COMPACT: A Comparative Package for Clustering Assessment

Roy Varshavsky[1,*], Michal Linial[2], and David Horn[3]

[1] School of Computer Science and Engineering,
The Hebrew University of Jerusalem 91904, Israel
royke@cs.huji.ac.il
[2] Dept of Biological Chemistry, Institute of Life Sciences,
The Hebrew University of Jerusalem 91904, Israel
michall@cc.huji.ac.il
[3] School of Physics and Astronomy, Tel Aviv University, Israel
horn@post.tau.ac.il

Abstract. There exist numerous algorithms that cluster data-points from large-scale genomic experiments such as sequencing, gene-expression and proteomics. Such algorithms may employ distinct principles, and lead to different performance and results. The appropriate choice of a clustering method is a significant and often overlooked aspect in extracting information from large-scale datasets. Evidently, such choice may significantly influence the biological interpretation of the data. We present an easy-to-use and intuitive tool that compares some clustering methods within the same framework. The interface is named COMPACT for **Com**parative-**Pa**ckage-for-**C**lustering-Assessmen**t**. COMPACT first reduces the dataset's dimensionality using the Singular Value Decomposition (SVD) method, and only then employs various clustering techniques. Besides its simplicity, and its ability to perform well on high-dimensional data, it provides visualization tools for evaluating the results. COMPACT was tested on a variety of datasets, from classical benchmarks to large-scale gene-expression experiments. COMPACT is configurable and expendable to newly added algorithms.

1 Introduction

In the field of genomics and proteomics, as well as in many other disciplines, classification is a fundamental challenge. Classification is defined as systematically arranging entities (data-points) into specific groups. Clustering, being an unsupervised learning problem, may be regarded as a special case of classification with unknown labels (for more details see [1], [2]). In gene expression microarray technology, a hierarchical clustering algorithm was first applied to gene-expression data at different stages of cell cycle in yeast [3]. During recent years several algorithms, originating from various theoretical disciplines (e.g., physics, mathematics, statistics and computational neuroscience), were adopted and adjusted to gene expression analysis. They

[*] Corresponding author.

G. Chen et al. (Eds.): ISPA Workshops 2005, LNCS 3759, pp. 159 – 167, 2005.

are useful for diagnosis of different conditions for example differentiating between sick and healthy tissues, and classification to subtypes of a disease. An additional outcome of applying such algorithms to gene-expression data was the revealing of functional classes of genes among the thousands used in experimental settings [4]. Furthermore, it became possible, and useful, to isolate groups of relevant genes that mostly contribute to a particular condition, in the correlative or derivative perspective, a procedure called bi clustering [5].

By their nature, data points that are collected from large-scale experimental settings suffer from being represented in a high dimensional space. This fact presents a computational and an applicative challenge. Compression methods that maintain the fundamental properties of the data are called for.

As clustering algorithms are rooted in different scientific backgrounds and follow different basic principles, it is expected that different algorithms perform differently on varied inputs. Therefore, it is required to identify the algorithm that suits best a given problem. One of the targets of COMPACT is to address this requirement, and to supply an intuitive, user-friendly interface that compares clustering results of different algorithms.

In this paper we outline the key steps in using COMPACT and illustrate it on two well-known microarray examples of Leukemia [4], and yeast datasets [6]. For a comparative analysis we included routinely used clustering algorithms and commonly applied statistical tests, such as K-Means, Fuzzy C-Means and a competitive neural network. One novel method, Quantum Clustering (QC) [7], was added to evaluate its relative performance. The benefit of applying COMPACT to already processed data is demonstrated. All four algorithms that were applied in analyzing these datasets were compared with a biologically based validated classification. We conclude that the compression of data that comprises the first step in COMPACT, not only reduces computational complexity but also improves clustering quality. Interestingly, in the presented tested datasets the QC algorithm outperforms the others.

2 Implementation

After downloading and configuring COMPACT, four steps should be followed: defining input parameters, preprocessing, selecting the clustering method and presenting the results.

2.1 Input Parameters

COMPACT receives two input parameters that are Matlab variables: data (a two-dimensional matrix) – represents the elements to be clustered, and 'real classification' (an optional, one-dimensional vector) – representing the elements according to an expert view and is based on bulk biological and medical knowledge.

