17th European Symposium on Computer Aided Process Engineering – ESCAPE17
V. Plesu and P.S. Agachi (Editors)

A Novel Clustering Approach: Global Optimum Search with Enhanced Positioning

Meng P. Tan,[a] James R. Broach, [b] Christodoulos A. Floudas,[a]

[a]*Department of Chemical Engineering, Princeton University, Princeton, NJ 08544, USA, floudas@titan.princeton.edu*
[b]*Department of Molecular Biology, Princeton University, Princeton, NJ 08544*

Abstract

Cluster analysis of DNA expression data is a useful tool for identifying biologically relevant gene groupings. It is hence important to apply a rigorous yet intuitive clustering algorithm to uncover these genomic relationships. Here, we describe a clustering framework [1,2] based on a variant of the Generalized Benders Decomposition, the Global Optimum Search [3,4]. We apply the proposed algorithm to experimental DNA microarray data and compare the results to that obtained with some commonly-used algorithms. We also propose an extension to iteratively uncover the optimal biologically coherent structures.

Keywords

Clustering, Expression Data, Optimization, Global Optimum Search

1. Introduction

The aim of cluster analysis is to establish a set of clusters such that the data in a cluster are more similar to one another than they are anywhere else. Clustering is used in many disciplines, such as market research, social network analysis, and geology, thus reflecting its broad utility as a key step in exploratory data analysis [5]. In biology, identifying genes that are co-regulated provides helps to extract regulatory motifs for transcription factors, allowing assembly of predictive transcriptional networks [6]. This information also provides insights into the functions of unknown genes, since functionally related genes are often co-regulated [7]. Furthermore, clustered data provides identification of distinct categories of otherwise indistinguishable cell types, which can have huge implications in areas such as disease progression [8]. In sequence analysis, clustering is used to group homologous sequences into gene families.

Two popular similarity metrics are correlation and Euclidean distance. The latter is often used, since it is intuitive, can be described by a familiar distance function, and satisfies the triangular inequality. Clustering methods that employ asymmetric distance measures [9, 10] are more difficult to intuitively comprehend even though they may be well suited to their intended applications. The earliest work on clustering emphasized visual interpretations for the ease of study, resulting in methods that utilize dendograms and color maps [11]. Other examples of clustering algorithms are (a) Hierarchical Clustering, (b) K-Methods, (c) Fuzzy Clustering, (d) Quality Cluster Algorithm (QTClust), (e) Graph-Theoretic Clustering, (f) Artificial Neural Networks for Clustering such as the Self-Organizing Map (SOM) and a variant that combines the SOM with hierarchical clustering, the Self-Organizing Tree Algorithm (SOTA), and (g) Information-Based Clustering.

2. Proposed Approach

2.1. Notation

We denote the measure of distance for a gene i, for i = 1,….,n having k features, for k = 1,….., s as a_{ik}. Each gene is to be assigned to only one of c possible clusters, each with center z_{jk}, for j = 1,….,c. The binary variables w_{ij} indicates whether gene i falls within cluster j (w_{ij} = 1, if yes; w_{ij} = 0, if no).

2.1.1. Hard Clustering by Global Optimization

The approach minimizes the Euclidean distances between the data and the assigned cluster centers as:

$$\underset{w_{ij},z_{jk}}{\mathrm{MIN}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \sum_{j=1}^{c} \sum_{k=1}^{s} w_{ij} \left(a_{ik} - z_{jk}\right)^2$$

To handle the nonlinear product of the variables w_{ij} and z_{jk}, we can introduce new variables y_{ijk} along with additional constraints [3] to reduce the formulation to an equivalent Mixed-Integer Linear Programming (MILP) problem. This however results in a very large number of variables. Without the y_{ijk} variables however the problem is nonlinear, which is difficult to solve. Theoretical advances and prominent algorithms for solving such problems are addressed in [3,12,13]. We use a variant of the Generalized Benders Decomposition (GBD) algorithm [3, 4], the Global Optimum Search (GOS) to handle the nonlinear problem. The GOS decomposes the problem into a primal problem and the master problem. The former solves the continuous variables while fixing the integer variables and provides an upper bound solution, while the latter finds the integer variables and the associated Lagrange multipliers while fixing the continuous variables and provides a lower bound solution. The two sequences are iteratively updated until they converge at an optimal solution in a finite number of steps.

