Two Phase Integrated Rule based Model (TPC-IRBM) for Clustering of Gene Expression Data of CA1 Region of Rat Hippocampus

Sudhakar Tripathi
Department of Computer Science & Engineering
Indian Institute of Technology(BHU)
Varanasi,India,221005

R.B.Mishra
Department of Computer Science & Engineering
Indian Institute of Technology(BHU)
Varanasi,India,221005

ABSTRACT
This paper propose a semi supervised clustering model TPC-IRBM(Two phase clustering-Integrated rule based model) for clustering large data set such as gene expression data. TPC-IRBM works in two phases to cluster the gene expression data set. The proposed model is based on rule based models CRT,C5,CHAID and QUEST. In the first phase of the model 30 % data(which may vary) is extracted to prepare training, testing and validation data (TTV data)using suitable heuristic or neural network based clustering techniques. The output of first phase is used as build the models and generate the rule base fitting to TTV data using aforesaid models. The proposed model is then constructed by selecting and integrating the quality rules of various models using qualifying criteria corresponding to every cluster.The number of quality rules in proposed model is much more compared to that of CRT,C5,CHAID and QUEST.The performance in terms of accuracy is better compared to the models. Although in some cases Neural Network based models performance is slightly better but a very high cost of complexity for very large data set.

Keywords
Gene expression clustering,semi supervised clustering, integrated rule based model, two phase clustering,CA1 region gene expression clustering of rat hippocampus.

1. INTRODUCTION
Clustering is very important technique in many data analysis applications. One of the most important applications is in bioinformatics. In bioinformatics clustering is mostly used to group genes whose expression patterns are available corresponding to various experimental(samples) or at various time points(time series). The most challenging task in clustering is to find actual number of clusters corresponding to given patterns of a dataset. Another is similarity measure; third one is how data is arranged. In bioinformatics clustering is very much useful because the genes that are co-expressed (belongs to same clusters) tend to have similar gene functions, premotor regulatory sequences and protein complexes. Clustering is the far most used method in gene expression analysis.

In some exceptional cases genes with nearly similar expressions may have different control strategies. After clustering a gene expression dataset into appropriate number of clusters we need not to evaluate individual gene functions, premotor sequences and protein complexes. Instead we can evaluate functions, premotor sequences and protein complexes of very few genes of various clusters and so genes belonging to same cluster will have mostly same functions, premotor sequences and protein complexes. The main types of microarray data analysis include gene selection, clustering, and classification[19]. Piatetsky-Shapiro and Tamayo present one great challenge that data mining practitioners have to deal with. Microarray datasets -in contrast with other application domains- contain a small number of records (less than a hundred), while the number of fields (genes), is typically in thousands[20].

Large amount of gene expression data have been generated, but there is great requirement of developing the methods to analyze and explore the genes and related information [9].Clustering is much useful technique for the analysis of gene expression data. Many clustering algorithms are there which are being used to analyze gene expression data, including heuristic based like hierarchal[10],k-means[11], self organizing maps[12], graph theoretic approaches[13][14] and support vector machine[15] , Model based clustering[16],Bayesian model based clustering([17].

Many clustering techniques have been used to cluster gene expression dataset. But no one is reported to be widely efficient and acceptable in all cases due to various reasons. Because gene expression datasets are very large and expressed over very large samples or time series. It is more challenging to cluster gene expression datasets. So it is very difficult and time consuming to analyze genes corresponding to all experiments and time series expressions.

As not all the sample experiments and time expressions play role to corresponding cluster identification. We need such clustering techniques which can explore the expression regarding the importance of experiments (or time series). So that gene data sets can be clustered using only important feature expressions. Clustering in this way reduces overall effort and cost to cluster the genes for predictions of their functionalities.

Here we have used two phase clustering (TPC) technique to cluster gene expression data set. We have assumed that data points have been generated from same or similar data sources. In first phase we have implemented techniques to predict correct number of clusters and generate training, testing, and validation data sets from data set which have to be clustered using two-step, kohonens, k-means algorithms. In second phase we have proposed to train various decisions tree (Rule-based) models for generating rule sets and then use the optimal rule sets generated from these models combining them all to cluster the rest of the data sets or data sets generated from same source(or distribution).

Rest of the paper have been organized as follows:
Section2: data sets, tools (implementation)
Section3: methodology
Section4: cluster assessment
Section 5: results and discussions
Section6: conclusions: summary, conclusion, future work, acknowledgement.

In our method we have used heuristic based algorithms(two-step, kohonens, k-means) for initial phase and on smaller data set. We have used the output of first phase to generate rule base which will be used to cluster rest of the data set originating from same or similar source.

As we have used rule based clustering technique for clustering the gene expression data sets. Rule based clustering techniques are easy to interpret and efficient. Once after second phase clustering model is generated it is simple and very fast to cluster the gene expression data sets with high efficiency and accuracy.

