I.J. Information Technology and Computer Science, **2017, 3, 71-79**
Published Online March 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2017.03.08

Priority Based New Approach for Correlation Clustering

Aaditya Jain
M.Tech Scholar, Department of Computer Science & Engg., R. N. Modi Engineering College, Rajasthan Technical University, Kota, Rajasthan, India
E-mail: aadityajain58@gmail.com

Dr. Suchita Tyagi
Associate Professor, Department of Computer Science & Engg., Sushila Devi Bansal College of Technology, Indore, MP, India
E-mail: suchitatyagi625@gmail.com

***Abstract*—**Emerging source of Information like social network, bibliographic data and interaction network of proteins have complex relation among data objects and need to be processed in different manner than traditional data analysis. Correlation clustering is one such new style of viewing data and analyzing it to detect patterns and clusters. Being a new field, it has lot of scope for research. This paper discusses a method to solve problem of chromatic correlation clustering where data objects as nodes of a graph are connected through color-labeled edges representing relations among objects. Purposed heuristic performs better than the previous works.

***Index Terms*—**Clustering Problems, Correlation Clustering, Chromatic Balls, and Priority Based Chromatic Balls.

I. Introduction

Clustering is an unsupervised from of machine learning aiming at grouping of data objects in a way that similar objects fall in the same group specifically called a "cluster". The traditional clustering algorithms like k-means [1] and fuzzy c-means [2] use the notation of similarity or closeness among objects to group them. Thus, they view objects as having binary or fuzzy relationship between them. The binary relationship categorizes which clusters are similar and should be grouped in the same cluster using some similarity / distance metric between them. The fuzzy relations, on the other hand, deduce a percentage of similarity between data objects, with the ones with higher percentage probable to fall in the same cluster. In real world problem, the relations among objects are more complex. Like those existing among people in social networks, who have varying kind of relationships family, professional, friendly etc. Such scenarios of complex relations also exist in authored documents library, protein-protein interactions etc.

Scenarios discussed above are best described through categorical relationships among objects, easily represented through graphs. Using graphic is advocated due to

- They are flexible and intense data structures.
- They can be easily ranged from very simple to very complicated relationships.
- They can be used to represent many kinds of relations, whether independent or co-existing.

Once a graph has been formed, the problem of analysis is converted into problem of partitioning the graph. Bansal et al. defined the problem of Correlation Clustering in [3]. It was successful enough to eradicate all the issues encountered in the traditional clustering algorithms so is being used in many applications like parallel and distributed system, pattern recognition, and image segmentation. Bonchi et al [4] further extended the concept of correlation clustering to chromatic correlation clustering by assigning colors to edges instead of positive or negative signed labels as used in correlation clustering. This paper presents a contribution in the direction of solving chromatic correlation clustering problem through revisiting the work of Bonchi et al [4, 5]. A Priority Based Chromatic Balls algorithm is presented to increase the probability of better solution of the algorithm and keeping its advantages of speed retained.

The rest of the paper is organized as follows. Section II describes brief literature search related to this work. In section III Chromatic Balls algorithm is described with its drawbacks to show the problem part. Section IV describes the proposed algorithm with its both versions. The experimental setup and comparative results are provided in section V and VI. Finally the paper concludes in section VII.

II. Related Work

A lot of research is headed in this direction for years by many authors. Detail analysis and literature search on this topic is done in my previous work [6]. Some of them introduced here.

Bansal et al in 2004 [3] introduced the concept of

Correlation Clustering. The clustering methodology required edge-labeled graphs with edges signed as positive or negative. Clustering depends on edge labels and can have any number of clusters. Clustering is based on the notion of maximizing agreements and minimizing disagreements. Here, agreement means the sum of the number of positively signed edges inside clusters and number of negatively signed edges between clusters. Disagreement, therefore, means the sum of number of negatively signed edges inside clusters and number of positively signed edges between clusters. Mathematically expressed, for a graph G= (V,E), where V is the set of objects to be clustered and E edges denoting relationships between V, a function $s: E \rightarrow \{+, -\}$ is defined to assign a sign for each edge, sign + denoting the similarity and – denoting dissimilarity. Therefore, for correlation clustering, a signed graph as (G,s) is used. Any similarity distance or real distance is used for the signing of edges. In any situation if it is not possible to put positive edges in a single cluster a trivial solution is agreeing with half of the edge labels for clustering for example in case of more positive edges and less negative edges, all the vertices could be put in a single big cluster and if not, then each vertex would lie in a different cluster [7,8]. Another feature of this clustering approach compare to conventional clustering is that it does not require any prior knowledge of the number of clusters to be formed.

V. Guruswami et al [9] in 2006 focused on the effect of keeping the number of clusters, k, fixed for the Correlation Clustering problem. The authors achieved a Polynomial Time Approximation Scheme (PTAS) for k>2 for both the maximizing agreements and minimizing disagreements case. Achieving PTAS was a trivial task for the minimizing disagreements problem which otherwise was observed to be an APX-hard problem when the constant k is not specified.

N. Ailon et al [10] in 2009 focused on the agnostic nature of the Bansal et al's work that assumes the non-existence of any ground clustering with the solution cost computed against the input similarity function. Opposed to the assumption, the authors assumed that an unknown ground truth clustering exists and that the accuracy of the resultant clustering should be measured against the ground clustering. Provable approximation guarantees are provided by the authors.