2.1.2. Determining the Optimal Number of Clusters

Most algorithms do not contain screening functions to determine the optimal cluster number. On the other hand, while it easy to propose indices of cluster validity, it is difficult to incorporate these measures into clustering algorithms and appoint thresholds on which to define key decision values [13,14]. Some indices used to measure cluster validity are the Dunn's validity index, the Davis-Bouldin validity index, and the Silhouette validation technique. We adapt the concept of a clustering balance [15], which is a weighted sum of two error sums and has been shown to have a minimum value when intra-cluster similarity is maximized and inter-cluster similarity is minimized.

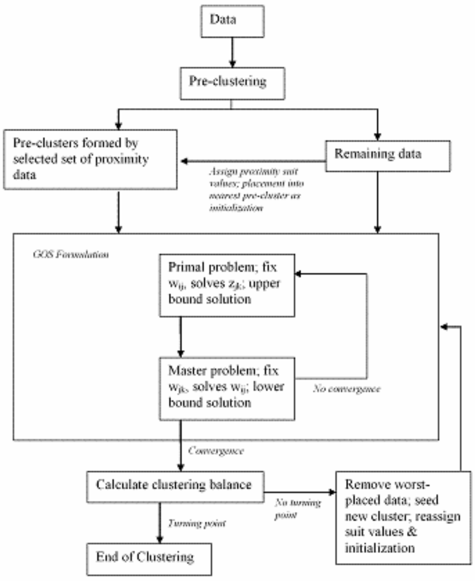
2.1.3. Proposed Algorithm (EP_GOS_Clust)

Gene Pre-Clustering: We pre-cluster the original data by proximity studies to reduce the computational demands by (i) identifying genes with very similar responses, and (ii) removing outliers deemed to be insignificant to the clustering process. To provide just adequate discriminatory characteristics, pre-clustering can be done by reducing the expression vectors into a set of representative variables {+, o, -}, or by pre-grouping genes that are close to one another by correlation or some other distance function.

Iterative Clustering: We let the initial clusters be defined by the genes pre-clustered previously, and find the distance between each of the remaining genes and these initial clusters and as a good initialization point placed these genes into the nearest cluster. For each gene, we allow its suitability in a limited number of clusters based on the proximity study. In the primal problem of the GOS algorithm, we solve for z_{jk}. These, together with the Lagrange multipliers, are used in the master problem to solve for w_{ij}. The primal gives an upper bound solution and the master a lower bound. The optimal solution is obtained when both bounds converge. Then, the worst-placed gene is removed and used as a seed for a new cluster. This gene has already been subjected to a membership search so there is no reason for it to belong to any one of the older clusters. The iterative steps are repeated and the number of clusters builds up gradually until the optimal number is attained.

Iterative Extension: Indication of strong biological coherence is characterized by good P-values based on gene ontology resources and the proportion of genes that reside in such clusters. As an extension, we would like to mine for the maximal amount of relevant information from the gene expression data and sieve out the least relevant data [16]. This is important because information such as biological function annotation drawn from the cluster content is often used in the further study of co-regulated gene members, common reading frames, and gene regulatory networks. From the clustered genes, we impose a coherence floor to demarcate genes that have already been well clustered. We then iterate to offer the poorly-placed genes an opportunity to either find relevant membership in one of the strongly coherent clusters, or regroup amongst themselves to form quality clusters. Through this process, a saturation point will be reached eventually whereby the optimal number of clusters becomes constant as the proportion of genes distributed within clusters of high biological coherence levels off.

Figure 1. Schematic flowchart of the EP_GOS_Clust algorithm. Although the formulation in the paper has been notated for DNA microarray data, the algorithm framework can be adapted for clustering any numeric data.


Data
Pre-clustering
Pre-clusters formed by selected set of proximity data
Remaining data
Assign proximity suit values; placement into nearest pre-cluster as initialization
GOS Formulation
Primal problem; fix w_{ij}, solves z_{jk}; upper bound solution
Master problem; fix w_{jk}, solves w_{ij}; lower bound solution
No convergence
Convergence
Calculate clustering balance
No turning point
Remove worst-placed data; seed new cluster; reassign suit values & initialization
Turning point
End of Clustering

A schematic of the EP_GOS_Clust algorithm can be seen in Figure 1.

