IJCNN 2019. International Joint Conference on Neural Networks. Budapest, Hungary. 14-19 July 2019

Qualitative data clustering: a new Integer Linear Programming model

Luiz Henrique Nogueira Lorena
Science and Technology Institute
Federal University of São Paulo
São José dos Campos, Brazil
luiz-lorena@hotmail.com

Marcos Gonçalves Quiles
Science and Technology Institute
Federal University of São Paulo
São José dos Campos, Brazil
quiles@unifesp.br

Luiz Antonio Nogueira Lorena
Computing and Applied Mathematics
National Institute for Space Research
São José dos Campos, Brazil
luizlorena54@gmail.com

André C. P. L. F. de Carvalho
Computer Science Institute
University of São Paulo
São Carlos, Brazil
andre@icmc.usp.br

Juliana Garcia Cespedes
Science and Technology Institute
Federal University of São Paulo
São José dos Campos, Brazil
jcespedes@unifesp.br

***Abstract*—Qualitative data clustering is a fundamental data analysis task, with applications in many areas, like medicine, sociology, and economics. An appealing way to deal with this task is via Integer Linear Programming, as it avoids inappropriate inferences by the final user. This approach has two main advantages: the data are directly used, without the need of being converted to quantitative values, and the optimal number of clusters is automatically obtained by solving the optimization problem. However, it might create large and redundant models, which can limit the size of the problems it can be applied. Recently, models that are more compact and able to avoid some redundancy have been proposed in the literature. These models consume less memory and are faster to obtain the optimal solution set. In this study, a new model is introduced and compared with the state-of-the-art alternatives using datasets from different application domains. Empirical results show that the new model outperforms its predecessors, achieving the optimal solution set with lower computational time and memory consumption.**

I. Introduction

Data clustering is fundamental for many data analysis applications. It is frequently used to learn and understand the relationship between entities. Due to the importance of data clustering, several clustering techniques have been proposed in the literature [1]. Formally, a clustering technique distributes a set of n entities, each characterized by m attributes, into disjoint subsets (or clusters) as homogeneous as possible, regarding the data attributes. In general, the clustering process is performed in two steps. First, a similarity measure is selected to distinguish the entities. Next, the entities are grouped into clusters according to their similarities [2].

Qualitative data clustering works with data composed only of qualitative attributes. Qualitative data are common in different knowledge domains, like medicine [3], sociology [4] and economics [5], where qualitative attributes, like "blood type", "gender" and "industrial sector", respectively, are frequently employed. In spite of a large number of datasets with qualitative attributes, there are few algorithms specifically tailored to cluster qualitative data [6].

This work focus on an Integer Linear Programming (ILP), which formulates this clustering problem as a Clique Partitioning Problem (CPP) [7]. This approach has some advantages when compared with related techniques. First, the data are directly used, avoiding inappropriate conversions from qualitative to quantitative data. Another benefit is the automatic identification of the optimal number of clusters, avoiding another improper inference of the specialist. Finally, Wang et al. [8] reported that this approach provides superior cluster structure recovery when compared to heuristic approaches, like the latent class cluster analysis [9] and the K-means [10] techniques.

The standard ILP model for CPP, however, may contain a large number of redundant constraints. If removed, redundant constraints do not change the optimal solution set. Nonetheless, they limit the size of the problems that can be solved. Some strategies were proposed to eliminate redundant constraints, like the cutting plane heuristic proposed by Grötschel et al. [7] and the development of more compact models by Dinh et al. [11] and Miyauchi et al. [12]. In this paper, the authors extend the results from [12], presenting a new CPP model. Experiments reported next show the good results obtained by the proposed model for datasets from different application domains.

This work is organized as follows. In Section II, the problem is defined, and the standard mathematical programming formulation is presented. Section III introduces the proposed model. The computational experiments are reported in Section IV. Finally, conclusions are discussed in Section V, along with future work directions.

978-1-7281-2009-6/$31.00 paper N-19227.pdf

II. Qualitative Data Clustering via ILP

Grötschel et al. [7] provided an Integer Linear Programming model based on the Clique Partitioning Problem (CPP) for qualitative data clustering. The CPP model this problem as a complete weighted graph where each entity of the dataset is represented by a vertex and the edges weight represents the similarity between entities. The goal is to partition the graph into complete subgraphs that have the highest similarity value among its members. To illustrate this approach, consider a dataset with n entities and m qualitative attributes (Table I).

TABLE I: Example Dataset.