Chaoli Wang et al [11] provided a study of hierarchical clustering of volumetric data having correlation relations. Not much work in this direction has been proposed earlier. The authors proposed three clustering algorithms which on the basis of quality threshold, k-means and random walks investigate the correlation relations of the data in a climate dataset. Evaluation and qualitative and quantitative comparison of the algorithms concludes the efficacy of the proposal.

Inspired from the Correlation Clustering, Bonchi et al [4, 5] extended the work by assigning colors to edges instead of signs. These colors acted as labels to the edges. Similarly colored edges showed similar relations between the adjoining vertices and hence are expected to fall in the same cluster. An objective function was introduced for ensuring that the edges within a cluster are as much as possible, of same color. Their algorithm named Chromatic Balls is a randomized algorithm for solving the chromatic clustering problem.

Correlation Clustering by Bansal et al [3] was encountered having issues in its average case models, to which Yi Makarychev et al in [12] proposed a semi-random model of Correlation Clustering. The average case models were found realistically impossible. Also, each pair of vertices had the same amount of similarity or dissimilarity which made clustering difficult. Two approximation algorithms were also proposed by authors in [12] in their semi-random model. The first algorithm had a Polynomial-Time Approximation Scheme (PTAS) for the instances and the second algorithm was a recovery algorithm for the planted partition giving a small classification error η.

Kookjin Ahn et al [13] focused on clustering correlated objects in a dynamic data stream model. Unlike the simple data stream model consisting of sequenced edges with their labels (referred to as weights in the paper), the associated data stream updates the edge labels of the related edge labeled graph containing n nodes as it arrives. The updates include insertions and deletions of edges. Three types of weights are considered: unit weights containing a set of only unit positive and unit negative edges, bounded weights which should necessarily be non zero and bounded by some constant and lastly, arbitrary weights consisting of all weights of O(poly n). The objective behind the proposal is to find a node partition efficiently able to partition the negatively labeled edges in different cluster and the positively labeled edges in the same cluster. For ensuring the quality of the associated node partition, authors develop data structures based on linear sketches. To solve the space- approximation problem in O (n.polylog n) space, the developed data structures are then combined with convex programming and sampling techniques.

III. Chromatic Balls

Chromatic Balls (CB) algorithm [4, 5] is a randomized approximation algorithm to solve the chromatic clustering problem. The basic working of the Chromatic Balls can be understood as a method to form groups of similar edges (relations). The algorithm takes input an edge-labeled graph and process it in iterations. The iterations continue until all edges have been removed from the graph. At each iteration, an edge(x, y) is picked randomly and the two vertices x and y are included in current cluster. Also all the vertices z for which edge(x, z) and (y, z) have same label as edge(x, y) are also included in current cluster. Then all the vertices included in the current cluster are removed from the graph. A new cluster is formed in each iteration and is named using the label of selected pivot edge. When all edges have been removed, the remaining isolated vertices are given a different cluster label, that is, they form singleton clusters. The algorithm is outlined as algorithm 1 discussed in fig.1.

Algorithm 1 Chromatic Balls

Input: Edge labeled graph $G = (V, E, L, l_0, l)$ where $l: V_2 \rightarrow L \cup \{l_0\}$
Output: Clustering $C: V \rightarrow N$; cluster labeling function $cl: [V] \rightarrow L$
Step 1: $i \leftarrow 1$
Step 2: While $E \neq \emptyset$ do steps 3-8
Step 3: Pick an edge $(x, y) \in E$ uniformly at random
Step 4: $C \leftarrow \{x, y\} \cup \{z \in V | l(x, z) = l(y, z) = l(x, y)\}$
Step 5: $C(x) \leftarrow i$, for all $x \in C$
Step 6: $cl(i) = l(x, y)$
Step 7: Remove C from $G: V \leftarrow V \backslash C, E \leftarrow E\{(x, y) \in E | x \in C\}$
Step 8: $i \leftarrow i + 1$
Step 9: For all $x \in V$ do steps 10 to 12
Step 10: $C(x) \leftarrow i$
Step11: $cl(i) \leftarrow a$ label from L
Step 12: $i \leftarrow i + 1$

Fig.1. Chromatic balls algorithm

A. *Drawback of Chromatic Balls*

Since Chromatic Balls is a heuristic algorithm, it cannot produce optimum clustering as an output all the time. Hence aim is to produce clusters "near-to-optimum" and any such output can be called of high quality. The quality of output depends on the probability of a "good" edge to get selected as the pivot edge. Since all edges have equal probability of getting selected as pivot in chromatic balls, the probability of obtaining good quality result is low.

IV. Priority Based Chromatic Balls

The randomized behavior of Chromatic Balls does not guarantee good results every time. The problem arises when the pivot edge chosen does not belong to the desire optimum solution.

Definition: “Good” edge is an edge (x, y) such that l(x, y) =cl(C(x))=l, C(x)=C(y) in ground truth or optimal clustering.

Properties of “Good” edge: It will always be part of a l-colored clique. The majority of other edges incident on x or y are also of label l.

To increase the probability that selected edge might be part of optimum solution, we restrict the space of selection. Instead of picking any edge uniformly at random, the edge is picked with probability proportional to the frequency of label of edge. It is similar to construction of a priority queue of all edges based on the frequency of the labels. Pivot edge is selected from among the edges with highest priority in the queue at random. Thus, advantage of the speed of randomized algorithm is retained while the probability of better solution is increased. The proposed algorithm has been outlined in fig.2. The first step constructs the priority queue of edges. Rest all the steps are same as Chromatic Balls for step 4 and 8 the criterion of while loop.