2.2. Case study

In this study, we used experimental microarray data from a study in the role of the Ras/protein kinase A pathway (PKA) on glucose signaling in yeast [17]. These experiments analyzed mRNA levels in cell samples extracted at various times following stimulation by glucose or following activation of either Ras2 or Gpa2, which are small GTPases involved in the metabolic and transcriptional response of yeast cells to glucose [18]. Levels of RNA for each of the 6237 yeast genes were measured using Affymetrix microarray chips and after filtering, we retained 5652 genes. The clustering algorithms to be compared are (a) K-Means, (b) K-Medians, (c) K-Correlation, (d) K-CityBlock, (e) K-AvePair, (f) QTClust, (g) SOM, (h) SOTA, and (i) EP_GOS_Clust.

2.3. Results & discussions

A good clustering procedure should minimize the intra-cluster error sum and maximize the inter-cluster error sum. Even without the iterative extension, we found the error sums of the clusters found using EP_GOS_Clust outperform that of the other clustering methods. Also, EP_GOS_Clust predicts the lowest number of optimal clusters. Together with the quality of the error sum comparisons, we infer the superior ‘economy’ of EP_GOS_Clust in producing tighter data groupings by utilizing a lower number of clusters, as it is actually possible to achieve tight groupings by using a large number of clusters, even with an inferior clustering algorithm. We also found EP_GOS_Clust capable of uncovering strongly correlated clusters with high levels of biological coherence. Tables 1 and 2 shows that EP_GOS_Clust performs consistently well when compared against the significance of cluster biological coherence uncovered by the other clustering methods, and this is before the application of the proposed extension.

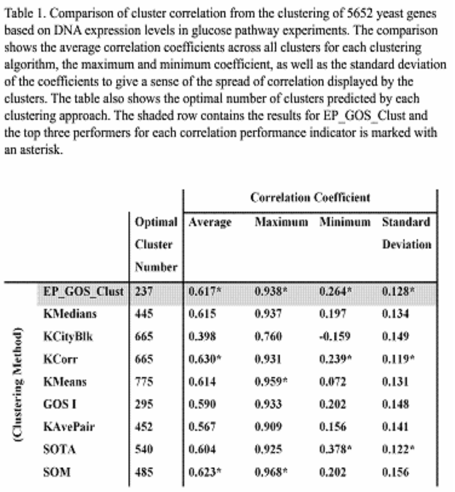


Table 1. Comparison of cluster correlation from the clustering of 5652 yeast genes based on DNA expression levels in glucose pathway experiments. The comparison shows the average correlation coefficients across all clusters for each clustering algorithm, the maximum and minimum coefficient, as well as the standard deviation of the coefficients to give a sense of the spread of correlation displayed by the clusters. The table also shows the optimal number of clusters predicted by each clustering approach. The shaded row contains the results for EP_GOS_Clust and the top three performers for each correlation performance indicator is marked with an asterisk.

			Correlation Coefficient			
		Optimal Cluster Number	Average	Maximum	Minimum	Standard Deviation
(Clustering Method)	EP_GOS_Clust	237	0.617*	0.938*	0.264*	0.128*
	KMedians	445	0.615	0.937	0.197	0.134
	KCityBlk	665	0.398	0.760	-0.159	0.149
	KCorr	665	0.630*	0.931	0.239*	0.119*
	KMeans	775	0.614	0.959*	0.072	0.131
	GOS I	295	0.590	0.933	0.202	0.148
	KAvePair	452	0.567	0.909	0.156	0.141
	SOTA	540	0.604	0.925	0.378*	0.122*
	SOM	485	0.623*	0.968*	0.202	0.156

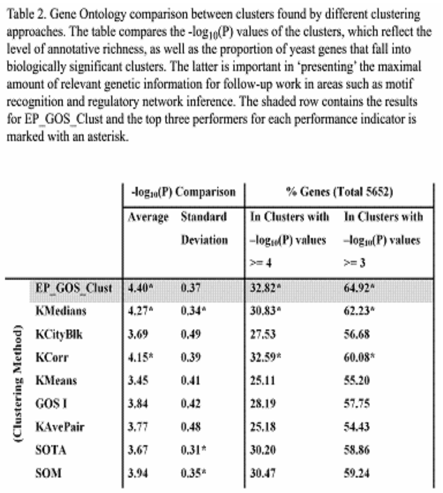


Table 2. Gene Ontology comparison between clusters found by different clustering approaches. The table compares the -$\log_{10}$(P) values of the clusters, which reflect the level of annotative richness, as well as the proportion of yeast genes that fall into biologically significant clusters. The latter is important in ‘presenting’ the maximal amount of relevant genetic information for follow-up work in areas such as motif recognition and regulatory network inference. The shaded row contains the results for EP_GOS_Clust and the top three performers for each performance indicator is marked with an asterisk.