2. DATA SET & IMPLEMENTATION

we have used subset of gene expression data of Rattus norvegicus (Rat) of hippocampal region for CA1 region of Gene expression profiling in differential cognitive outcomes in aging: CA1 from GEO database with geo_accession id GSE14723[18].

The subset of CA1 expression data we used contains 3491 instances over 23 RNA samples from CA1 region of the hippocampi of young animals, aged animals with unimpaired spatial learning, and aged animals with impaired spatial learning. In the dataset there are 6 samples of aged (24-26 months old) male unimpaired (AU), 8 samples of aged (24-26 months old) male impaired (AI) and 9 samples of young (6 months old) male. Features, Sample GEO id and Feature names are shown in Table 1.

Table 1. Feature Variables of dataset.

Feature Variable name	Sample GEO id	Sample_title/ Feature Name
SX1	GSM367827	Aged Impaired CA1, biological
SX2	GSM367828	Aged Impaired CA1, biological
SX3	GSM367829	Aged Impaired CA1, biological
SX4	GSM367830	Aged Impaired CA1, biological
SX5	GSM367831	Aged Impaired CA1, biological
SX6	GSM367832	Aged Impaired CA1, biological
SX7	GSM367833	Aged Impaired CA1, biological
SX8	GSM367834	Aged Impaired CA1, biological
SX9	GSM367835	Aged Unimpaired CA1,
SX10	GSM367836	Aged Unimpaired CA1,
SX11	GSM367838	Aged Unimpaired CA1,
SX12	GSM367839	Aged Unimpaired CA1,
SX13	GSM367840	Young CA1, biological rep9
SX14	GSM367841	Aged Unimpaired CA1,
SX15	GSM367842	Aged Unimpaired CA1,
SX16	GSM367843	Young CA1, biological rep1
SX17	GSM367844	Young CA1, biological rep2
SX18	GSM367845	Young CA1, biological rep3
SX19	GSM367846	Young CA1, biological rep4
SX20	GSM367847	Young CA1, biological rep5
SX21	GSM367848	Young CA1, biological rep6
SX22	GSM367849	Young CA1, biological rep7
SX23	GSM367850	Young CA1, biological rep8

We implemented all the methods (CRT,C5,CHAID,QUEST)using SPSS Clementine 11.1 computing environment. Microsoft Excel have been used for all data storage and manipulation. C&RT(CRT) stands for Classification and Regression Trees, originally described in the book by the same name [1]. C5 (improvement of C4.5) is an algorithm used to generate a decision tree developed by Ross Quinlan[3][4][2].CHAID stands for Chi-squared Automatic Interaction Detector. It is a highly efficient statistical technique for segmentation, or tree growing, developed by Gordon V. Kass[5].QUEST stands for Quick, Unbiased, Efficient Statistical Tree. It is a relatively new binary tree-growing algorithm [6].

3. METHODOLOGY

Two phase Clustering described in this research, works in two phases. In first phase actual number of clusters is predicted and then using any heuristic based algorithm which is efficient enough(we have used K-means) we can cluster the data set for training, testing,and validation. In second phase using training testing and validation data set we generate the models using CART,C5,Chaid & Quest techniques.After the models have been generated rule sets are generated corresponding to all models for clustering.Then we combine all the rule sets to generate a rule base. In rule base we select only those rules whose confidence factor are very high(greater than rcf). Then finally the clustering is performed using this rule based model. The rule based clustering is more efficient and of high quality compared to any of heuristic algorithm, decision tree or other rule based classifiers. The detailed description of the TPC-IRBM is as follows:

3.1 TPC Phase1: (See Figure1)

Step1: **data preparation:** 30% data have been selected from the original data set for predicting the actual no of cluster as well as preparing the training, testing and validation data set for phase2.

Step2: if actual number of clusters are known then this step is skipped else we have used two-step clustering(hierarchal) technique to find lower bound of number of clusters.Kohonens self-organizing map have been used to find upper bound of cluster number. Then we have used K-means(optionally any other efficient clustering method can be used) clustering technique for clustering the TTV data set from k= C_{lb} to C_{ub}.Here we have used proximity measure, mean , standard deviation to predict actual number of clusters.as proximity measure of any pairs of clusters gets less than standard deviation, we stop and that is the predicted actual number of clusters.

On our data set applying this we got actual number of clusters predicted to be K_{opt}=6. Depending upon cases any one can analyze the outcome with other measures and then predict the actual number of clusters.

Step3: After predicting actual number of clusters i.e. K_{opt}, cluster the TTV data set using K-means for K_{opt}=6(optionally any other suitable clustering technique can be used). After clustering the dataset it is partitioned in training, testing and validation data set(T,T,V) to be used in Phase2.