Id	Age	Prescription	Astigmatic	Tear
1	young	hypermetrope	yes	reduced
2	young	hypermetrope	yes	normal
3	pre-presbyopic	hypermetrope	yes	?
4	pre-presbyopic	hypermetrope	no	normal
5	presbyopic	myope	no	reduced
6	presbyopic	myope	yes	reduced

The similarity between entities i and j is calculated as:

$$s_{ij} = 2\sum_{k=1}^{m} r_{ij}^{k} - (m - I_{ij}) \tag{1}$$

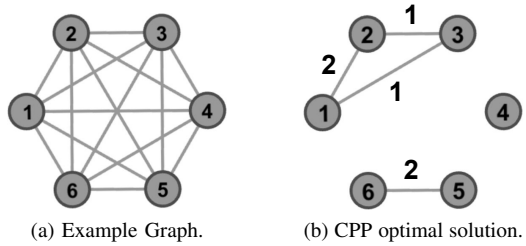
where

$$r_{ij}^{k} = \begin{cases} 1 & \text{attribute k is the same in i and j,} \\ 0 & \text{otherwise} \end{cases}$$

and I_{ij} is the total of incomplete attributes. The similarity matrix S is obtained after applying Equation 1 to the data from Table I.

$$\begin{pmatrix} & 2 & 1 & -2 & -2 & 0 \\ 2 & & 1 & 0 & -4 & -2 \\ 1 & 1 & & 1 & -3 & -1 \\ -2 & 0 & 1 & & -2 & -4 \\ -2 & -4 & -3 & -2 & & 2 \\ 0 & -2 & -1 & -4 & 2 & \end{pmatrix}$$

Figure 1a shows the problem presented in Table I as a complete weighted graph. Figure 1b represents the optimal solution obtained by the CPP.



(a) Example Graph. (b) CPP optimal solution.

Fig. 1: Example graph and the optimal solution obtained by CPP.

Based on the similarity matrix S, Grötschel et al. [7] provided the following ILP model, defined here as GM, to solve this problem:

$$\text{(GM):} \quad \text{Maximize} \quad \sum_{i<j} s_{ij} x_{ij}$$

$$\begin{aligned} \text{subject to} \quad x_{ij} + x_{jk} - x_{ik} &\leq 1 \quad i<j<k \quad (2) \\ x_{ij} - x_{jk} + x_{ik} &\leq 1 \quad i<j<k \quad (3) \\ -x_{ij} + x_{jk} + x_{ik} &\leq 1 \quad i<j<k \quad (4) \\ x_{ij} &\in \{0,1\} \quad i,j \in [1..n] \end{aligned}$$

where

$$x_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are in the same group} \\ 0 & \text{otherwise.} \end{cases}$$

Constraints (2-4) are called "transitivity constraints". They enforce that: if vertex i is in the same cluster as vertex j, and j is in the same cluster as vertex k then i is in the same cluster as k. The GM model has a total of $O(n^3)$ transitivity constraints. This huge number of constraints may prevent the use of this model on larger instances since it demands more computational resource as n grows.

Grötschel et al. [7], however, experimentally discovered that a large number of the transitivity constraints are redundant. Only a subset of then was needed by their cutting plane algorithm [13] to achieve the problem optimal solution. This fact justified the development of new models that could avoid such redundancy.

Recently, Miyauchi et al. [12] provided two ILP models that are more compact and faster than the standard GM model. Although these models are able to reduce considerably the number of redundant constraints, there is still room for improvement. Hence, a new model is proposed in this paper. It addresses the problems encountered in previous proposals while achieving superior redundancy elimination.

In the next section the models proposed by [12], denoted here as GM1 and GM2, are revisited highlighting some of its shortcomings. Next, the new model is introduced, exposing some particular aspects of the qualitative data clustering problem that were considered to create a more compact ILP model.

III. Proposal of a new CPP model

To better understand the new model proposed in this work, consider a complete undirected data graph denoted by $G = (V, E)$, where V and E are, respectively, sets of vertices and edges. An edge is denoted by $\{i, j\}$ $(i, j \in V)$ and its weight value is represented by the similarity (s_{ij}).

Let $x' = (x_{ij})$ be a feasible solution of model GM, and

$$E' = \{\{i,j\} \in E \mid x_{ij} = 1\}$$

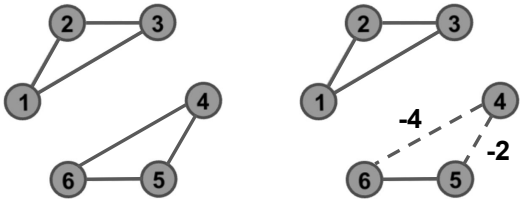
$$C = \{(V_1, E'_1), (V_2, E'_2), ..., (V_p, E'_p)\}$$

where E' is the set of edges present inside the clusters of the feasible solution and C is the set of the corresponding p clusters.

Miyauchi et al. [12] proposed new ILP models that have less redundant constraints than the GM model. The authors analyzed the groups obtained in the optimal partition and defined a sufficient condition that its edge weights should follow to preserve group structure. A constraint will be included in those models only if it respects such condition, otherwise, it is considered as redundant.

The sufficient condition states that if any cluster C in the optimal solution is partitioned in two clusters $\{A, B\}$, the set of edges between clusters (E_{AB}) must contain at least one non-negative edge weight. Otherwise, the cluster structure cannot be preserved.

