Volume 2, No. 1, Jan-Feb 2011

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Clustering Gene Expression for Colon and Leukemia Dataset Using Affinity Propagation

D. Napoleon*
Assistant Professor
Department of Computer Science
Bharathiar University
Coimbatore, Tamil Nadu, INDIA
mekaranapoleon@yahoo.co.in

G.Baskar
Research Scholar
Department of Computer Science
Bharathiar University
Coimbatore, Tamil Nadu, INDIA
baskarb@yahoo.com

Abstract: The most prominent and widely used clustering algorithm is Lloyd's algorithm sometimes also referred to as the k-means algorithm. The k-means algorithm is one of the most widely used methods to partition a Dataset into groups of patterns. The main strength of the algorithm is that it can quickly determine Clustering's of the same point set for many values of k. However, the k-means method converges to one of many local minima. And it is known that, the final result depends on the initial starting points. We introduce an global k-means, x-means and affinity propagation. our experimental results show that, good initial starting points lead to improved solution

Keyword: Data mining, Global k-means, x-means, Affinity propagation.

I. INTRODUCTION

Clustering has many applications in different areas of computer sciences such as computational biology, machine learning, data mining and pattern recognition. Since the quality of a partition is rather problem dependent, there is no general clustering algorithm. Consequently, over the years many different clustering algorithms have been developed. These algorithms can be characterized as hierarchical algorithms or partitioning algorithms. Clustering in gene expression data sets is a challenging problem. Different algorithms for clustering of genes have been proposed. [11] However due to the large number of genes only a few algorithms can be applied for the clustering of samples. k-means algorithm and its different variations are among those algorithms. (Al-Sultan 1995, Brown & Entail 1992, de Merle et al. 2001, Diehr 1985, Dubes & Jain 1976, Hanjoul & Peeters 1985, Hansen & Jaumard 1997, Hansen & Mladenovic 2001a, Hansen & Mladenovic 2001b, Koontz et al. 1975, Selim & Al-Sultan 1991, Spath 1980, Sun et al. 1994)). Since the number of genes in gene expression data sets are very large most of these algorithms cannot be applied for clustering of samples in such data sets. A is computed and in
order to compute k-partition at the k-th iteration this algorithm uses centers of k-1 clusters from the previous iteration.

Affinity propagation takes a different approach to clustering. Rather than make hard decisions on the cluster centers at each iteration, soft information about cluster exemplars is propagated through the dataset by way of a message passing algorithm. Affinity Propagation performs the max-sum algorithm on a factor graph model of the data to solve for a good configuration of cluster members.

A. *The Proposed Algorithm*

In this paper we describe our algorithm. That produces good starting points for the k-means algorithm instead of selecting them randomly. And this will leads to better clusters at the final result than that of the original k-means.

II. GLOBAL K-MEANS

Introduced by A. Likas, N. Vlasis and J.J. Verbeekin the paper entitled "The Global *k*-means clustering algorithm" in 2003, the concept of clustering with Global *k*-means is partitioning the given dataset into *M* clusters so that a clustering criterion is optimized. The common clustering criterion is the sum of squared Euclidean distances between each data point and the cluster centroid.

$$E(M_1,,,M_K)=\sum_{i=1}^{N}{}_K\sum_{m=1}^{M}{}_K \left\| X_i\text{-}u_m \right\|^2$$

Global *k*-means deploys the k-means algorithm to Find locally optimal solutions by trying to keep the Clustering error to a minimum. The *k*-means algorithm starts by placing the cluster center arbitrarily and at each step moves the cluster center with the aim to minimize the clustering error. The down side to this algorithm is that it is sensitive to the initial position of the cluster centers. To overcome this, *k*-means can be scheduled to run several times and each time with a different starting point. The gist of Global *k*-means is that instead of trying to find all cluster centers at once, it proceeds in an incremental fashion. Incremental in the sense that one cluster center is found at a time. Assume a *K*-clustering problem is to be solved; the algorithm starts by solving for a 1-clustering problem and the placement of the cluster center in this instance would equal the centroid of the given dataset. The next step would be to add another cluster center at its optimal position, given, the first cluster center has already been found. To do this, *N*-executions of *k*-means algorithm will be executed with the initial positions of the cluster centers being the first cluster which was found when solving for a 1-clustering problem and the second cluster's starting position will be at *n x* where.1 *N n*̃ The final answer

for a 2-clustering problem will be the best solution from the *N*-executions of *k*-means algorithm. Let (c1(k),…,ck(k)) denote the final solution for the *k*-clustering problem. We will solve it iteratively which means solving a 1-clustering problem, then a 2-clustering problem, until a (*k*-1)-clustering problem and the solution of *k*-clustering problem can be solved by performing *N*-executions of *k*-means algorithm with starting positions of (c1(k-1),…,c(k-1)(k-1),Xn). A simple pseudo code of it will be Problem: to solve *k*-clustering problem for dataset, *X*