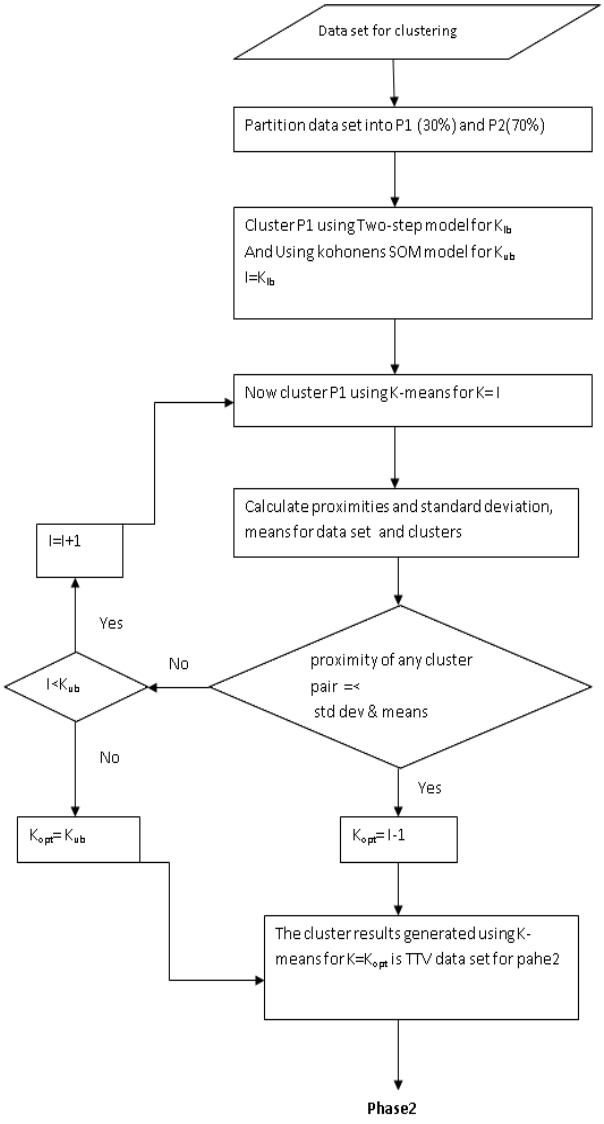

Fig1.TPC-IRBM Phase1

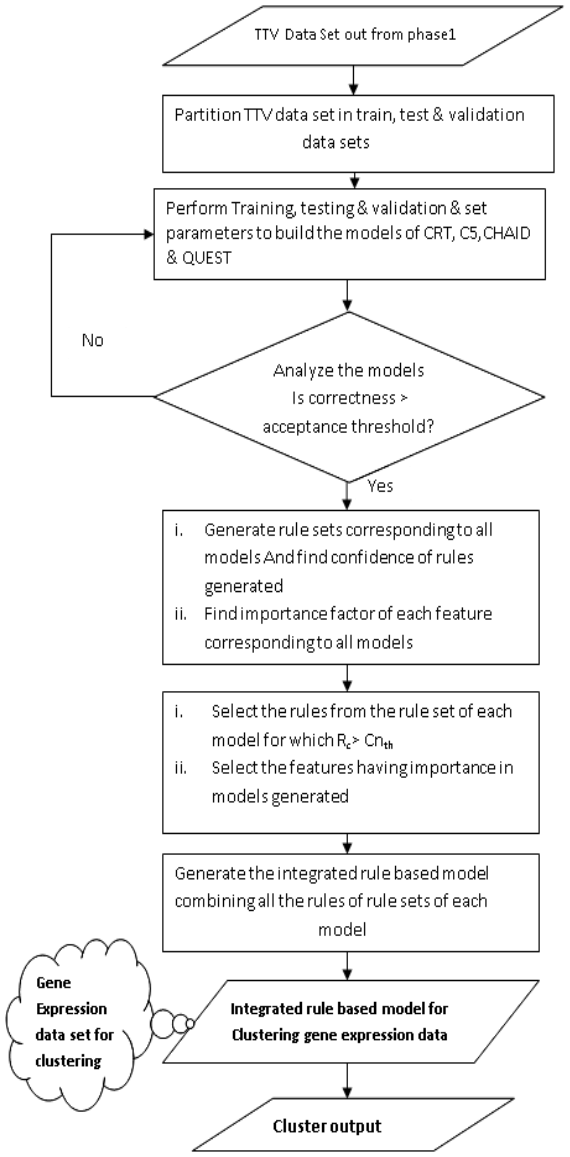

Fig2.TPC-IRBM Phase2

3.2 TPC Phase2: In phase 2(See Figure2) use T,T,V, data set outcome of phase1 to train, test and validate the the CRT,C5,Chaid,Quest models to generate the rule sets. From the rule sets generated from these models find out the importance factor of various features of patterns(experiments. Set threshold for importance factor that is acceptable to T_I. Now use only those features(samples or experiments) whose Importance factor >= T_I. Now find out those rules whose confidence is greater than R_{CI}(confidence threshold). Now combine all the rules whose confidence >= R_{CI}. Generated from all models to produce a rule base to be used for clustering data set. Now this Rule bsed model will be validated with data set from GC1 to GC10. Then we will generalize the model for clustering.

Benefits of this approach are-
Reduced experimentation required

(i) Improved rule base
(ii) Data reduction
(iii) Improved efficiency in terms of speed and storage.
(iv) Although ANN models may give slightly better results but at much more cost of speed and storage. Experiment (samples) features are more to that compared with rule based model of TPC.