An example illustrating the sufficient condition is given in Figure 2. To explain it, suppose a feasible partition (Figure 2a). If the cluster composed of vertices $\{4, 5, 6\}$ is partitioned in two ($A = \{4\}$ and $B = \{5, 6\}$), the set E_{AB} will contain only negative edge weights (Figure 2b). Hence, there is an improvement in the objective function of the problem if the vertex 4 is set as a new cluster. Cluster $\{4, 5, 6\}$ from Figure 2a is feasible but will never occur in the optimal solution. Constraints that consider such invalid state can be considered as redundant.



(a) Feasible Partition. (b) Negative edge weights in E_{AB}.

Fig. 2: Negative edge weights in the set E_{AB}.

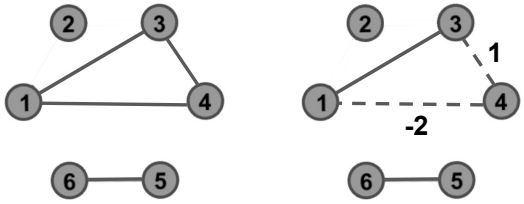
The first model proposed by [12], denoted here as GM1, was created to prevent the inclusion of constraints that fails to respect the sufficient condition they have defined. The GM model is modified to include the following conditional clauses for each transitivity constraint:

$$GM1 \begin{cases} (2), & s_{ij} \geq 0 \vee s_{jk} \geq 0 \\ (3), & s_{ij} \geq 0 \vee s_{ik} \geq 0 \\ (4), & s_{jk} \geq 0 \vee s_{ik} \geq 0 \end{cases}$$

GM1 tests each constraint during the ILP model construction checking if there is at least one non-negative edge weight in the set E_{AB}. Constraints that do not respect these clauses are not included in the model.

As can be observed in the section of experimental results, GM1 reduces the number of redundant constraints and consequently improves the computational times for the ILP solver. However, this improvement is still modest, justifying the search for alternative models.

There are conditions that GM1 is unable to prevent. Suppose the feasible partition presented in Figure 3a. If the cluster composed of vertices $\{1, 3, 4\}$ is partitioned in two, clusters $A = \{1, 3\}$ and $B = \{4\}$ will be obtained. The set E_{AB} respect the condition that it must contain at least one non-negative edge weight, however, there is an improvement in the objective function of the problem if the vertex 4 is set as a new cluster. This occurs because the sum of the weight of the edges in the set E_{AB} is negative. Constraints that consider such invalid state can be considered as redundant.



(a) Feasible Partition. (b) Negative edge weight sum.

Fig. 3: Negative edge weight sum in the set E_{AB}.

A second model, denoted here as GM2, was derived by [12] to prevent a negative value for the sum of the weight of the edges in E_{AB}. The GM model is modified to include the following conditional clauses for each transitivity constraint.

$$GM2 \begin{cases} (2), & s_{ij} + s_{jk} \geq 0 \\ (3), & s_{ij} + s_{ik} \geq 0 \\ (4), & s_{jk} + s_{ik} \geq 0 \end{cases}$$

GM2 tests each constraint during the ILP model construction checking if the sum of the weight of the edges in E_{AB} is non-negative. Constraints that do not respect these clauses are not included in the model. The number of redundant transitivity constraints is reduced by controlling the role of negative edge weights in the set E_{AB}. This provides better results when compared to GM1 (see the section of experimental results).

Although the GM1 and GM2 models provide superior results over the GM model, there is still room left for improvement in terms of computational time and memory consumption. These models focused solely on the role of edges in set E_{AB}. Hence, here we expand the analysis of which transitivity constraints should be included to preserve cluster structure in the context of the optimal solution.

To derive a new model, let's first examine the edge weight distribution within triangles considered by the transitivity constraints of the GM model (Figure 4). It can be observed that all triangles in T4 and some triangles in T1 are extreme cases in which there is no doubt about the decision on keeping the vertices together or apart.

It is not necessary to create constraints that check for T1 triangles with strictly positive edges since there is a consensus that the vertices should be together. The maximization model will try to set the corresponding variables to 1. In the same way, T4 triangles should be disregarded by the constraints during optimization, since there is a consensus that the vertices should be separated. Consequently, transitivity constraints should be included only for the triangles T1 (not strictly

positive weights), T2 and T3, where such consensus is not present.

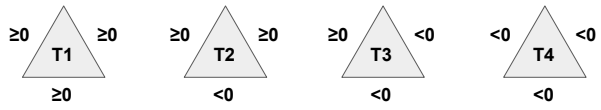


Fig. 4: Possible edge weight distribuitions.