Algorithm 2(a) Proposed Priority Based Chromatic Balls (Version 1)

Input: Edge labeled graph $G = (V, E, L, l_0, l)$ where $l: V_2 \rightarrow L \cup \{l_0\}$
Output: Clustering $C: V \rightarrow N$, cluster labeling function $cl: [V] \rightarrow L$
Step 1: Construct a priority queue Q of edges based on frequency of label.
Step 2: $i \leftarrow 1$
Step 3: While $Q \neq \emptyset$ do steps 4 to 9
Step 4: Remove an edge(x, y) from Q with highest priority. If more than one edge has equal priority, pick any one uniformly at random.
Step 5: $C \leftarrow \{x, y\} \cup \{z \in V | l(x, z) = l(y, z) = l(x, y)\}$
Step 6: $C(x) \leftarrow i$, for all $x \in C$
Step 7: $cl(i) = l(x, y)$
Step 8: Remove C from graph, $G: V \leftarrow V \backslash C, E \leftarrow E \backslash \{(x, y) \in E | x \in C\}$ and $Q \leftarrow Q \backslash \{(x, y) \in E | x \in C\}$
Step 9: $i \leftarrow i + 1$
Step 10: For remaining vertices in V do step 11 to 13
Step 11: $C(x) \leftarrow i$
Step 12: $cl(i) \leftarrow a$ label from L
Step 13: $i \leftarrow i + 1$

Fig.2. Priority based chromatic balls algorithm (Version 1)

Another version of proposed algorithm is presented in fig.3. The conventions followed are same as Version 1 except that a separate clustering label function is not required to map between cluster number and its color label. Rather it is assume that L is a set of labels expresses through ordinal numbers rather than actual categorical values. For example, L-{Red, Blue, Green} is input to the algorithm in the form L={1, 2, 3}.

Algorithm 2(b) Proposed Priority Based Chromatic Balls (Version 2)

Input: Edge labeled graph $G = (V, E, L, l_0, l)$ where $l: V_2 \rightarrow L \cup \{l_0\}$
Output: Clustering $C: V \rightarrow N$
Step 1: Construct a priority queue Q of edges based on frequency of label.
Step 2: While $Q \neq \emptyset$ do steps 3 to 6
Step 3: Remove an edge(x, y) from Q with highest priority. If more than one edge has equal priority, pick any one uniformly at random.
Step 4: Decoding member of cluster as $C \leftarrow \{x, y\} \cup \{z \in V | l(x, z) = l(y, z) = l(x, y)\}$
Step 5: $C(x') = l(x, y), \forall x' \in C$
Step 6: Remove C from graph, $G: V \leftarrow V \backslash C, E \leftarrow E \backslash \{(x, y) \in E | x \in C\}$ and $Q \leftarrow Q \backslash \{(x, y) \in E | x \in C\}$
Step 7: $i \leftarrow \max(C)$
Step 8: For remaining vertices in V do step 9 to 10
Step 9: $i \leftarrow i + 1$
Step 10: $C(x) \leftarrow i$

Fig.3. Priority based chromatic balls algorithm (Version 2)

V. Methodologyanalysis

The performance of the proposed Priority Based Chromatic Balls (PCB) algorithm has been analyzed on synthetic data generated using the Synthetic Data Generator Algorithm as shown in fig.4. All the data generation and experiments are conducted on MATLAB platform. MATLAB is a very powerful development and simulation tool. The experiments are conducted by a varying a number of parameter such as numbers of vertices, numbers of labels, noise levels etc. Later sections discuss the details of the experiments.

Algorithm 3 Synthetic Data Generator

Input: Number of vertices n, number of clusters K, number of labels h, probability p of intra-cluster edges, probability q of inter- cluster edges, probability w that an edge inside a cluster has a color different from the cluster, probability v that one ground cluster is larger than all others.
Output: Edge labeled graph $G = (V, E, L, l_0, l)$
Step 1: $V \leftarrow [1, n], E \leftarrow \emptyset, L \leftarrow \{l_1, l_2, \ldots \ldots, l_h\}$
Step 2: For $\lceil n * v \rceil$ vertices out of n, produce a clustering C by assigning each vertex $x \in V$ to a cluster selected uniformly at random. Assign remaining vertices to a single randomly selected cluster.
Step 3: Assign to each cluster a label selected uniformly at random from L.
Step 4: for all pair $(x, y) \in V_2$ do
Step 5: Pick 3 random numbers $r_1, r_2, r_3 \in [0,1]$
Step 6: if $C(x) = C(y)$ then
Step 7: if $r_1 < p$ then
Step 8: if $r_2 < w$ then
Step 9: $E \leftarrow E \cup \{(x, y)\}$
Step10: $l(x, y) \leftarrow a$ random label from $L \backslash \{cl(C(x))\}$ else
Step 11: $E \leftarrow E \cup \{(x, y)\}$
Step 12: $l(x, y) \leftarrow cl(C(x))$
Step 13: end if
Step 14: end if
Step 15: else if $r_3 < q$ then
Step 16: $E \leftarrow E \cup \{(x, y)\}$
Step 17: $l(x, y) \leftarrow a$ random label from L
Step 18: end if
Step 19: end for

