Features Clustering Around Latent Variables for High Dimensional Data

EZ-ZARRAD Ghizlane [1*], *SABBAR Wafae* [1], and *BEKKHOUCHA Abdelkrim* [1]

[1] LIM Laboratory, Faculty of Sciences and Techniques Mohammedia, University Hassan II of Casablanca, Morocco

Abstract. *Clustering of variables is the task of grouping similar variables into different groups. It may be useful in several situations such as dimensionality reduction, feature selection, and detect redundancies. In the present study, we combine two methods of features clustering the clustering of variables around latent variables (CLV) algorithm and the k-means based co-clustering algorithm (kCC). Indeed, classical CLV cannot be applied to high dimensional data because this approach becomes tedious when the number of features increases.*

1 Introduction

Cluster analysis is the process of the partitioning of data into subsets of identical characteristics, and it can be used to construct clusters in such a way that objects in the same group are similar to each other [1] [2]. Clustering technique also concerns the task of grouping similar variables into different groups [3].

The Clustering of variables is an alternative technique that allows variables to arrange in homogeneous clusters [4, 5]. Moreover, it may be used to describe objects. The classification of variables around latent variables CLV [4] method involves two stages, a hierarchical clustering algorithm is used to find an initial partition of dataset followed by a partitioning algorithm. CLV method works well when applied to small datasets; mainly, it is hard to apply to high dimensional dataset because of memory and analyze barriers.

Large datasets are difficult to manage, visualize, and analyze on a massive scale in terms of volume. Two computational barriers for large datasets analysis: (i) the data can be too big to support in a computer's memory; (ii) the computing task can take too long to wait for the results [5].

CLV method shows a good results when applied to several datasets; However, in the case of high dimensional data, it is difficult at the time of big dataset [6]. Indeed, it is no longer able to group similar variables to develop meaningful structure. It becomes computationally expensive in terms of memory and execution time requirements.

There is a need to manage such large volumes of data and to cluster them easily for data analysis [7] while choosing a cluster's number and maximizing the clustering criteria for large datasets. Therefore, CLV algorithms should be efficient, scalable, and highly accurate. There is a need to enhance them to suit large datasets.

To deal with this problem, a k-means based co-clustering (kCC) algorithm [8] have proposed better cluster initialization dealing with outliers, instead of the hierarchical clustering algorithm proposed in the CLV algorithm, accordingly, we propose a novel method by using cluster initialization algorithm of kCC in the CLV algorithm.

The rest of this paper is organized as follows. In section 2, the background work related to the CLV and kCC algorithms is briefly described. The proposed solution is explained in section 3. Section 4 contains our obtained results, and we conclude this paper by discussing some limitations and prospects in Section 5.

2 Background Literature

2.1 Clustering of variables around latent variables

The objective of Classification of variables around Latent Variables (CLV) [4] is to find meaningful clusters of variables according to the criteria and algorithms on which the method is based.

Let us denote by:

- $\mathcal{D} = \left(d_{ij}\right)_{1\leq i\leq n, 1\leq j\leq p}$ a dataset with p centred variables observed on n examples.

- $\mathcal{F} = \{x_1, x_2, \dots, x_p\}$ of p columns of $\mathcal{D}$.

- $\mathcal{P} = \{C_1, C_2, \dots, C_M\}$ a partition of $\mathcal{F}$ into M clusters of variables associated with $\mathcal{U} = \{u_1, u_2, \dots, u_M\}$ a set of latent variables.

* G.ezzarrad@gmail.com

The aim of CLV method is to get a couple $(\mathcal{P}, \mathcal{U})$, and thus to maximize the internal coherence of the clusters concerning the following clustering criterion:

$$\mathcal{L}(\mathcal{P}, \mathcal{U}) = \sum_{m=1}^{M} \Gamma(C_m, u_m) \qquad (1)$$

where Γ measures the linear link, in each cluster C_m, between the variables x_j in this cluster and u_m the associated latent variable.

Two types of criteria $(\mathcal{P}, \mathcal{U})$ are considered, which define two different cluster types of variables: directional groups and local groups.