4. RESULT & DISCUSSION

4.1 Phase 1 Result:

In this section, we show the results of each phase and steps involved in TPC. On TTV dataset Two Step Clustering and Kohonen's Self Organising Map have been applied which gives lower and upper bound of number of clusters and for the TTV dataset it comes out to be klb=2 and kub=12.Then in next step we applied K-means clustering on TTV dataset for k=2 t0 k=12 and stopping criteria is considered to be for k=Kopt where proximity between clusters is more than

std.dev. . For our TTV dataset stopping criteria is met for K=6 which is the value of Kopt=6.Cluster proximities for k=2 to k=6 are shown in table2 and std dev. are shown in table3. For k=2 to k=6 cluster proximities are more or nearly equal than that of std.dev. but for k>6 it is overlapping or less than std. dev..So the Kopt=6 is taken as actual number of clusters.

Table 2. Cluster proximities for k=2 to k=6.

Value of K	Proximity of clusters						
K=2		C1	C2				
	C1	0	2.057				
	C2	2.057	0				
K=3		C1	C2	C3			
	C1	0	1.566	1.036			
	C2	1.566	0	2.602			
	C3	1.036	2.602	0			
K=4		C1	C2	C3	C4		
	C1	0	2.116	0.840	0.874		
	C2	2.116	0	2.957	1.242		
	C3	0.840	2.957	0	1.715		
	C4	0.874	1.242	1.715	0		
K=5		C1	C2	C3	C4	C5	
	C1	0	2.336	0.751	0.656	1.472	
	C2	2.336	0	3.087	1.679	0.863	
	C3	0.751	3.087	0	1.408	2.223	
	C4	0.656	1.679	1.408	0	0.815	
	C5	1.472	0.863	2.223	0.815	0	
K=6		C1	C2	C3	C4	C5	C6
	C1	0	2.509	0.688	1.119	1.880	0.539
	C2	2.509	0	3.197	1.389	0.628	1.969
	C3	0.688	3.197	0	1.807	2.568	1.227
	C4	1.119	1.389	1.807	0	0.761	0.579
	C5	1.880	0.628	2.568	0.761	0	1.341
	C6	0.539	1.969	1.227	0.579	1.341	0

Table 3. Std deviation & means: for k=2 to k=6.

K-Means	Cluster No.	Total No. of Records	*Mean*	*Std. Dev*
k=2	Cluster-1	1952	9.150478	1.728522
	Cluster-2	1539	2.998739	1.201652
k=3	Cluster-1	1091	7.367261	1.072087
	Cluster-2	1372	2.681913	0.820043
	Cluster-3	1028	10.46652	1.192174
k=4	Cluster-1	1044	8.869348	0.750565
	Cluster-2	1284	2.543652	0.642696
	Cluster-3	522	11.38309	0.97687
	Cluster-4	641	6.254087	0.944
k=5	Cluster-1	818	9.414957	0.594913
	Cluster-2	1187	2.425826	0.505261
	Cluster-3	414	11.66196	0.902739
	Cluster-4	696	7.449304	0.658522
	Cluster-5	376	5.007783	0.818739
k=6	Cluster-1	649	9.858391	0.528522
	Cluster-2	1107	2.348957	0.426348
	Cluster-3	326	11.91861	0.846696
	Cluster-4	448	6.516087	0.618304
	Cluster-5	312	4.239913	0.668
	Cluster-6	649	8.244739	0.509522

After the estimation of Kopt=6 K-means algorithm is applied on TTV dataset and output cluster tags of instances are considered to be the classification tags for building predictive models of C%,CART,CHAID and QUEST. The output of K-means algorithm is output of phase1 that is dataset having 3491 instances tagged with cluster number. The number of instances ,mean and std.dev. are shown in table3.

4.2 Phase2 Result:

For phase two the TTV dataset is partitioned in three sets i.e. training dataset (2384 instances), testing dataset (758 instances) and validation dataset(349 instances).

The training data set is used to build the predictive models of C5,CART, CHAID and Quest. We have taken three metrics for analysis of the models i.e. features selected and their importance factor in model buiding, rules generated by the models with their confidence factor and predictive accuracy for training testing and validation datasets.

The features selected and their relative importance factor used in building each model are different. C5 model used 17 features, CART 10 features, CHAID 4 features and Quest 3 features out of total 23 features in training dataset. The most important feature in C5,CART, CHAID and QUEST are SX19,SX9,Sx9 and SX12 respectively and the least important are SX10,SX10,SX7,S15 respectively. Considering all models feature selection and relative importance only 17 features are selected at most out of which 7 features are having very less importance compared to others which can be ignored having negligible effect on results but we have taken all 17 features. Thus the large amount of data is reduced to considerable level which improves the efficiency in terms of time and space complexity as in bioinformatics a huge amount of data is available to analyze space and time complexity is major factor.