Fig.4. Synthetic data generator algorithm

The experiments on the generated synthetic data of varying parameters have been performed on CB and the proposed PCB for performance comparison. Control parameters help in the proper analysis of the proposed work in all aspects denotes as:

- Number of vertices denote by n.
- Number of labels denote by h.
- Number of ground truth clusters denote by K.
- Probability of sampling intra-cluster edges denote by p
- Probability of sampling inter-cluster edges denote by q.
- Probability of having an edge of a color different from the cluster is w.
- Probability of adding prominence to a label is v, (if v=1,there are equal edges of all labels)

The attributes tested upon are the runtime, number of clusters formed, cost and isolated points depending upon the parameters varied. With varying parameters, characteristics of the graphs generated using Synthetic Data Generator algorithm listed above change in the following ways

- Ratio of intra-cluster edges to inter-cluster edges (by varying the probabilities p and q).
- Uniform /Non uniform distribution of labels over edges (by keeping all labels of equal importance or any one label "prominent" among all the other levels).
- Density of the edges (by varying K).
- Ratio of different colored edges within a cluster (by varying w).

VI. Experimental Results

The proposed PCB is analyzed and compared with the standard CB algorithm on the basis of the effect observed when the discussed parameters are varied. This section gives the details of the same.

A. Effect of the Number of Vertices

This section discusses the effect of parameter n practically. According to the varying criteria of the control parameters, the effect is noted.

1. With Equal Number of Same and Different Colored Edges: The number of vertices (n) of the graph representing the data points is varied from 100 till 500, first for equal number of same and different colored edges of the graph (w=0.5), keeping other parameters constant at values K=10, h=4, p=0.5 and v=1. Table 1 lists the experimental results for varying n.

Table 1. Effect of the Number of Vertices on Runtime and Cost of the Algorithms-I

	Runtime(in seconds)		Cost	
n	**CB**	**PCB**	**CB**	**PCB**
100	0.0276 ±0.010	0.0306 ±0.004	1138 ±4	1136 ±3
200	0.1807 ±0.050	0.1677 ±0.1677	4666 ±5	4678 ±2
300	1.0808 ±0.400	0.5316 ±0.5316	10423 ±6	10396 ±8
400	2.235 ±0.759	1.2890 ±0.070	18304 ±18	18262 ±15
500	6.5317 ±1.479	2.3198 ±0.094	28778 ±27	28708 ±21

Figure 5 illustrates the effect of varying n on runtime of the CB and proposed PCB algorithms. And figure 6 shows the corresponding cost growth plot of the proposed PCB algorithm. As observed, both the runtime and cost increase with increasing n since the overall size of dataset is increasing. As compared to CB, PCB has lesser cost and lower run time.

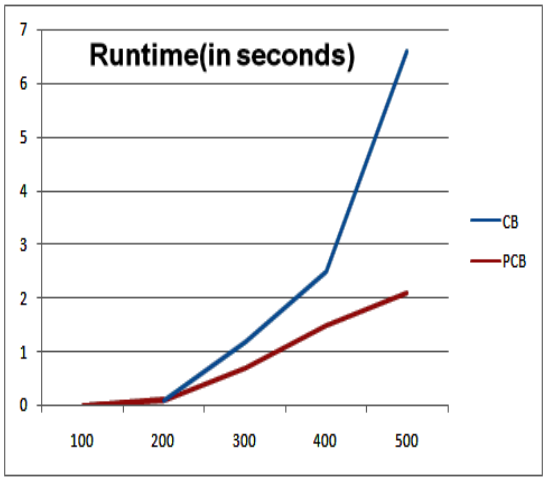

Fig.5. Growth of runtime with n-I

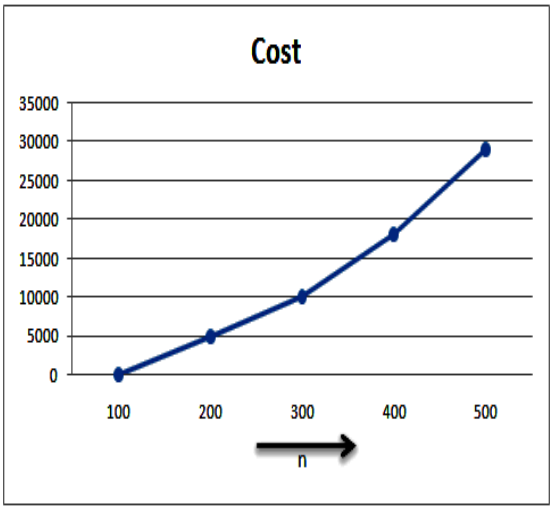

Fig.6. Growth of cost of PCB with n-I

2. With More Edges of Single Color: The effect of increasing n can be also seen when there are more edges of the same color per cluster by keeping probability "p" at a high value of 0.7 and "w" at a small value 0.4 and the other parameters K, h, q and v constant 10, 3, 0.1 and 1 respectively. Table 2 lists the result obtained.