For *i*=1 to *k*
{
If *i* = 1 then
Ci= centroid of dataset, *X*
Else
For j=1 to N
Run *k*-means with initial values of
{ *j i i X c c* , ,..., 1 ~ }
}

With the final solution, (c1(k),…,ck(k)), Global *k*-means has actually found solutions of all *k*-cluster problem where *k*=1,…,*K* without needing any further computations. This assumption seems very natural: we expect that the solution of a *k*-clustering problem to be reachable (through local search) from the solution of a (*k*-1)-clustering problem, once the additional center is placed at an appropriate position within the data set. Alas, the downside is that the computational time of Global *k*-means can be rather long.[3]

III. X-MEANS ALGORITHM

X-means algorithm (Dan Pelleg and Andre Moore, 2000) searches the space of cluster locations and number of clusters efficiently to optimize the Bayesian Information Criterion (BIC) or The Akaike Information Criterion (AIC) measure. The kd-tree technique is used to improve the speed for the algorithm. In this algorithm, numbers of clusters are computed dynamically using lower and upper bound Supplied by the user. The algorithm consists of mainly two steps which are repeated until completion. algorithm will be executed with the initial positions of the cluster center.

Steps*:*

Step1 **:**(Improve-Params) In this step, we apply k-means algorithm initially for k clusters till convergence. Where k is equal to lower bound supplied by the user.

Step2 :(Improve -Structure) this structure improvement step begins by splitting the each cluster center into two children in opposite directions along a randomly chosen vector. After that we run k-means locally within each cluster for two clusters. The decision between the children of each center and itself is done comparing the BIC-values of the two structures.

Step 3**:** if k> =kmax (upper bound) stop and report to best scoring model found during search otherwise go to step 1.

IV. AFFINITY PROPAGATION

The process of Affinity Propagation can be viewed as a message communication process on a factor graph (Kschischang *et al.*, 2001). There are two kinds of messages exchanged between data points, i.e., 'responsibility' and 'availability'. The responsibility *r*(*i*, *k*), sent from data point *i* to candidate exemplar point *k*, reflects the accumulated evidence for how well-suited point *k* is to serve as the exemplar for point *i*, taking into account other potential exemplars for point *i*. The availability *a*(*i*, *k*), sent from candidate exemplar point *k* to point *i*, reflects the accumulated evidence For how appropriate it would be for point *i* to choose point *k* as its exemplar, taking into account the support from other points that point *k* should be an exemplar. The messages need only be exchanged between pairs of points with known similarities

Steps:

Step1**:** Initialization the availability a (i.k) to zero

$$a(i, k)=0 \quad (1)$$

step2**:** update the responsibility using rule

$$r(i,k) \leftarrow s(i,k) - \max_{k' \text{ s.t. } k' \neq k} \{a(i, k'). s(i, k')\}. \quad (2)$$

step3**:** update the availability using the rule

$$a(i,k) \leftarrow \min\{0, r(k,k) + \sum_{i' \text{ s.t. } i' \neq i,k} \max\{0, r(i',k)\}\} \quad (3)$$

The self-availability is updated differently

$$a(k,k) \leftarrow \sum_{i' \text{ s.t. } i' \neq k} \max\{0, r(i',k)\}. \quad (4)$$

Step 4**:** The message-passing procedure may be terminated after a fixed number of iterations, after changes in the messages fall below a threshold or after the local decisions stay constant for some number of iterations.[6]

Input:
s(*i*, *k*): the similarity of point *i* to point *k*.
p(*j*): the preferences array which indicates the preference that data point *j* is chosen as a cluster center.
Output:
idx(*j*): the index of the cluster center for data point *j*.
dpsim: the sum of the similarities of the data points to their cluster centers.
netsim: the net similarity (sum of the data point similarities and preferences).
expref: the sum of the preferences of the identified cluster centers
netsim: the net similarity (sum of the data point similarities and preference)

Availabilities and responsibilities can be combined to make the exemplar decisions. For point i, the value of *k* that maximizes a(i, k)+r(i, k) either identifies point *i* as an exemplar if *k*=*i* or identifies the data point that is the exemplar for point *i*. When updating the messages, numerical Oscillations must be taken into consideration. As a result, each message is set to λ times its value from the previous iteration plus 1–λ times its prescribed updated value.