The rules generated by the models on training dataset are shown in table4 and their compositions are shown in table 5. The rules for each cluster generated by the models are shown along with number of instances and confidence factor of the rule. Higher confidence factor of the rule indicates lower misclassification and vice-versa. Total numbers of rules corresponding to all clusters generated by C5 are 32, by CART are 16, by CHAID are 11 and by QUEST are 7.Rule composition is done by logical OR operator.

Table4. Rule set generated by models after training

C5 Rule Set		Rule (R) (No. of instances; Confidence factor)
	Rules for Cluster 1 $\mathbf{R_{C5}1}$	Rule 1 (5; 1.0) $\mathbf{R_{C5}11}$
		Rule 2 (9; 1.0) $\mathbf{R_{C5}12}$
		Rule 3 (4; 1.0) $\mathbf{R_{C5}13}$
		Rule 4 (407; 1.0) $\mathbf{R_{C5}14}$
		Rule 5 (5; 1.0) $\mathbf{R_{C5}15}$
		Rule 6 (5; 1.0) $\mathbf{R_{C5}16}$
	Rules for Cluster 2 $\mathbf{R_{C5}2}$	Rule 1 (764; 1.0) $\mathbf{R_{C5}21}$
		Rule 2 (8; 1.0) $\mathbf{R_{C5}22}$
		Rule 3 (2; 1.0) $\mathbf{R_{C5}23}$
	Rules for Cluster 3 $\mathbf{R_{C5}3}$	Rule 1 (3; 1.0) $\mathbf{R_{C5}31}$
		Rule 2 (208; 1.0) $\mathbf{R_{C5}32}$
	Rules for Cluster	Rule 1 (3; 1.0) $\mathbf{R_{C5}41}$
		Rule 2 (2; 1.0) $\mathbf{R_{C5}42}$

R_{C5}	4 R_{C5}**4**	Rule 3 (279; 1.0) R_{C5}**43**
		Rule 4 (3; 1.0) R_{C5}**44**
		Rule 5 (6; 1.0) R_{C5}**45**
		Rule 6 (8; 1.0) R_{C5}**46**
		Rule 7 (2; 1.0) R_{C5}**47**
	Rules for Cluster 5 R_{C5}**5**	Rule 1 (5; 1.0) R_{C5}**51**
		Rule 2 (2; 1.0) R_{C5}**52**
		Rule 3 (213; 1.0) R_{C5}**53**
		Rule 4 (1; 1.0) R_{C5}**54**
		Rule 5 (3; 1.0) R_{C5}**55**
	Rules for Cluster 6 R_{C5}**6**	Rule 1 (2; 1.0) R_{C5}**61**
		Rule 2 (1; 1.0) R_{C5}**62**
		Rule 3 (6; 1.0) R_{C5}**63**
		Rule 4 (3; 1.0) R_{C5}**64**
		Rule 5 (5; 1.0) R_{C5}**65**
		Rule 6 (398; 0.997) R_{C5}**66**
		Rule 7 (18; 1.0) R_{C5}**67**
		Rule 8 (1; 1.0) R_{C5}**68**
		Rule 9 (3; 1.0) R_{C5}**69**
	Default: Cluster	Default
CRT Rule Set		**Rule** (**No. of instances; Confidence factor)**
R_{CRT}	Rules for Cluster 1 R_{CRT}**1**	Rule 1 (2; 1.0) R_{CRT}**11**
		Rule 2 (15; 0.6) R_{CRT}**12**
		Rule 3 (418; 0.995) R_{CRT}**13**
		Rule 4 (5; 1.0) R_{CRT}**14**
	Rules for Cluster 2 R_{CRT}**2**	Rule 1 (771; 1.0) R_{CRT}**21**
		Rule 2 (2; 1.0) R_{CRT}**22**
	Rules for Cluster 3 R_{CRT}**3**	Rule 1 (4; 0.75) R_{CRT}**31**
		Rule 2 (208; 1.0) R_{CRT}**32**
	Rules for Cluster 4 R_{CRT}**4**	Rule 1 (295; 0.976) R_{CRT}**41**
		Rule 2 (6; 1.0) R_{CRT}**42**
		Rule 3 (4; 1.0) R_{CRT}**43**
	Rules for Cluster 5 R_{CRT}**5**	Rule 1 (6; 0.833) R_{CRT}**51**
		Rule 2 (217; 0.991) R_{CRT}**52**
	Rules for Cluster 6 R_{CRT}**6**	Rule 1 (9; 0.667) R_{CRT}**61**
		Rule 2 (414; 1.0) R_{CRT}**62**
		Rule 3 (8; 0.625) R_{CRT}**63**
	Default: Cluster	
CHAID Rule Set		**Rule** **(No. of instances; Confidence factor)**
R_{CHAID}	Rules for Cluster 1 R_{CHAID}**1**	Rule 1 (189; 0.746) R_{CHAID}**11**
		Rule 2 (26; 1.0) R_{CHAID}**12**
		Rule 3 (238; 1.0) R_{CHAID}**13**
	Rules for Cluster	Rule 1 (715; 1.0) R_{CHAID}**21**
	Rules for Cluster	Rule 1 (238; 0.887) R_{CHAID}**31**
	Rules for Cluster 4 R_{CHAID}**4**	Rule 1 (239; 0.812) R_{CHAID}**41**
		Rule 2 (215; 0.507) R_{CHAID}**42**
	Rules for Cluster	Rule 1 (238; 0.752) R_{CHAID}**51**
	Rules for Cluster 6 R_{CHAID}**6**	Rule 1 (24; 1.0) R_{CHAID}**61**
		Rule 2 (238; 1.0) R_{CHAID}**62**
		Rule 3 (24; 0.833) R_{CHAID}**63**
	Default: Cluster	