Table 2. Effect of the Number of Vertices on Runtime and Cost of the Algorithm-II(a)

	Runtime(in seconds)		Cost	
n	**CB**	**PCB**	**CB**	**PCB**
100	0.0213 ±0.009	0.0223 ±0.001	802 ±8	795 ±4
200	0.1821 ±0.100	0.1357 ±0.003	3153 ±8	3134 ±6
300	0.4363 ±0.171	0.4018 ±0.032	7182 ±19	7127 ±13
400	1.1016 ±0.204	1.1016 ±0.064	12913 ±32	12837 ±11
500	2.9068 ±1.208	1.7832 ±0.092	20172 ±28	20095 ±37

Figure 7 and 8 shows the corresponding growth plots of runtime and cost with n. The growth of runtime and cost with n is linear as expected. PCB shows a better performance in terms of runtime and cost when compared with CB.

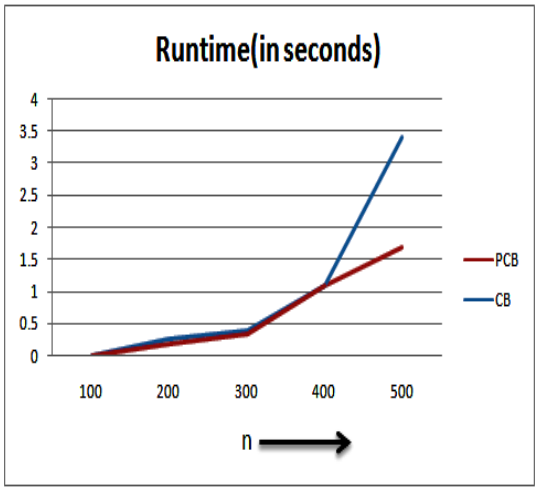

Fig.7. Growth of runtime with n=II

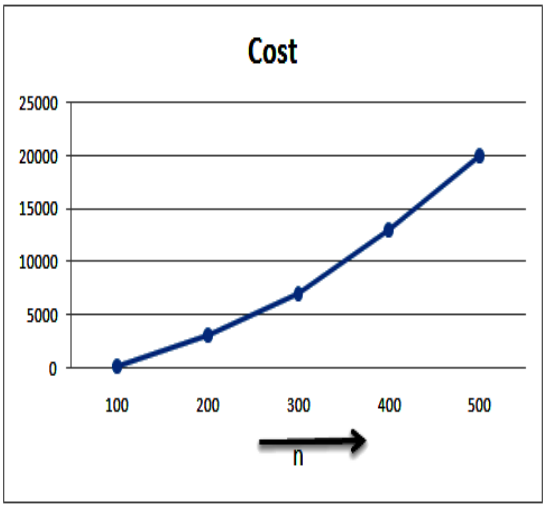

Fig.8. Growth of cost of PCB with n-II

3. With One Label Prominent over Others: In the last set of experiments, the effect of n on keeping any one label permanents over the other is analyzed. The value of parameter "v" is fixed at 0.8. The values of parameters K, h, p, q and w are 8, 3, 0.7, 0.1 and 0.5 respectively. The parameter n is varied from 300 till 800. The value of parameter "K" is kept small keeping the clusters dense. Table 3 lists the results obtained.

Table 3. Effect of the Number of Vertices on Runtime and Cost of the Algorithms-II(b)

	Runtime(in seconds)		Cost	
n	**CB**	**PCB**	**CB**	**PCB**
300	0.5171 ±0.096	0.3966 ±0.016	8855 ±27	8813 ±34
400	1.7104 ±0.346	0.884 ±0.050	15902 ±37	15828 ±29
500	3.6463 ±1.488	1.776 ±0.072	24430 ±45	24366 ±75
600	7.2052 ±2.767	2.859 ±0.205	36968 ±86	36820 ±194
700	16.4317 ±5.101	4.024 ±0.170	47831 ±85	47784 ±55
800	27.6317 ±16.824	6.149 ±0.201	65446 ±98	65378 ±195

Figure 9 demonstrates the growth in runtime and cost with n and figure 10 is the growth plot for cost of the proposed algorithm with n. The results show linear growth in runtime and cost for both the algorithms. But

PCB performing better with deceased values of runtime and cost compare to CB.