Directional groups

Positively or negatively correlated variables will be merged together, no matter whether their sign of the correlation coefficients (Figure 1), and the Γ considered for maximization on (1) is:

$$\Gamma_D(C_m, u_m) = \sum_{x_j \in C_m} \gamma_{mj} cov^2(x_j, u_m) \qquad (2)$$

with $(x_j)_{1 \le j \le p}$ are the p variables to be clustered, $cov^2(x_j, u_m)$ the squared correlation measuring the link between the variable x_j and the latent variable u_m, the variance of u_m equal to 1, and $\gamma_{mj} = \begin{cases} 1 \text{ if } x_j \in C_m \\ 0 \ otherwise. \end{cases}$

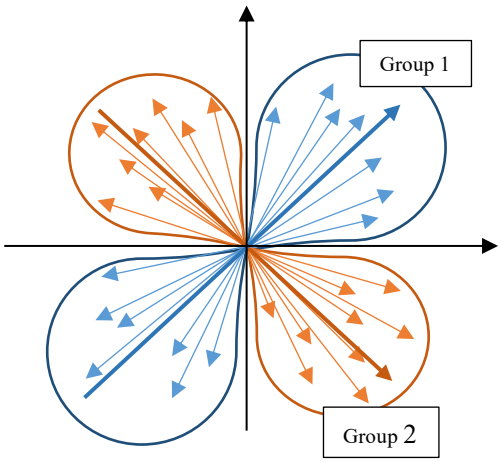

Fig. 1. Directional groups: positively and negatively correlated variables.

Local groups

Only positively correlated variables will be associated in the same group (Figure 2), and the Γ considered for maximization on (1) is:

$$\Gamma_L(C_m, u_m) = \sum_{x_j \in C_m} \gamma_{mj} cov(x_j, u_m) \qquad (3)$$

with $(x_j)_{1 \le j \le p}$ are the p variables to be clustered, $cov(x_j, u_m)$ the covariance measuring the link between the variable x_j and the latent variable u_m, the variance of u_m equal to 1, and $\gamma_{mj} = \begin{cases} 1 \text{ if } x_j \in C_m \\ 0 \ otherwise. \end{cases}$

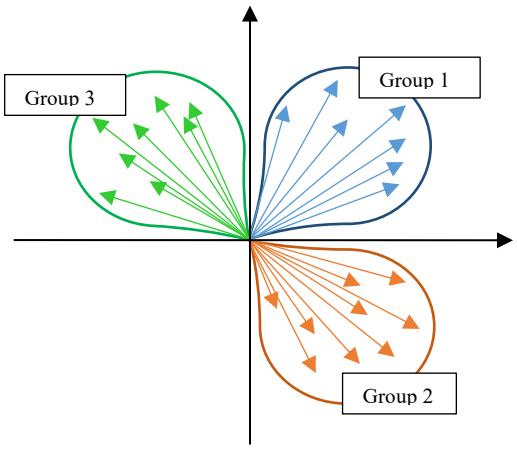

Fig. 2. Local groups: positively correlated variables.

Latent Variable

In cluster C_m, the latent variable u_m used on (2,3) is defined as: the first standardized principal component of X_m formed with the variables belonging to their C_m for direction groups, and the standardized centroid variable in C_m for local groups.

2.2 Partitioning Algorithm

The aim of a partitioning algorithm is to find an optimum couple $(\mathcal{P}, \mathcal{U})$, therefore it is used for optimization of the clustering criterion $\mathcal{L}(\mathcal{P}, \mathcal{U})$ given in (1).

It consists of two alternating steps: the estimation step, on which the latent variables u_m are defined given the initial partition $\mathcal{P}$, and the allocation step of the variables given the latent variables. More precisely, this algorithm is developed as follows:

Algorithm 1 Partitioning Algorithm

Input: $\mathcal{P}^0 = (C_1^0, C_2^0, \ldots, C_M^0)$.
Output: $(\mathcal{P}, \mathcal{U})$.
Repeat until convergence
Estimation Phase
Find $\mathcal{U} = u_1, u_2, \ldots, u_M$
Compute u_m the latent variable of C_m;
for $m = 1, \ldots, M$ do
Directional groups: u_m is the first standardized principal component of the matrix X_m formed by the variables belonging to C_m.
Local groups: u_m is the centroid variable of the variables in C_m.
end for
Assignment Phase
Find $\mathcal{P}^i = \left(C_1^i, C_2^i, \ldots, C_M^i\right)$.
Each variable is allowed to the cluster if its correlation criterion between x_j and u_m is higher;
for i=1 to maximum iterations **do**
while $\mathcal{P}^i != \mathcal{P}^{i+1}$ **do**
find a such that,
Directional groups: $a = \underset{m=1\ldots M}{\operatorname{argmax}}\ cov^2\left(x_j, u_m\right)$
Local groups: $a = \underset{m=1\ldots M}{\operatorname{argmax}}\ cov\left(x_j, u_m\right)$
let $\mathcal{P}^{i-1} = \left(C_1^{i-1}, C_2^{i-1}, \ldots, C_M^{i-1}\right)$ the previous partition of C_m,
if $a \neq m$
$C_a^i = C_a^{i-1} \cup \{x_j\}$ and, $C_m^i = C_m^{i-1} \backslash \{x_j\}$
else stop
end if
end while
end for