QUEST Rule Set		**Rule** (No. of instances; Confidence factor)
R_{QU}	Rules for Cluster	Rule 1 (434; 0.961) R_{QUEST}**11**
	Rules for Cluster	Rule 1 (781; 0.986) R_{QUEST}**21**
	Rules for Cluster	Rule 1 (214; 0.963) R_{QUEST}**31**
	Rules for Cluster 4 R_{QUEST}**4**	Rule 1 (231; 0.944) R_{QUEST}**41**
		Rule 2 (81; 0.889) R_{QUEST}**42**
	Rules for Cluster	Rule 1 (206; 0.971) R_{QUEST}**51**
	Rules for Cluster	Rule 1 (437; 0.95) R_{QUEST}**61**
	Default: Cluster	

Table 5: Rule Composition For Each Cluster Corresponding To The Models Generating The Rules.

	Number or Rules cluster wise	**Total number of rules**
C5 Rule Composition: $R_{C5}11 + R_{C5}12 + R_{C5}13 + R_{C5}14 + R_{C5}15 + R_{C5}16 \rightarrow$ **$R_{C5}1$** $R_{C5}21 + R_{C5}22 + R_{C5}23 \rightarrow$ **$R_{C5}2$** $R_{C5}31 + R_{C5}32 \rightarrow$ **$R_{C5}3$** $R_{C5}41 + R_{C5}42 + R_{C5}43 + R_{C5}44 + RC545 + R_{C5}46 + R_{C5}47 \rightarrow$ **$R_{C5}4$** $R_{C5}51 + R_{C5}52 + R_{C5}53 + R_{C5}54 + R_{C5}55 \rightarrow$ **$R_{C5}5$** $R_{C5}61 + R_{C5}62 + R_{C5}63 + R_{C5}64 + R_{C5}65 + R_{C5}66 + R_{C5}67 + R_{C5}68 + R_{C5}69 \rightarrow$ **$R_{C5}6$**	Cluster1 (**6 R**) Cluster2 (**3 R**) Cluster3 (**2 R**) Cluster4 (**7 R**) Cluster5 (**5 R**) Cluster6 (**9 R**)	**32 R**
CRT Rule Composition: $R_{CRT}11 + R_{CRT}12 + R_{CRT}13 + R_{CRT}14 \rightarrow$ **$R_{CRT}1$** $R_{CRT}21 + R_{CRT}22 \rightarrow$ **$R_{CRT}2$** $R_{CRT}31 + R_{CRT}32 \rightarrow$ **$R_{CRT}3$** $R_{CRT}41 + R_{CRT}42 + R_{CRT}43 \rightarrow$ **$R_{CRT}4$** $R_{CRT}51 + R_{CRT}52 \rightarrow$ **$R_{CRT}5$** $R_{CRT}61 + R_{CRT}62 + R_{CRT}63 \rightarrow$ **$R_{CRT}6$**	Cluster1 (**4 R**) Cluster2 (**2 R**) Cluster3 (**2 R**) Cluster4 (**3 R**) Cluster5 (**2 R**) Cluster6 (**3 R**)	**16 R**
CHAID Rule Composition: $R_{CHAID}11 + R_{CHAID}12 + R_{CHAID}13 \rightarrow$ **$R_{CHAID}1$** $R_{CHAID}21 \rightarrow$ **$R_{CHAID}2$** $R_{CHAID}31 \rightarrow$ **$R_{CHAID}3$** $R_{CHAID}41 + R_{CHAID}42 \rightarrow$ **$R_{CHAID}4$**	Cluster1 (**3 R**) Cluster2 (**1 R**) Cluster3 (**1 R**) Cluster4 (**2 R**) Cluster5	**11 R**

$R_{CHAID}51 \rightarrow \mathbf{R_{CHAID}5}$ $R_{CHAID}61 + R_{CHAID}62 + R_{CHAID}63 \rightarrow \mathbf{R_{CHAID}6}$	(**1 R**) Cluster6 (**3 R**)	
QUEST Rule Composition: $R_{QUEST}11 \rightarrow \mathbf{R_{QUEST}1}$ $R_{QUEST}21 \rightarrow \mathbf{R_{QUEST}2}$ $R_{QUEST}31 \rightarrow \mathbf{R_{QUEST}3}$ $R_{QUEST}41 + R_{QUEST}42 \rightarrow \mathbf{R_{QUEST}4}$ $R_{QUEST}51 \rightarrow \mathbf{R_{QUEST}5}$ $R_{QUEST}61 \rightarrow \mathbf{R_{QUEST}6}$	Cluster1 (**1 R**) Cluster2 (**1 R**) Cluster3 (**1 R**) Cluster4 (**2 R**) Cluster5 (**1 R**) Cluster6 (**1 R**)	**7 R**