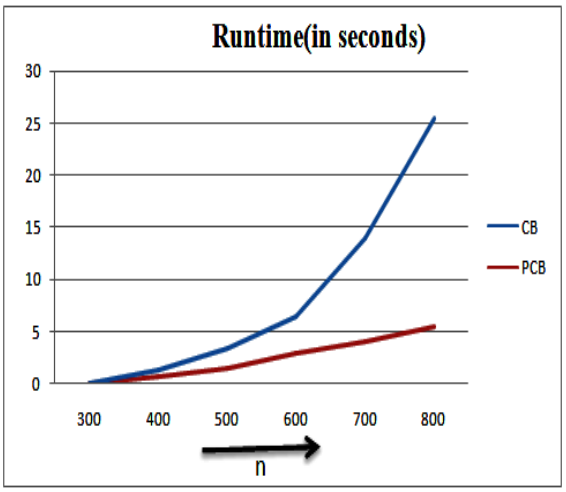

Fig.9. Growth of runtime with n-III

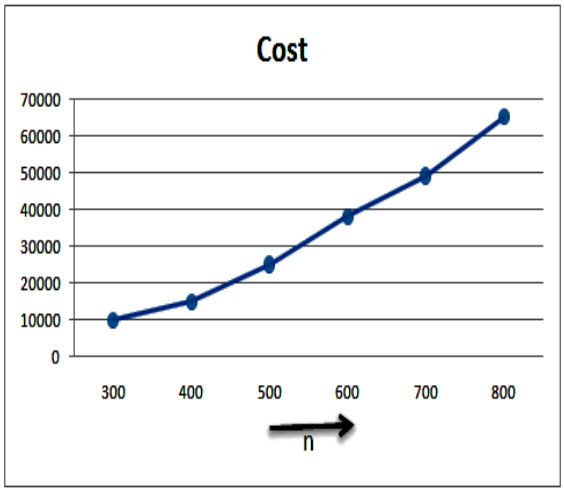

Fig.10. Growth of cost of PCB with n-III

B. *Effect of the Number of Ground Truth Clusters*

The parameter K, representing the number of clusters in ground truth clustering is varied keeping the probabilities of sampling intra-cluster and inter-cluster edges almost equal and making a label prominent by probability v equal to 0.7. The more the number of ground-truth clusters, the sparser each cluster is, because of the edges being distributed in every cluster easily and not confining to one. Since the clusters are distinct, they can be easily identified and hence, the cost is lesser with increasing parameter K, when running both the algorithms. The second effect of increasing parameter K is increasing number of total cluster formed. Table 4 lists the experimental result with varying K at n=1000, h=5, p=0.6, q=0.05, w=0.5, v=0.7.

Figure 11 shows the total cluster output obtained through CB and proposed PCB with increasing K. Figure 12 shows the growth plot of cost of PCB with K. A decrease in the cluster count and cost of proposed PCB, through slight, can be observed through the results favoring the algorithm over CB.

Table 4. Effect of the Number of Ground Truth Clusters on Total Clusters and Cost of the Algorithms

	Total cluster output		Cost	
K	**CB**	**PCB**	**CB**	**PCB**
10	260±8	252±10	77934±151	77733±165
15	304±8	305±10	66580±115	66248±54
20	328±32	320±6	62253±139	62016±30
25	351±5	343±6	60806±157	60598±29
30	363±5	351±5	58086±104	58007±105

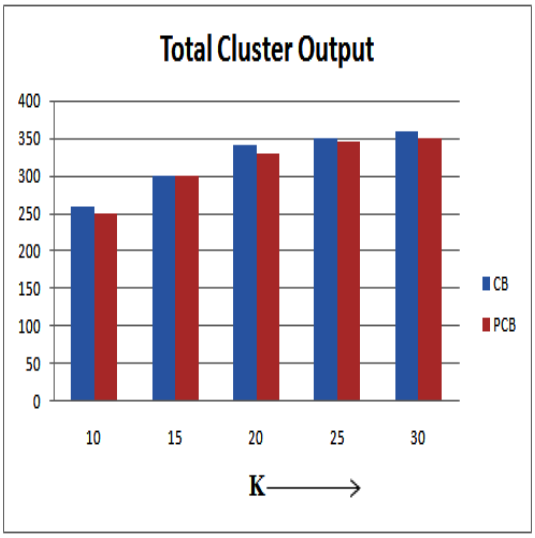

Fig.11. Growth of total cluster with K

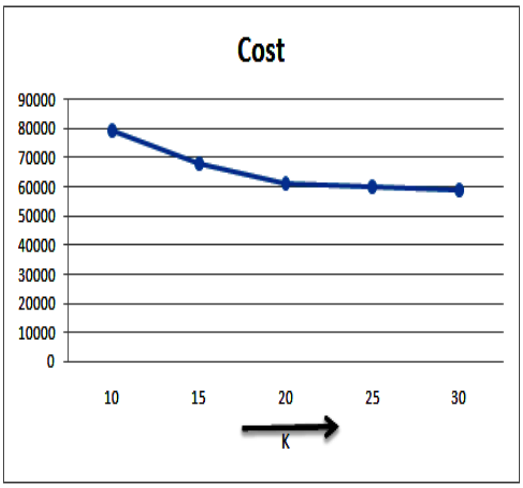

Fig.12. Growth of cost of PCB with K

C. *Effect of the Number of Labels*

Both the algorithms are tested for varying number of labels, h. With increasing parameter h, the number of clusters per color label increases, thereby reducing the cost. Table 5 lists the results of the experiments conducted on both the algorithms for increasing h at n=1000, K=20, p=0.6, q=0.05 w=0.5 and v=0.7.

Table 5. Effect of the Number of Labels on Total Clusters And Cost of the Algorithms

	Total cluster output		Cost	
h	CB	PCB	CB	PCB
2	264±8	260±5	64361±82	64308±245
3	316±5	310±8	63598±64	63522±98
4	326±4	324±5	62021±105	61980±141
5	329±15	328±6	62260±93	62279±396
6	340±12	328±7	62870±143	62666±116

The proposed PCB performs better than CB both in term of cluster-count and cost. Figure 13 a column graph shows the growth in the total clusters obtained by both the algorithms. Figure 14 shows the growth in cost of PCB with h. The cost is observed to be decreasing initially with increasing h and tends to increase again. The increase in cost corresponds to a stage where more number of differently colored edges lie in the clusters formed.