The proposed model, named here GM3, can eliminate transitivity constraints corresponding to extreme values of triangles T1 and T4 and control the distribution of the negative edge weights within triangles T2 and T3.

$$GM3 \begin{cases} (2), & (s_{ij} + s_{jk} \geq 0) \wedge (s_{ij} \geq 0) \wedge (s_{ik} \leq 0) \\ (3), & (s_{ij} + s_{ik} \geq 0) \wedge (s_{ij} \geq 0) \wedge (s_{jk} \leq 0) \\ (4), & (s_{ik} + s_{jk} \geq 0) \wedge (s_{ik} \geq 0) \wedge (s_{ij} \leq 0) \end{cases}$$

The conditional clauses of GM3 respect the sufficient conditions presented previously. When applied to constraint (2), the conditional clause assures that at least one of the edges in E_{AB} are non-negative by fixing $(s_{ij} \geq 0)$ and states that the sum of the edge weights in the set E_{AB} is non-negative by using $(s_{ij} + s_{jk} \geq 0)$. The conditional $(s_{ik} \leq 0)$ ensures that strictly positive triangles are avoided.

In the context of the triangles of Figure 4, GM3 avoids the creation of transitivity constraints that checks for triangles T4 by using $(s_{ij} \geq 0)$ and strictly positive triangles T1 using $(s_{ik} \leq 0)$. The clause $(s_{ij} + s_{jk} \geq 0)$ controls the negative edge weights in the context of triangles T2 and T3.

In the next section, a set of experiments were conducted to compare the models presented in this section according to redundancy elimination, computational time and memory consumption.

IV. Experimental Results

The experiments and algorithms were coded in C++14 and executed on a computer with the following configuration: Intel Core i7-6770HQ (3,5GHz) with 32 GB RAM running Windows 10 64-Bit. The commercial solver IBM ILOG CPLEX [14] 12.7.1 was used to solve the ILP models, and the R [15] language was employed in the statistical analysis. The source code is available online at *https://github.com/LuizHNLorena/QualitativeClustering/*.

Figure 5 presents the methodology used to conduct computational experiments. First, at step 1 and 2, the datasets were standardized by removing their class attribute (if provided) and by using the symbol "?" to represent missing attribute values. At step 3, four different ILP models for each dataset were created: GM is the ILP model composed of all the constraints; GM1 and GM2 are the models proposed by Miyauchi et al. [12], and GM3 is the model proposed in this work. At step 4, the ILP models were solved by the Integer Linear Programming solver. Finally, the obtained results were analyzed in step 5 using statistical tests, which verified whether there are significant differences between the performance of each model.

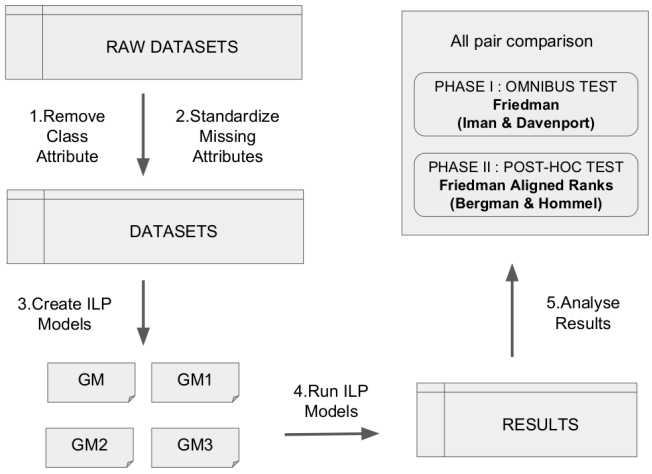


Fig. 5: Methodology used to conduct the computational experiments.

The ILP models were evaluated using various real-world problem datasets (Table II). Each dataset received a unique id (first column) and a name (second column). The remaining columns in this table have the following meanings: n is the number of objects; m is the number of attributes; $Incomplete$ informs if the dataset has objects with missing attributes; and, finally, $\%Pos$ reveals the proportion of edges with positive weights values.

The datasets whose ID belongs to the set {2-7,9,13,14,17,18} are classical instances of the CPP proposed in the literature. These instances can be found in the

TABLE II: Real-world datasets used in the experiments.

ID	Dataset	n	m	Incomplete	%Pos
1	Lenses	24	4	No	60.87
2	Wildcats	30	14	No	62.76
3	Lung Cancer	32	56	Yes	69.15
4	Cars	33	13	No	72.73
5	Workers	34	13	No	58.82
6	Cetacea	36	16	Yes	27.94
7	Micro	40	14	No	28.59
8	Soybean (Small)	47	35	No	99.17
9	UNO	54	3	No	40.81
10	Sponge	76	45	Yes	65.33
11	Zoo	101	16	No	71.23
12	Bridges	108	13	Yes	47.87
13	UNO1b	139	3	No	47.72
14	UNO2b	145	15	No	75.64
15	Lymphography	148	18	No	67.46
16	Teaching Evaluation	151	5	No	8.12
17	UNO1a	158	3	No	39.73
18	UNO2a	158	15	No	64.45
19	Hayes-Roth	160	3	No	24.01
20	Primary Tumor	339	17	Yes	93.87

Appendix of [7]. Datasets whose ID belongs to the set {1,3,8,10-12,15,16,19,20} represent instances of problems from different knowledge domains selected from UCI Machine Learning Repository [16]. All datasets are composed only of categorical attributes.

The following subsections detail the results obtained after applying the steps 1-4 of the methodology presented in Figure 5. Each model is compared in terms of constraint elimination, computational time, space efficiency and robustness.