		-$\log_{10}$(P) Comparison		% Genes (Total 5652)	
		Average	Standard Deviation	In Clusters with –$\log_{10}$(P) values >= 4	In Clusters with –$\log_{10}$(P) values >= 3
(Clustering Method)	EP_GOS_Clust	4.40*	0.37	32.82*	64.92*
	KMedians	4.27*	0.34*	30.83*	62.23*
	KCityBlk	3.69	0.49	27.53	56.68
	KCorr	4.15*	0.39	32.59*	60.08*
	KMeans	3.45	0.41	25.11	55.20
	GOS I	3.84	0.42	28.19	57.75
	KAvePair	3.77	0.48	25.18	54.43
	SOTA	3.67	0.31*	30.20	58.86
	SOM	3.94	0.35*	30.47	59.24

With the extension, we found the original clustering results to be significantly improved. For instance, the proportion of genes that fall in clusters of -log(P) values of above 3 went from 65% to over 80% and the average cluster correlation improved by over 5%. This showed the extension to be useful and relevant in refining the initial clusters for optimal biological coherence.

We have also tested the EP_GOS_Clust algorithm with its extension on a number of other data sets (not described here) and have shown that the level of clustering quality is consistently high compared to other clustering techniques and that the extension is able to improve the clusters’ level of biological coherence.

3. Conclusions

In our study, we propose a novel clustering algorithm (EP_GOS_Clust) based on an MINLP formulation, and apply a novel decomposition technique to solve the MINLP optimization problem. We test our proposed algorithm on a large dataset of gene expression patterns from the yeast Saccharomyces Cerevisiae, and show that our method compares favorably with other clustering methods. We also highlighted an extension to the clustering algorithm that is able to further refine the level of biological coherence of the clusters, which is particularly useful for further genomic and cellular network research.

Acknowledgements

The authors gratefully acknowledge support from the National Science Foundation and the National Institutes of Health.

References

1. Tan, M.P., Broach, J. R., Floudas, C. A. *Submitted for Publication*
2. Tan. M. P., Broach, J. R., Floudas, C. A. *Submitted for Publication*
3. Floudas, C. A.: Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications. Oxford University Press (1995)
4. Floudas, C. A., Aggarwal, A., Ciric, A. R. Comp. & Chem. Eng. 13(10), 1117-1132 (1989)
5. Jain, A. K., Murty, M. N., Flynn, P. J. ACM Computing Surveys 31(3), 264-323 (1999)
6. Beer, M., Tavazoie, S. Cell 117, 185-198 (2004)
7. Troyanska, O. G. et al. Proc. Nat. Acad. Sci. U.S.A. 100, 8348-8353 (2003)
8. Sorlie, T. et al. Proc. Nat. Acad. Sci. U.S.A. 100, 8418-8423 (2003)
9. Pipenbacher, P. et al. Bioinformatics 18 (Supplement 2), S182-191 (2002)
10. Leisch, F., Weingessel, A., Dimitriadou, E. Proceedings of the 8th Int. Conference on Artificial Neural Networks 2, 779-784, Sk"ovde, Sweden. Springer. (1998)
11. Claverie, J. Human Molecular Genetics 8, 1821-1832 (1999)
12. Wang, Y. et al. Plos Biology 2(5), 610-622 (2004)
13. Schneper, L., Düvel, K., Broach, J. R. Curr. Opin. Microbiol. 7(6), 624-630 (2004)
14. Floudas, C. A.: Deterministic Global Optimization: Theory, Algorithms, and Applications. Kluwer Academic Publishers (2000)
15. Floudas, C. A. et al. Computers and Chemical Engineering 29, 1185-2002 (2005)
16. Tan. M. P., Broach, J. R., Floudas, C. A. *Submitted for Publication*
17. Halkidi, M., Batistakis, Y., Vazirgiannis, M. SIGMOD Record 31(2), 40-45 (2002)
18. Jung, Y. et al. Journal of Global Optimization 25, 91-111 (2003)