The λ should be larger than or equal to 0.5 and less than 1. If λ is very large, numerical oscillation may be avoided, but this is not guaranteed. Hence a maximal number of iterations are set to avoid.

V. DATASET

We used two different cancer datasets to make a study of various k-mean based algorithms. The Leukemia data set is acollection of gene expressionmeasurements from 72 leukemia (composed of 62 bone marrow and 10 peripheral blood) samplesreported by Golub. It contains an initial training set composed of 47 samples of acute lymphoblastic

leukemia (ALL) and 25 samples of acute myeloblastic leukemia (AML). Here we take two variants of leukemia dataset one with 50-genes and another one with 3859-genes.

The Colon dataset is a collection of gene expression measurements from 62 Colon biopsy samples reported by Alon. It contains 22 normal and 40 Colon cancer samples. The Colon dataset consists of 2000 genes.

Table I: The top 5 genes in each of the 2 clusters found in the colon-cancer dataset

Cluster	Rank	Accession Number	Name
1	1		PUTATIVE ATP-DEPENDENT RNA HELICASE C06E1.10 IN CHROMOSOME
1	2	H05814	III (Caenorhabditis elegans)
1	3		
1	4	U33429	Human mRNA for (2'-5') oligo A synthetase E (1,6 kb RNA)
1	5		
		X02874	human K+ channel beta 2 subunit mRNA, complete cds
		H22579	INTEGRIN ALPHA-6 PRECURSOR (Homo sapiens)
		H25940	PUTATIVE SERINE/THREONINE-PROTEIN KINASE PSK-H1 (Homo sapiens)
2	1	T73092	EUKARYOTIC INITIATION FACTOR 4A-I (Homo sapiens)
2	2		
2	3	R26146	NUCLEAR FACTOR NF-KAPPA-B P105 SUBUNIT (HUMAN)
2	4		
2	5	T90851	ADP-RIBOSYLATION FACTOR-LIKE PROTEIN 4 (Rattus norvegicus)
		R9337	HOMEOTIC GENE REGULATOR (Drosophila melanogaster)
		T69446	EUKARYOTIC INITIATION FACTOR 4A-I (HUMAN)

VI. CONCLUSION AND FUTURE WORK

The k-means use in this study is global k-means,x-means and affinity propagation, the average accuracy of these is show below in table. Analysis colon dataset and leukemia dataset with the clustering algorithm the average accuracy of affinity propagation is better in leukemia dataset and. The convergence rate is also higher and speed of execution time is good however the variations of k-means required more trails to reach at a stable and good clustering solution. Performance of this algorithm can be improved with the help of other clustering algorithm and **fuzzy** logic to get better quality of cluster. So these algorithm help to get good result

Result Over Clustering Algorithm Using 2000 Gene Colon Dataset (Total Number of Records Present In Data Set =62)

Table 2

Clustering Algorithm	Correctly Classified	Average accuracy
Global K-means	65	91.67
x-means	64	88.89
Affinity Propagation	66	92.12

Result over different variation of k-means algorithm using 3859-gene leukemia(total number of record present in dataset=72)

Table 3

Clustering Algorithm	Correctly Classified	Average accuracy
Global K-means	37	59.68
x-means	37	59.68
Affinity Propagation	38	60.54

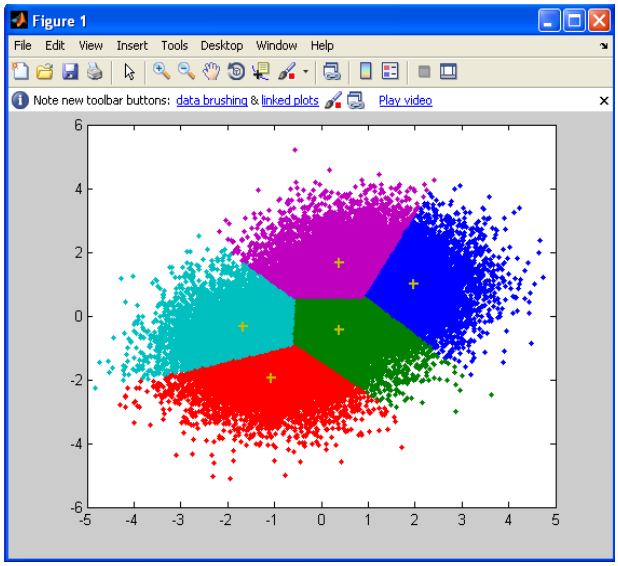
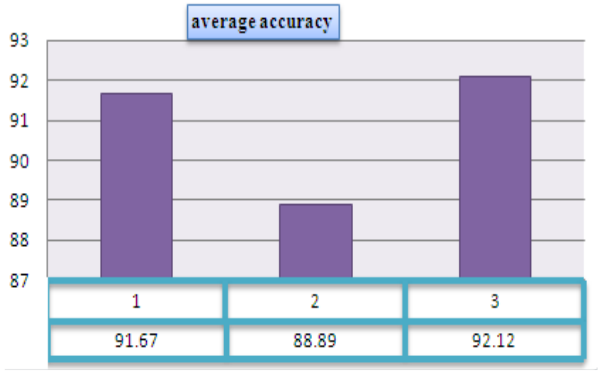