Coincidence matrices of all the models shows the accurately and misclassified number of instances for all clusters corresponding to rules generated by the model. The integrated rule based model is built by combining the qualifying rules generated by each model for each cluster. Qualifying rules are those rules whose misclassification is less than threshold set by the user in terms of percentage of total instances in actual cluster. Here we have set misclassification threshold to be 10 %. The qualifying rules are selected based on rule confidence factor and coincidence matrix applying the qualifying criteria.

According to coincidence matrix of C5 all the rules are qualifying rules as misclassification is less than 10% for all rules. All rules of CART model are also qualifying rules as their misclassification is below 10%. For CHAID model rules generated for cluster 1 , cluster 4 and cluster 5 are misclassifying the instances more than acceptable limit hence for these clusters only the rules having confidence factor more than 0.8 are qualifying rules(threshold is optional for the user). For cluster 1 RCHAID12 and RCHAID13 have been selected as qualifying and RCHAID 11 as no qualifying rule, for cluster 4 RCHAID41 as qualifying rule and RCHAID42 as no qualifying rule and for cluster 5 there is only one rule RCHAID51 which is no qualifying. All rules of Quest model are also qualifying as their misclassification is less than 10%. Rules for integrated rule based model (TPC-IRBM) for each cluster are integration of all qualifying rules generated by each predictive model. The rule composition of integrated rule based model is shown in table6.

Table 6: Rule composition for each cluster corresponding to the TPC-IRBM model

	Number or Rules cluster wise	**Total number of rules**
TPC-IRBM Rule Composition: $R_{C5}11 + R_{C5}12 + R_{C5}13 + R_{C5}14 + R_{C5}15 + R_{C5}16 + R_{CRT}11 + R_{CRT}12 + R_{CRT}13 + R_{CRT}14 + R_{CHAID}12 + R_{CHAID}13 + R_{QUEST}11 \rightarrow \mathbf{R1}$ $R_{C5}21 + R_{C5}22 + R_{C5}23 + R_{CRT}21 + R_{CRT}22 + R_{CHAID}21 + R_{QUEST}21 \rightarrow \mathbf{R2}$ $R_{C5}31 + R_{C5}32 + R_{CRT}31 + R_{CRT}32 + R_{CHAID}31 + R_{QUEST}31 \rightarrow \mathbf{R3}$ $R_{C5}41 + R_{C5}42 + R_{C5}43 + R_{C5}44 + RC545 + R_{C5}46 + R_{C5}47 + R_{CRT}41 + R_{CRT}42 + R_{CRT}43 + R_{CHAID}41 + R_{QUEST}41 + R_{QUEST}42 \rightarrow \mathbf{R4}$ $R_{C5}51 + R_{C5}52 + R_{C5}53 + R_{C5}54 + R_{C5}55 + R_{CRT}51 + R_{CRT}52 + R_{QUEST}51 \rightarrow \mathbf{R5}$ $R_{C5}61 + R_{C5}62 + R_{C5}63 + R_{C5}64 + R_{C5}65 + R_{C5}66 + R_{C5}67 + R_{C5}68 + R_{C5}69 + R_{CRT}61 + R_{CRT}62 + R_{CRT}63 + R_{CHAID}61 + R_{CHAID}62 + R_{CHAID}63 + R_{QUEST}61 \rightarrow \mathbf{R_{C5}6}$	Cluster1 (**13 R**) Cluster2 (**7 R**) Cluster3 (**6 R**) Cluster4 (**13 R**) Cluster5 (**8 R**) Cluster6 (**16 R**)	**63 R**

The accuracy of each predictive model for training, testing and validation dataset is compared. The accuracy of C5 model is highest i.e. 99.96% for training dataset, 98.15 % for testing dataset and 98.85 % for validation dataset. For CART it is 98.95% for training dataset, 96.97% for testing dataset and 97.42% for validation dataset. For QUEST it is 96.39% for training dataset, 97.36 % for testing dataset and 97.13% for validation dataset. For CHAID it is lowest and is 87.88% for training dataset, 87.34 % for testing dataset and 91.12% for validation dataset.