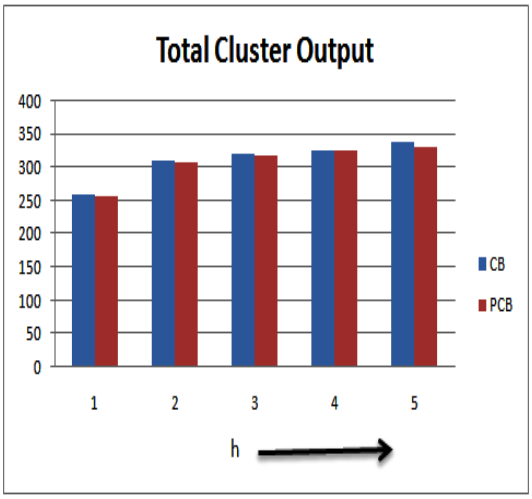

Fig.13. Growth of total cluster output with h

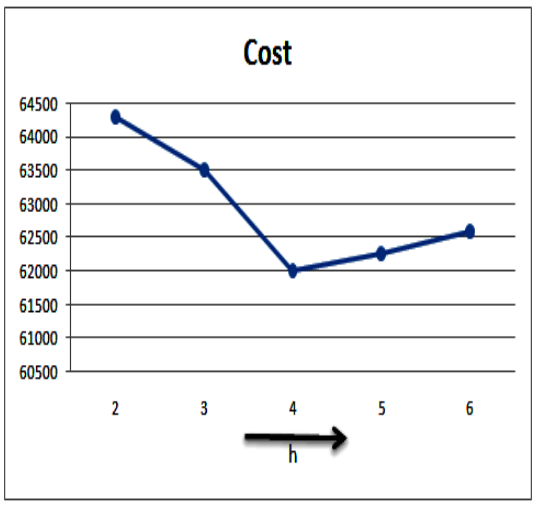

Fig.14. Growth of runtime of PCB with h

D. Effect of Inter-Cluster Edges

For the last set of experiments the parameter q is varied from 0.05 to 0.25. The parameter q denotes the probability of sampling of inter-cluster edges. Increasing value of q tends to decrease the performance of the algorithms. With more inter-cluster edges, the cost increases manifold. The attributes taken here for measuring the effect of q are the number of isolated points and cost of the PCB and CB algorithms. Table 6 lists the results of the experiments at values of n, K, h, p, w and v equal to 1000, 20, 4, 0.6, 0.5 and 0.7 respectively.

Table 6. Effect of the Probability of Inter Cluster Edges on Total Clusters and Cost of the Algorithms

	No. of isolated points		Cost	
q	CB	PCB	CB	PCB
0.05	12±4	9±2	64094±91	63840±24
0.1	8±2	5±3	84792±119	84800±106
0.15	5±2	4±3	106180±106	106207±62
0.2	3±2	3±1	127646±25	127743±84
0.25	3±2	2±1	148821±137	149021±77

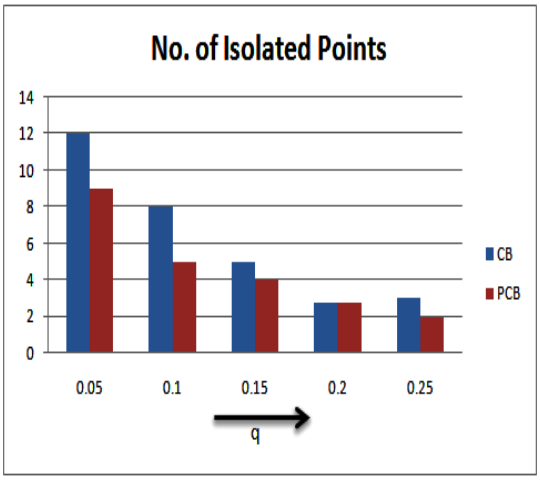

Fig.15. Growth of no. of isolated points with q

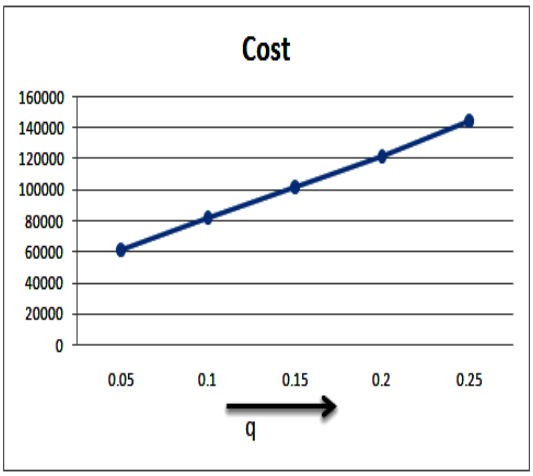

Fig.16. Growth of Cost of PCB with q

Figure 15 and figure 16 show the growth in number of isolated points for both algorithms and cost of proposed algorithm with q. The number of isolated points and cost of the proposed PCB algorithm are observed to be

slightly less as compared to the results obtained for CB. This linear growth of cost with variations in q implies that proposed PCB is robust enough even when many interconnections among ground truth clusters exist. This is also supported by lesser number of singleton clusters produced. It is an indication that proposed PCB is capable of associating more points together into groups.