A. *Constraint elimination and computational time*

Table III provides detailed results regarding the number of constraints and computational time obtained by each model. The first column makes reference to the dataset ID from Table II, the objective function value is presented in column Obj, columns $\#Constraints$ and $Time$ represents the total of constraints of the model and the computational time it takes to solve it, respectively. The best results are highlighted in bold.

It can be observed, from Table III, that all models achieve the optimal objective value. However, the performance of GM3 was superior to the others in the context of constraint elimination and computational time.

Figure 6 complements the results presented in Table III. It summarizes the results related to the percentage of constraints elimination. Model GM2 performs better than GM1, but the boxplot confirms the improvement obtained by GM3, 100% of the distribution is above GM1 and GM2 median value. According to its lower quartile, 75% of the instances obtained a percentage of constraint elimination above 80%.

Figure 7 presents the speedup obtained by each model. The graph horizontal axis is in log scale. It can be concluded that GM2 performs better than GM1, but GM3 achieved better results than its competitors. Its lower quartile shows that 75% of the instances obtained a speedup above 5.

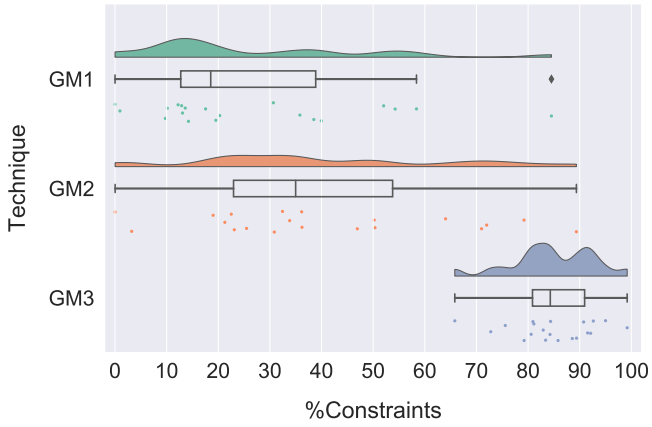


Fig. 6: Percentage of constraints elimination.

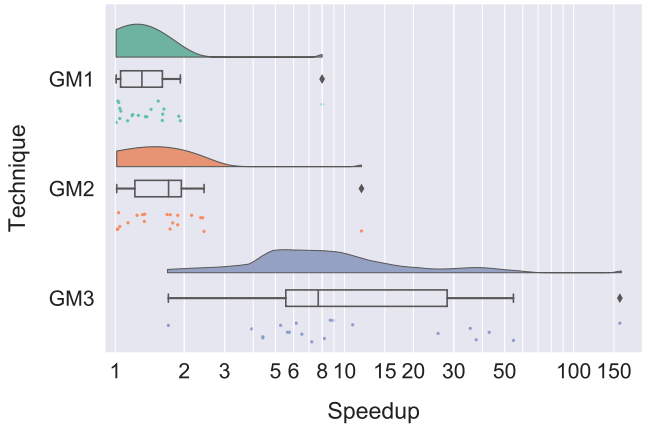


Fig. 7: Speedup obtained over GM model.

TABLE III: Experimental Results.

ID	Objective	#Constraints				Time			
		GM	GM1	GM2	GM3	GM	GM1	GM2	GM3
1	72	6072	5208	3024	**2076**	0.387	0.370	0.289	**0.227**
2	1304	12180	10043	8060	**1905**	0.163	0.144	0.131	**0.025**
3	3472	14880	12959	11089	**2809**	0.315	0.301	0.277	**0.029**
4	1501	16368	14708	13257	**2324**	0.425	0.414	0.406	**0.059**
5	964	17952	14444	12413	**3422**	0.433	0.366	0.329	**0.053**
6	967	21420	9798	6214	**1980**	0.138	0.102	0.082	**0.024**
7	406	29640	14217	6161	**3147**	0.207	0.127	0.086	**0.047**
8	14613	48645	48638	48630	**395**	2.865	2.709	2.791	**0.018**
9	798	74412	45756	36945	**12703**	1.939	1.209	1.116	**0.217**
10	25677	210900	182196	162169	**43798**	2.959	2.347	2.313	**0.596**
11	16948	499950	451130	387355	**136133**	106.897	56.505	45.335	**20.300**
12	3867	612468	393126	220328	**96487**	305.371	294.935	125.122	**54.120**
13	11775	1313967	910908	838054	**254913**	57.562	35.733	33.199	**2.246**
14	71818	1492920	1310497	1175303	**117649**	34.119	28.599	25.368	**0.797**
15	19174	1588188	1380477	1072877	**387728**	4359.761	3043.938	2327.782	**989.907**
16	1108	1687425	261201	179647	**84337**	23.071	2.888	1.944	**0.614**
17	12197	1934868	1161623	1026713	**321261**	99.061	51.521	46.043	**2.798**
18	72820	1934868	1542583	1235377	**142490**	53.037	38.628	29.776	**0.971**
19	2800	2009760	836313	563121	**229952**	553.723	359.689	295.675	**89.836**
20	323614	19307067	19121209	18680264	**1634216**	612.824	607.379	592.644	**70.632**

TABLE IV: Memory Consumption in megabytes for each model.