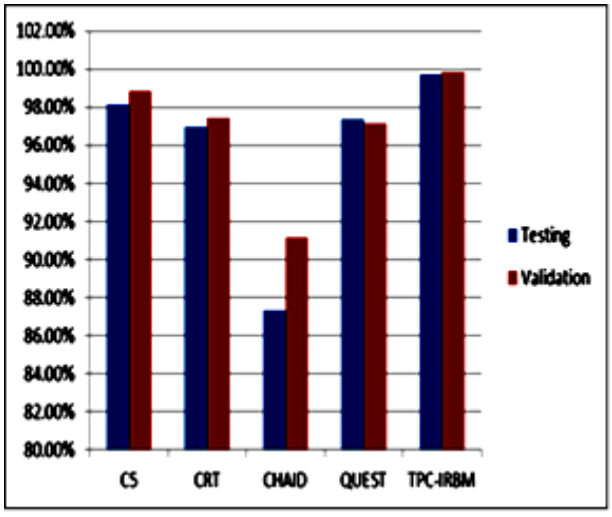

Fig3. Comparative view of accuracy of models

The accuracy of integrated rule based model evaluating on test data set and validation dataset is improved compared to all models individual results and is 99.7% and 99.8% with an upper bound of 100% comparative view of results of models are shown in figure3.

5. CONCLUSION

Proposed method in this article is quite useful and easy to implement for clustering large biological data with large number of features. Rules generated by integrating the qualifying rules of various rule based model are high quality rules to achieve higher accuracy with much less complexity compared to neural network clustering methods,graph based and model based clustering methods with very marginal improvement in predicted accuracy for semi supervised clustering.

6. FUTURE WORK

As the proposed model works in two phases. First phase for generating and preparing TTV data for second phase. We have used K-means clustering method in first phase and Heirarichal ,SOM for estimation of lower and upper number of clusters respectively. More robust methodology can be applied to improve the quality of prediction and clustering in first phase.In second phase the criteria that we have chosen for qualifying rules to be used in TPC-IRBMis confidence factor(misclassification rate) as threshold. The other criteria for rule integration and qualification can be explored in future work.

7. REFERENCES

[1] Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. Classification and Regression Trees. New York: Chapman & Hall/CRC.

[2] Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.

[3] Quinlan, J. (1996). Bagging, Boosting, and C4.5, Proceedings of the Thirteenth National Conferenceon Artificial Intelligence, Portland , Oregon (American Association for Artificial Intelligence Press, Menlo Park, California), pp. 725 – 730.

[4] Rulequest Research. (2013) See5/c5.0.[Online]. Available: http://www.rulequest.com/see5-info.html.

[5] Kass, Gordon V.; An Exploratory Technique for Investigating Large Quantities of Categorical Data, Applied Statistics, Vol. 29, No. 2 (1980), pp. 119–127.

[6] Loh, W. Y., and Y. S. Shih. 1997. Split selection methods for classification trees. Statistica

[7] www.ncbi.nlm.nih.gov

[8] Clementine® 11.1 Algorithms Guide , Copyright © 2007 by Integral Solutions Limited.

[9] Lander E.S. , "Array of hope," Nature genet.,21,3-4,1999.

[10] Eisen,M.B. et al., " Cluster analysis and display of genome-wide expression patterns," Proc. Natl. Acad. Sci. Am., 95, 14863–14868, 1998..

[11] Tavazoie,S. et al., " Systematic determination of genetic network architecture," Nat. Genet., 22, 281–285, 1999..

[12] Tamayo P. et al., "Interpreting patterns of gene Expression with self organizing maps: methods and application to hematopoieitic differentiation," Proc. Natl acad.Sci. USA,96,2907-2912,1999.

[13] Ben-Dor,A. and Yakhini Z., "clustering gene Expression patterns," inRECOMB99: Proceedings of the third annual international conference on computational molecular biology. Lyon,france,1999,pp.33-42.

[14] Hartuv E. et al., "An algorithm for clustering cDNAs for gene expression analysis," in RECOMB99: Proceedings of the third annual international conference on computational molecular biology. Lyon,france,1999, pp.188-197.

[15] Browm M.P.S., et al., "Knowledge based analysis of micro array gene expression data using support vector machine," Proc. Natl acad.Sci. USA,97,262-267, 2000.

[16] Yeung K.Y.,et al., "Model based clustering and data transformations for gene expression data," Bioinformatics , Vol. 17 no. 10 2001, 2001, pp.977-987.

[17] Moyses Nascimento, et al., " Bayesian model based clustering of temporal gene expression using autoregressive panel data approach," Bioinformatics,Vol. 28 no. 15 2012, 2012, pp. 2004-2007.

[18] Haberman RP, Colantuoni C, Stocker AM, Schmidt AC et al. Prominent hippocampal CA3 gene expression profile in neurocognitive aging. Neurobiol Aging 2011 Sep;32(9):1678-92. PMID: 19913943

[19] C G. Piatetsky-Shapiro, T. Khabaza, S. Ramaswamy apturing Best Practice for Microarray Gene Expression Data Analysis, , in Proceedings of KDD-2003 (ACM Conference on Knowledge Discovery and Data Mining), Washington, D.C., 2003.

[20] Gregory Piatetsky-Shapiro and Pablo Tamayo Microarray Data Mining: Facing the Challenges, SIGKDD Explorations, Dec 2003.