2.3 k-means based co-clustering algorithm

In the case of k-means algorithm, the optimization problem is to find a number of homogeneous clusters. However, finding the initial k cluster centers is a big challenge, these initial clusters are chosen at random, consequently, it can lead to poor starting points. This means that the final clusters of k-means are highly dependent on the initial clusters. To improve the selection of such cluster centers, k-means based co-clustering (kCC) algorithm proposes a new cluster initialization algorithm based on a random walk based similarity measure.

This method constructs clusters in such a way that objects/variables are merged according to find L data objects/variables instead of a single cluster center. These selecting L points represent each cluster. This helps to decrease the chance of selecting a single outlier. The points selected in the same cluster have high intra-cluster similarity amongst themselves but have low inter-cluster similarity with all other clusters.

Given two documents $d_r$and d_s, the t[th] order walk between them is given by

$$D_{rs}^t = \sum_{\alpha=1}^{p}\sum_{\beta=1}^{p} u_{r\alpha}.W_{\alpha\beta}^{t-1}.u_{\beta s} \qquad (4)$$

and the t[th] order walk between two words w_r and w_s is given by

$$W_{rs}^t = \sum_{\alpha=1}^{n}\sum_{\beta=1}^{n} v_{r\alpha}.D_{\alpha\beta}^{t-1}.v_{\beta s} \qquad (5)$$

Where $u_{ij}\,(v_{ij})$ are the probability of starting at document d_i(word w_i) and arriving at word w_j (document d_j) given by $u_{ij} = \frac{x_{ij}}{deg(x_i)}$ and $v_{ij} = \frac{x_{ij}}{deg(x_j)}$

Using the similarity measure given in (4), the l^{th} $(l \in 1..L)$ cluster center for the m^{th} $(m \in 1..M)$ cluster is given by

$$c_{ml} = \underset{\alpha=1\ldots n}{\operatorname{argmax}}\left(\frac{\sum_{\beta=1}^{l-1} D_{\alpha\beta}^{t-1}}{l-1} + \frac{\sum_{\gamma=1}^{m-1} 1 - D_{\alpha\gamma}^{t-1}}{(l-1)(m-1)}\right) \qquad (6)$$

The aim of the kCC algorithm is to find M number of initial clusters $\mathcal{P}^0 = (C_1^0, C_2^0, \ldots, C_M^0)$, it is considered as the initialization step on the CLV algorithm (algorithm 1)

Algorithm 2 kCC Algorithm

Input: $\boldsymbol{D} = \left(\boldsymbol{d_{ij}}\right)$ a data set, number of centroid clusters L, and M number of clusters.
Output: $\mathcal{P}^0 = (C_1^0, C_2^0, \ldots, C_M^0)$.
Repeat until convergence
Find $c_{11}, \ldots, c_{ML}$ centroid clusters;
Select at random x_j as the first center of the first cluster c_{11}.
for $l = 1 \ldots L$
for $m = 1 \ldots M$
Set c_{ml} using (6)
endfor
endfor

3 Proposed kCC+CLV

The problem of initializing the partitioning algorithm is solved in the CLV method by implementing a hierarchical clustering algorithm (HCA) [12] based on the maximization of the criteria $\mathcal{L}(\mathcal{P},\mathcal{U})$.

However, when the number of variables is large, the HCA needs time consuming to execute. For this reason, we change the initialization phase with the kCC method. This method has the same parameters and options as the classic k-means [13] but performs only the partitioning procedure. In this case, the objective is to find a partition $\mathcal{P}^0 = (C_1^0, C_2^0, \ldots, C_M^0)$ of a set $\mathcal{D} = \left(d_{ij}\right)$. Indeed, the result of the algorithm dependent not only on the choice of the number of clusters but especially on the initial cluster centers. The number of clusters M, should be given as an input parameter.

4 Results

To evaluate the performance of the proposed method kCC+CLV, an experiment was used on artificial data to analyze its effectiveness compared to other methods. The results are evaluated using the accuracy mesure which describes the correctness of an algorithm with respect to the real class labels of the data.