VII. Conclusion and Future Scope

Data with categorical pair wise relations are increasingly becoming common. Some popular examples are users of social sites who have different relations like friends, classmates, clubmates, relatives etc, protein-protein intreaction network, bibliographic data, and many more such data network. The problem of clustering objects in a data network based on categorical similarity is called Chromatic Correlation Clustering. A simple heuristic called Chromatic Balls proposed by Bonchi et al. attempts to solve to problem of chromatic correlation clustering in a fast manner. But it fails to perform in certain situations where networks of many vertices are dense enough in each category of relations. This paper proposes an improvement in the chromatic balls (CB) algorithm to improve the chances of obtaining a near-to-optimal clustering. The proposed algorithm is named Priority Based Chromatic Balls (PCB). Comparison of CB and PCB is presented through outputs generated on synthetic data. The synthetic data is generated with varying density and size to demonstrate that PCB performs better than CB in each case.

Research in field of data networks and clustering these structures is very new. It has must scope of variation in the data models itself. Newer models would then require corresponding changes in the algorithms. Extending the existing concept of CB to "fuzzy" version of the problem by having overlapping clusters is an open problem. Involving more than one CB approach into a clustering ensemble can be an interesting approach. Exploring situations where the proposed PCB algorithm has more chances of poor output is an open problem.

Acknowledgment

I would like to express my deep sense of respect and gratitude towards all the faculty members, Department of Computer Science & Engineering in R. N. Modi Engineering College Kota, and thanks to each person who has been the guiding force behind this work. Without their unconditional support it wouldn't have been possible.

References

[1] E.W. Forgy, "Cluster Analysis of Multivariate Data: Efficiency v/s Interpretability of Classification", Biometrics, 21, 768-769, 1965.

[2] J. C. Bezdek , R. Ehrlich and W. Full, "FCM: The Fuzzy C-Means Clustering Algorithm" , Computers & Geosciences, vol. 10, No. 2-3, pp. 191 -203, 1984.

[3] N. Bansal, A. Blum and S. Chawla, "Correlation Clustering ", Machine Learning, Vol. 56, Pp. 89-113, 2004.

[4] F. Bonchi, A Gionis , F. Gullo and Ukkonen, " Chromatic Correlation Clustering ", Proceedings of The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12) ,pp. 1321-1329,2012.

[5] F. Bonchi, A. Gionis, F. Gullo Charalampos, E. Tsourakakis And A. Ukkonen,"Chromatic Correction Clustering", ACM Transactions on Knowledge Discovery from Data (TKDD),Vol.9 Issue 4 ,No. 34, 2015.

[6] Aaditya Jain, Dr. Bala Buksh, "Advancement in Clustering with the Concept of Correlation Clustering-A Survey", International Journal of Engineering Development and Research, Vol. 4, Issue 2, 2016.

[7] M. Rice and V. J. Tsotras, "Graph indexing of road networks for shortest path queries with label restrictions", Proceedings of the VLDB Endowment, Vol. 4, No. 2, pp. 69-80, 2010.

[8] M. J. Kearns, R. E. Schapire, and L. M. Sellie, "Toward Efficient Agnostic Learning ", Machine Learning , Vol. 17, No. 2-3, pp. 115-142, 1994.

[9] I. Giotis and V. Guruswami, "Correlation Clustering with a Fixed Number of Clusters", Theory Of Computing, Vol. 2, pp. 249–266, 2006.

[10] N. Ailon and E. Liberty, "Correlation Clustering Revisited: The "True" Cost of Error Minimization Problems", Proceedings of the 36th International Colloquium on Automata, Languages and Programming: Part I (ICALP '09), pp. 24-36, 2009.

[11] Yi Gu and Chaoli Wang, "A Study of Hierarchical Correlation Clustering for Scientific Volume Data", Advances in Visual Computing, Volume 6455 of the series Lecture Notes in Computer Science, pp 437-446, 2010.

[12] K. Makarychev, Y. Makarychev and A. Vijayaraghavan, "Correlation Clustering with Noisy Partial Information" in JMLR: Workshop and Conference Proceedings, vol 40, pp.1–22, 2015.

[13] Kookjin Ahn, Graham Cormode, Sudipto Guha, Andrew Mcgregor and Anthony Wirth, "Correlation Clustering in Data Streams" in Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pp:2237-2246, 2015.

Authors' Profiles

Aaditya Jain is currently pursuing M.Tech in Computer Science & Engineering from R. N. Modi Engineering College Kota which is affiliated to Rajasthan Technical University, Kota (Raj). He received B.E. degree in Computer Science & Engineering from Mandsaur Institute of Technology Mandsaur which is affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (MP) in 2013. He is the author of many scientific publications in International and National Conferences and Journals. His two papers has awarded by "Best Paper Award" in international conferences. His areas of interests are Correlation Clustering, Network Security, Cryptography, Mobile Computing, Internet of Things, Ad Hoc Networks, and Wireless Sensor Networks.

Dr. Suchita Tyagi is currently working as Associate Professor in Department of Computer Science & Engg. in Sushila Devi Bansal College of Technology, Indore. She has done her Ph.D in Computer Science from Dr. A.P.J. Abdul Kalam Technical

University, Lucknow, UP. She has 14 year of academic experience. She has published many high quality research papers in International and National Journals and Conferences. She is life member of CSI community. Her research interests are Clustering techniques and methodology, Classifications related problems, Wireless networks and Named data networking.

How to cite this paper: Aaditya Jain, Suchita Tyagi,"Priority Based New Approach for Correlation Clustering", International Journal of Information Technology and Computer Science (IJITCS), Vol.9, No.3, pp.71-79, 2017. DOI: 10.5815/ijitcs.2017.03.08