ID	GM	GM1	GM2	GM3
1	4.8	4.2	3.0	**2.5**
2	8.5	7.1	6.2	**2.5**
3	9.8	8.8	8.0	**3.0**
4	11.2	9.7	9.1	**2.7**
5	12.2	9.7	8.6	**3.4**
6	13.7	7.1	5.0	**2.6**
7	18.6	9.7	5.1	**3.3**
8	30.4	30.4	30.4	**1.9**
9	46.1	29.3	23.7	**9.0**
10	123.1	111.5	95.3	**29.2**
11	282.2	262.5	229.6	**85.7**
12	357.0	232.2	131.7	**61.0**
13	746.8	525.9	482.3	**159.7**
14	884.1	745.8	690.9	**73.4**
15	923.0	838.7	649.6	**232.3**
16	963.4	163.0	114.1	**54.5**
17	1092.0	686.2	573.9	**189.8**
18	1092.1	905.0	716.1	**91.4**
19	1122.5	712.9	277.3	**192.9**
20	11174.4	11098.7	10920.3	**961.9**

B. Space Efficiency

Table IV shows the amount of memory in megabytes required to store each model. It can be observed that GM1 and GM2 reduce memory consumption when compared to GM, but GM3 clearly outperforms both. For dataset 20, for instance, it requires 960 MB of memory, while the remaining models require 10 GB or more.

Figure 8 summarizes the results presented in Table IV. The amount of memory consumption reduction is calculated for each model using the GM model as a base. The graph shows that GM1 and GM2 are able to reduce memory consumption, but GM3 outperforms its competitors. Its lower quartile shows that it obtained above 75% of memory consumption reduction for 75% of the instances.

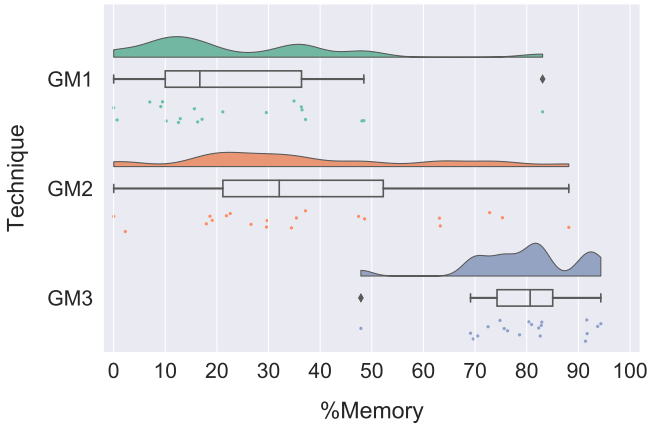


Fig. 8: Percentage of memory consumption reduction over GM model.

C. Robustness

The authors in [12] mentioned that the model GM1 would work better on graphs with a small proportion of positive edge weights. This behavior can be observed in the graph of Figure 9. The bottom horizontal axis of this graph lists the datasets sorted in ascending order of proportion of positive edge weight values ($\%Pos$ of Table II), while the top horizontal axis lists the dataset's ID (ID of Table II). The vertical axis represents the percentage of constraint elimination obtained by each model regarding the GM model.

Figure 9 shows that GM1 is *de facto* affected by the proportion of positive edge weights. Hence, the Spearman's rank correlation coefficient [17] (ρ) was used to verify this influence on each model. The Spearman correlation coefficient value is $\rho = -0.97$, showing a strong negative linear correlation between the variables $\%Constraints$ and $\%Pos$.

The GM2 model has the same shortcoming of GM1. It is affected by the increase of the positive edge weights proportion, but, on average, it obtained better results. Its correlation coefficient is $\rho = -0.95$, which represents a strong negative linear correlation between the variables $\%Constraints$ and $\%Pos$.

The proposed model (GM3) presented some fluctuations, but was not directly affected by the proportion of positive edge weights (Figure 9). It obtained a correlation coefficient equal to -0.058, which represents a weak negative linear correlation association between the analyzed variables. This model presented the best overall performance. It obtained its worst performance on Dataset 1 (65.81% of constraint elimination), but this result is superior to GM1 and GM2, which obtained 14.23% and 50.19% of constraint elimination.

D. Statistical significance

Finally, following the step 5 from the methodology proposed in Figure 5, an all pair comparison test was executed in order to ensure that there is a significant difference between the performance of each model. This work followed the recommendations from Demsar [18] and Garcia et al. [19], [20] to compare the percentage of constraint elimination ($\%Constraints$).

The Friedman test with Iman and Davenport [21] extension was applied as *omnibus* test to detect if at least one of the models performs differently from the others. Friedman's Aligned Rank Test [22] allied with Bergman and Hommel [23] method to correct the obtained *p-values*, was employed as *post-hoc* test. The significance level of $\alpha = 0.05$ was considered. This work used the R [15] package provided by Calvo et al. [24], which contains the *omnibus* and *post-hoc* tests, to conduct the hypothesis tests.