Figure 1.Cluster Formation



GRAPH 1

VII. REFERENCES

[1] Bradley P. S., Fayyad U. M. "Refining Initial Points for K-Means Clustering", Proc. of the15th International Conference on Machine Learning (ICML98), J. Shavlik (ed.), MorganKaufmann, San Francisco, 1998, pp.

[2] Fahim A. M., Salem A. M., Torkey F. A. and Ramadan M. "An efficient enhanced k-means clustering algorithm". Journal of Zhejiang University Science A, 2006, vol 7(10),

[3] Hung M., WU J., Chang J. and Yang D.,"An Efficient k-Means Clustering Algorithm UsingSimple Partitioning", journal of Information Science and Engineering, 2005, vol. 21,

[4] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," Data Mining and Knowledge Discovery, 1998,

[5] Frey, B.J., Dueck, D., 2006. Mixture Modelingby Affinity Propagation. Neural Information Processing neural information processing system

[6] Brendan j.Frey and Delbert Duec clustering passing message between data point science, 315(5814):972{976}

[7] G.W. Milligan, M.C. Cooper, An examination of procedures for determining the number of clusters in a data set, Psychometrika 50 (1985) 159–179.

[8] Anjan Goswami. Department of Computer Science and Engineering" Fast and Exact Out of-Core and Distributed K-Means Clustering 2001

[9] Bagirov, A.M. [Adil M.], Modified global k-means algorithm for minimum sum-of-squares clustering problems, October 2008

[10] E. Papageorgiou, I. Kotsioni, A. Linos, "Data Mining: A New Technique in Medical Research", Hormones 2005, 4(4):189-191

[11] Jaiwei Han, Michelinne Kamber, "Data Mining : Concepts and Techniques ", 2001, II Edition

[12] Jason Shasha (EDS), "Data mining in bioinformatics" Pg. no: 654,

[13] MacQueen, J.B. (1967). Some Methods for Classification and Analysis of Multivariate Observations. In Proc. of 5th Berkley Symposium on Mathematical Statistics and Probability, Volume I: Statistics, pp. 281–297.

[14] Pawlak. Z. Rough Sets International Journal of Computer and Information Sciences, (1982), 341-356.

[15] P awan Lingras, Chad West. Interval set Clustering of Web users with Rough k-Means, submitted to the Journal of Intelligent Information System in 2002.

[16] Yeung K.Y, Haynor D.R, Ruzzo W.L. Validating clustering for gene expression data. Bioinformatics. 2001.

[17] Zhang Y. , Mao J. and Xiong Z.: An efficient Clustering algorithm, In Proceedings of Second International Conference on Machine Learning and Cyber netics, November 2003.

[18] L. Kaufman and P. J. Rousseeuw, Finding Groups in Data: an Introduction to Cluster Analysis, John Wiley & Sons, 1990.

[19] Springer T.L. Wang, Mohammed J. Zaki, Hannu T.T Toivonen and Dennis International Edition tumours: a gene expression study. Lancet2002.

AUTHOR PROFILE

D. Napoleon received the Master's Degree in Computer Applications from Madurai Kamaraj University, Tamil Nadu, India in 2002, and the M.Phil degree in Computer Science from Periyar University, Salem, Tamil Nadu, India in 2007. He has published articles in National and International Journals. He has presented papers both in National and International Conferences. His Current research interest includes: Knowledge discovery in Data Mining and Computer Networks.

G.Baskar received his Master's degree in Information Technology in K.S.Rangasamy College of Technology, Tiruchengode, Tamil Nadu India in 2008 and M.Phil Degree in Computer Science from Bharathiar University, Coimbatore, Tamil Nadu, India in 2010. His area of interest includes Data Mining**.**