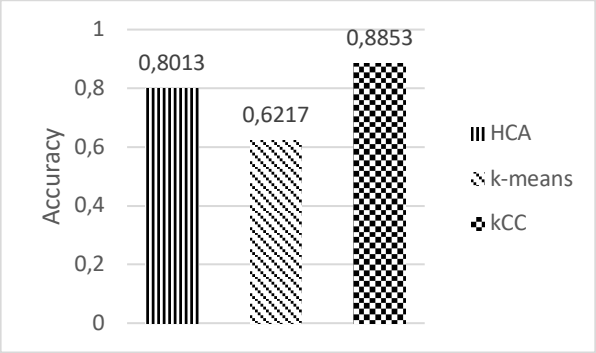

Figure 3 : Accuracy of cluster initialization algorithms

The results of the accuracy measure, kCC shows an improvement of 62% over the other methods h-means and HCA.

In general, the running time of our proposed kCC is lower than other approaches when either the number of variables is high or the number of clusters are large (Table 1).

	p=50	p=100	p=1000	p=10000	p=50000
CLV+HCA	2.03	102.15	506.15	-----	----
CLV+ k-means	3.17	153.08	405.6	567.19	684.25
CLV+ kCC	3.00	142.12	315.75	388.26	408.64

Table 1 : CPU time in seconds for n= 10000 objects in varying number of variables

We calculate the CPU time in seconds for p variables (from p=50 to p=50000) and n fixed objects (n=1000). Table 1 shows that the kCC+CLV remain fast even if the number p of variables increases. In the same case, the CLV with HCA is the slower function.

Conclusion

Clustering method converts information into various clusters where the object in that group has similar properties as compared to other but not same to other clusters properties.

The iterative procedure of CLV algorithm initialization supplies a partition into M clusters, which maximizes the criteria $\mathcal{L}$.

However, this optimum is often local and can depend on the initial partition [14].

A solution to avoid this problem and to reduce the influence of the arbitrary choice of the initial partition is to consider kCC initialization algorithm.

In this case, all phases in the algorithm (algorithm1) are repeated several times, and the highest value of $\mathcal{L}$ is considered as the best one.

References

1. L. Kaufman and RJ. Rousseeuw, Finding Groups Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.
2. H. C. Romesburg, Cluster Analysis for Researchers. Lifetime Learning Publications, Belmont, CA, 1984.
3. J. A. Hartigan, Direct Clustering of a Data Matrix. Journal of the American Statistical Association, 67:337, 123-129, 1972.
4. E. Vigneau and E. M. Qannari, Clustering of Variables Around Latent Components. Communications in Statistics: Simulation and Computation, 32(4), 1131-1150, 2003.
5. C. Wang, M. Chen, E. Schifano, J. Wu and J. Yan, Statistical methods and computing for big data. Statistics and its interface, 9(4), 399–414, 2016.
6. A. M. El-Mandouh, H. A. Mahmoud, L. A. Abd-Elmegid and M. H. Haggag, Big Data Clustering Model based on Fuzzy Gaussian. International Journal of Computer Science and Information Security (IJCSIS), Vol. 16, No. 4, 2018.
7. C. Sreedhar, N. Kasiviswanath and P. C. Reddy, Clustering large datasets using K-means modified inter and intra clustering (KM-I2C) in Hadoop. Journal of Big Data, no. 1, 4-27, 2017.
8. S. F. Hussain, M. Haris , A k-means based Co-clustering (kCC) Algorithm for Sparse, High Dimensional Data, *Expert Systems With Applications* (2018)
9. M. Chavent, V. Kuentz and J. Saracco, A partitioning method for the clustering of categorical variables. Classification as a Tool for Research, Springer, 91–99, 2010
10. J. Saracco, M. Chavent and V. Kuentz, Clustering of categorical variables around latent variables. Working Papers of GREThA, n2010-02, http://ideas.repec.org/p/grt/wpegrt/2010-02.html, 2010.
11. M. Chavent, V. Kuentz, B. Liquet and J. Saracco, ClustOfVar: An R Package for the Clustering of Variables. Journal of Statistical Software, University of California, Los Angeles, 50 (13), 1-16, 2012.
12. E. Vigneau, M. Chen and E. M. Qannari, ClustVarLV: an R package for the clustering of variables around latent variables. The R Journal, 7(2), 134-148, 2015.
13. T. Zhang and B. Yang, Big Data Dimension Reduction Using PCA. In Proceedings of the 2016 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 152–157, 2016.
14. E. Forgy. Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. *Biometrics*, 21, pp. 768–769. 1965.