The *omnibus* test obtained a *p-value* equals to $2.2e{-}16$, showing that at least one of the models performed differently. The *post-hoc* showed that all the models performed differently and that GM3 outperformed all the other models.

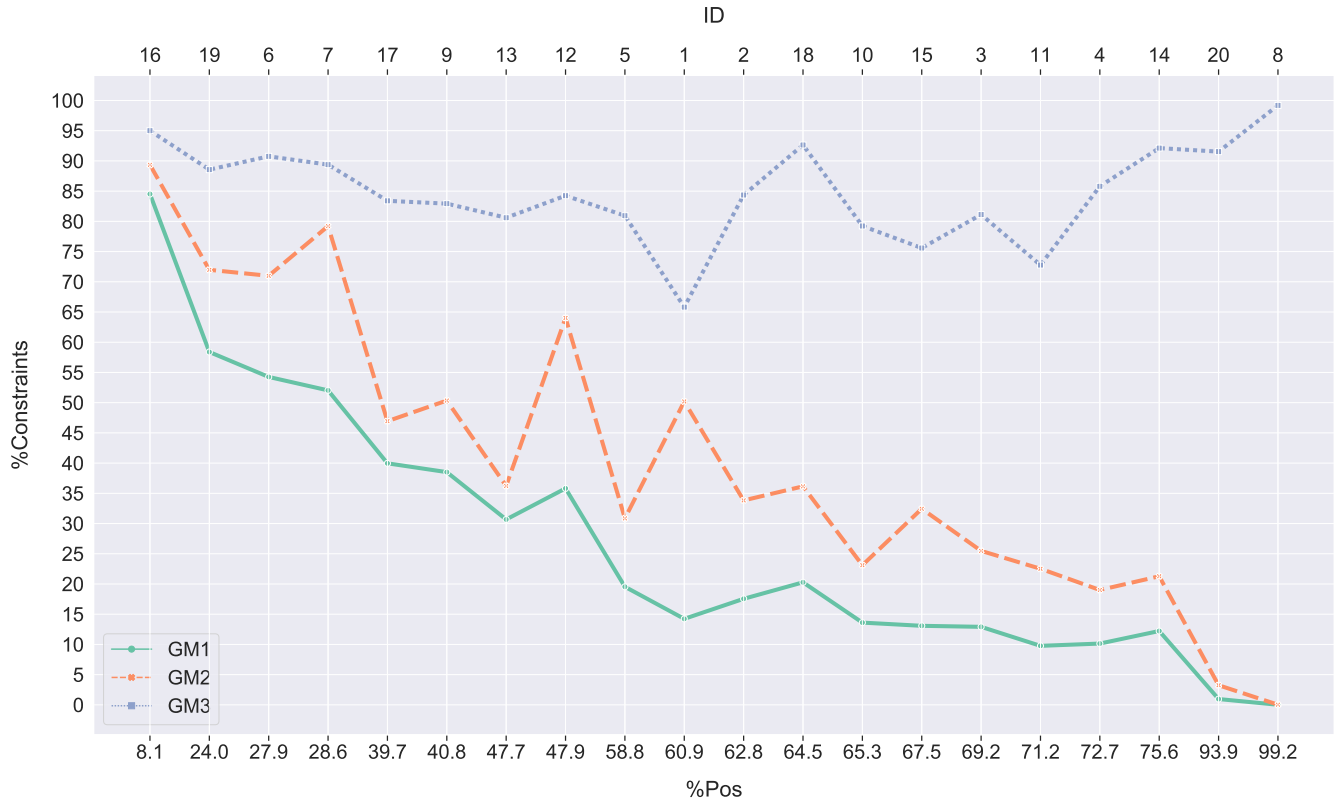


Fig. 9: Influence of the proportion of positive edge weights in the performance of each model.

V. Conclusions

Qualitative data clustering via integer linear programming has its advantages. But one of its shortcomings is the number of redundant constraints in the traditional ILP model. Recently, more compact models were created [12]. Although these models are able to reduce the number of redundant constraints considerably, there is still room for improvement.

In this work, a new model, named GM3, was proposed. Experimental results showed that it achieved superior performance over its counterparts. It required less computational time and memory to achieve the optimal solution. The experiments also showed its robustness. GM3 performed well on datasets with different properties, avoiding the shortcomings of previous models.

As future work, the authors will study how feature selection may impact the performance of the new model. The similarity matrix depends on the number of features of the dataset and any reduction of features will change its values. This new configuration may be beneficial for solving more computationally costly problems, such as the instance 15 used in the experiments.

The results obtained in the mathematical approach context can be used to guide the construction of better heuristics for this problem. The authors also expect to use model GM3 in other contexts, like clustering of microarray data [25], community detection [11], [26] Group Technology Problem [27] and Flight-gate scheduling [28].

