Fitness Criteria: Clusters are created such as that they meet the specific fitness criteria. The fitness criteria can be the distance between each data point of a cluster or compactness of the clusters.

Termination of algorithm: When all clusters in a given space meet the fitness criteria, the algorithm terminates automatically.

Steps of algorithm: This algorithm follows the steps 1 -2 if no initial population of cluster is available or if continues with the available data points. If clusters are already available in the data space, other subsequent steps of algorithm will be executed.

[Part (i)]

1. [Find out the initial population of clusters]
   - A. eneration=0
   - B. for each centre k in r(k),
     - i. for each nearest data point in D,
     - a. Check whether this data point meets the fitness criteria for the centre, if yes add this data point to the C.
   - C. Save set of clusters C = {c1, ..., ck} in the current Generation.
2. [Check whether the clusters meets fitness criteria]

If yes, find out the mean of each data points of each cluster to find out its centroid else go to the step 1.

[Part (ii)] [Reproduction]

3. [To populate clusters automatically with data points of highest fitness]

In order to increase the size of the cluster with fittest members to strengthen the clusters follow the steps below.

- A. Generation = Generation +1
- B. For each cluster in C = {c1,..., ck}
  - i. Select the fittest member (nearest to the centroid)
  - ii. Copies it and add it to the cluster
  - iii. Add new data points to the D.
- C. Save set of clusters C = {c1, ..., ck} in the current Generation

This step will strengthen the clusters as after performing this step, the cluster will have increase number of data point to meet fitness criterion.

[Part (iii)] [Cross over]

4. [To populate clusters with probabilistic data point for the purpose of prediction]

To incorporate probably new data points to the clusters continue with the following steps. The data points generated by this step may or may not be new-

A. Generation = Generation +1

B. For each cluster in C= {c1,..., ck}

i. Select two fittest data points from the cluster

ii. Find out the mean of these data points

iii. Create a new data point with this mean and add it to the cluster

iv. Add this new data point to the set D

C. Save set of Clusters C = {c1, ..., ck} in the current Generation

[Part (iv)] [Mutation]

5. [To populate clusters with vigorous data point that may not be previously incorporated in the clusters]

In order to increase the vigour in the cluster, continue with the following steps. These steps incorporate new data points to the clusters-

A. Generation = Generation +1

B. For each cluster in in set C = {c1, ... , ck}

i. Find out the data points with lowest fitness (LF) and highest fitness (HF).

ii. Generate a random data point within the range of LF and HF.

iii. Add this newly generated data point to the cluster

iv. Add this newly generated data point to the set D.

C. Save set of clusters C = {c1, ..., ck} in the current Generation

6. [Termination Step]

If each data point in each clusters have met the fitness criteria and there is a user intervention indicating the termination requirement-

Return C = {c1, c2, ..., ck} for each Generation

Explanation

The evolutionary k-means algorithm for automated clustering starts with n number of data points in the set D={x1,..., xn}. The algorithm clusters these data points in k number of clusters of the set C={c1, ..., ck}. The process is started by selection of k number of random points represented as r(k), function as the centres of clusters. The algorithm has been divided into four different parts. When there is no evidence of clusters in the data space, the part (i) of the algorithm will be executed. Part (i) of algorithm comprises of step 1 and 2. This part of algorithm is responsible for creating fresh clusters for a given data set. Once clusters are generated using part (i) of the algorithm, they will automatically strengthen or change so as to predict future trends of clustering in a given data set. This will be done by other parts of the algorithm. The part (ii) (step 3) will be triggered if we need to strengthen the clusters. This is implemented in the algorithm using the concept of reproduction in evolutionary programming. This part of the algorithm will produce copy of the fittest data points to strengthen the clusters. Copies of fittest data points increase the ratio of fit and un-fit data points in the cluster. Increase in fitness reflects increase in the strength of cluster. The part (iii) (step 4) and part (iv) (step 5) are responsible for an increase in cluster vigour. Part (iii) is working as a crossover process and part (iv) is reflects a mutation process of evolutionary programming. Part (iii) selects two fittest elements to cross over and recombine by acquiring mean of these two values. The value will probably be similar to one of the two selected fittest values. In part (iv) one random value will be generated and added to the cluster to increase cluster vigour. The algorithm will be terminated when the system fulfils the fitness criteria and there is an indication of termination which can be done by triggering the function or setting up a flag to 0.

4. Summary

A hybrid algorithm for automated clustering has been introduced in the present paper. The proposed algorithm is a blend of concepts of k-means algorithm of clustering and evolutionary programming. The key idea behind this algorithm is the process of evolution which is continuous which can help us further to automate the clustering system. In evolution, the fittest is always selected to get the chance of survival by natural selection. By keeping this idea in focus, the proposed algorithm has been prepared. Hence it can be concluded that the algorithm is of great use in automated clustering systems and in predicting clustering trends.

References

[1] Darwin, C. (1859). *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* John Murray, London; modern reprint Charles Darwin. Julian Huxley (2003). *On The Origin of Species*. Signet Classics. ISBN 0-451-52906-5. The complete work of Charles Darwin. Retrieved from http://darwin-online.org.uk/content/frameset?itemID=F373&viewtype=side&pageseq=2.

[2] Fogel, L. J., Owens, A. J. & Walsh, M. J. (1966). *Artificial Intelligence through Simulated Evolution*. John Wiley.

[3] MacQueen, J. (1967). *Some Methods for Classification and Analysis of Multivariate Observations*. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1, 281-297. Berkeley, California: University of California Press.

[4] K-means Clustering. Retrieved from http://en.wikipedia.org/wiki/K-means_clustering

[5] Evolutionary Programming. Retrieved from http://en.wikipedia.org/wiki/Evolutionary_programming

[6] Evolutionary Algorithm. Retrieved from http://en.wikipedia.org/wiki/Evolutionary_algorithm