Acknowledgment

The authors would like to thank the FAPESP (Grant No. 2011/18496-7 and 2013/07375-0), CNPq (GrantNo. 310908/2015-9 and 301836/2014-0), CAPES and IBM for support.

References

[1] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[2] C. C. Aggarwal and C. K. Reddy, *Data clustering: algorithms and applications*. CRC Press, 2013.

[3] Z. Chen, S.-H. Yang, H. Xu, and J.-J. Li, "Abo blood group system and the coronary artery disease: an updated systematic review and meta-analysis," *Scientific reports*, vol. 6, 2016.

[4] C. Goeke, S. Kornpetpanee, M. Köster, A. B. Fernández-Revelles, K. Gramann, and P. König, "Cultural background shapes spatial reference frame proclivity," *Scientific reports*, vol. 5, 2015.

[5] C. Li, L. Sun, J. Jia, Y. Cai, and X. Wang, "Risk assessment of water pollution sources based on an integrated k-means clustering and set pair analysis method in the region of shiyan, china," *Science of The Total Environment*, vol. 557, pp. 307–316, 2016.

[6] S. Cherevko and A. Malikov, "Review of modern techniques of qualitative data clustering," in *Young Scientists International Workshop on Trends in Information Processing (YSIP)*, 2014, p. 7.

[7] M. Grötschel and Y. Wakabayashi, "A cutting plane algorithm for a clustering problem," *Mathematical Programming*, vol. 45, no. 1-3, pp. 59–96, 1989.

[8] H. Wang, T. Obremski, B. Alidaee, and G. Kochenberger, "Clique partitioning for clustering: a comparison with k-means and latent class analysis," *Communications in Statistics-Simulation and Computation*, vol. 37, no. 1, pp. 1–13, 2007.

[9] J. K. Vermunt and J. Magidson, "Latent class cluster analysis," *Applied latent class analysis*, vol. 11, pp. 89–106, 2002.

[10] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.
[11] T. N. Dinh and M. T. Thai, "Toward optimal community detection: From trees to general weighted networks," *Internet Mathematics*, vol. 11, no. 3, pp. 181–200, 2015.
[12] A. Miyauchi and N. Sukegawa, "Redundant constraints in the standard formulation for the clique partitioning problem," *Optimization Letters*, vol. 9, no. 1, pp. 199–207, 2015.
[13] H. Marchand, A. Martin, R. Weismantel, and L. Wolsey, "Cutting planes in integer and mixed integer programming," *Discrete Applied Mathematics*, vol. 123, no. 1-3, pp. 397–446, 2002.
[14] IBM, "Ibm ilog cplex 12.7.1," 1987-2017.
[15] R. D. C. Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2008, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org
[16] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml
[17] C. Spearman, "The proof and measurement of association between two things," *The American journal of psychology*, vol. 15, no. 1, pp. 72–101, 1904.
[18] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.
[19] S. Garcia and F. Herrera, "An extension on"statistical comparisons of classifiers over multiple data sets"for all pairwise comparisons," *Journal of Machine Learning Research*, vol. 9, no. Dec, pp. 2677–2694, 2008.
[20] S. García, A. Fernández, J. Luengo, and F. Herrera, "Advanced non-parametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power," *Information Sciences*, vol. 180, no. 10, pp. 2044–2064, 2010.
[21] R. L. Iman and J. M. Davenport, "Approximations of the critical region of the friedman statistic," *Communications in Statistics-Theory and Methods*, vol. 9, no. 6, pp. 571–595, 1980.
[22] J. Hodges, E. L. Lehmann *et al.*, "Rank methods for combination of independent experiments in analysis of variance," *The Annals of Mathematical Statistics*, vol. 33, no. 2, pp. 482–497, 1962.
[23] B. Bergmann and G. Hommel, "Improvements of general multiple test procedures for redundant systems of hypotheses," in *Multiple Hypothesenprüfung/Multiple Hypotheses Testing*. Springer, 1988, pp. 100–115.
[24] B. Calvo and G. Santafe, "scmamp: Statistical comparison of multiple algorithms in multiple problems," 2015. [Online]. Available: https://cran.r-project.org/web/packages/scmamp/index.html
[25] G. Kochenberger, F. Glover, B. Alidaee, and H. Wang, "Clustering of microarray data via clique partitioning," *Journal of Combinatorial Optimization*, vol. 10, no. 1, pp. 77–92, 2005.
[26] A. Miyauchi and Y. Miyamoto, "Computing an upper bound of modularity," *The European Physical Journal B*, vol. 86, no. 7, pp. 1–7, 2013.
[27] M. Oosten, J. H. Rutten, and F. C. Spieksma, "The clique partitioning problem: facets and patching facets," *Networks*, vol. 38, no. 4, pp. 209–226, 2001.
[28] U. Dorndorf, F. Jaehn, and E. Pesch, "Modelling robust flight-gate scheduling as a clique partitioning problem," *Transportation Science*, vol. 42, no. 3, pp. 292–301, 2008.