2.2 Preprocessing

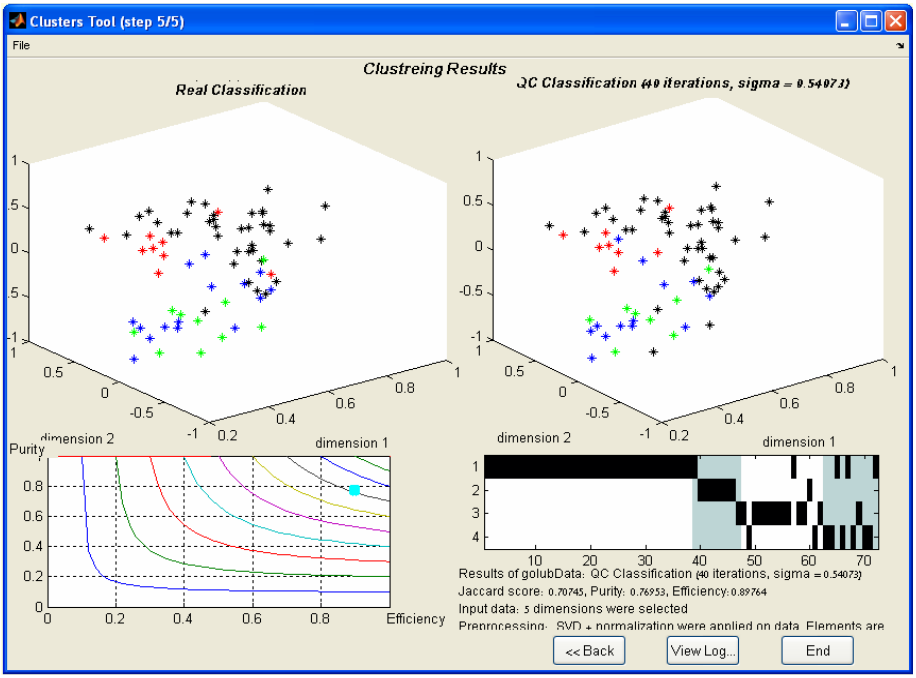
a) Determining the matrix shape and which vectors are to be clustered (rows or columns).
b) Preprocessing Procedures: SVD, normalization and dimension selection.

2.3 Selecting the Clustering Method

a) Points' distribution preview and clustering method selection: The elements of the data matrix are plotted. If a 'real classification' exists, each of its classes is displayed in a different color. One of the clustering methods, K-means, FCM (fuzzy C-means), Competitive NN (Neural Network) or QC (Quantum Clustering) is to be chosen from the menu.
b) Parameters for clustering algorithms: depending on the chosen method, a specific set of parameters should be defined (e.g., in the K-Means method – number of clusters).

2.4 COMPACT Results

Once COMPACT completes its run, the results are displayed in both graphical and textual formats (results can be displayed also in a log window). In the graphical display, points are tagged by the algorithm. The textual display represents Purity and Efficiency (also known as precision and recall or specificity and sensitivity, respectively) as well as the joint Jaccard Score[1]. These criteria for clustering assessment are defined as follow:



```
Results of golubData: QC Classification (40 iterations, sigma = 0.54073)
Jaccard score: 0.70745, Purity: 0.76953, Efficiency: 0.89764
Input data: 5 dimensions were selected
Preprocessing: SVD + normalization were applied on data. Elements are
```

Fig. 1. A screenshot of the graphical view on the results produced by COMPACT

[1] The Jaccard score reflects the 'intersection over union' between the algorithm and 'real' clustering, and its values range from 0 (void match) to 1 (perfect match).

$$Purity = \frac{n_{11}}{n_{11} + n_{01}}, \ \ Efficiency = \frac{n_{11}}{n_{11} + n_{10}}, \ \ Jaccard = \frac{n_{11}}{n_{11} + n_{01} + n_{10}} \quad (1)$$

Where:

- n_{11} is the number of pairs that are classified together, both in the 'real' classification and in the classification obtained by the algorithm.
- n_{10} is the number of pairs that are classified together in the correct classification, but not in the algorithm's classification.
- n_{01} is the number of pairs that are classified together in the algorithm's classification, but not in the correct classification.

Ending the application will add a new variable to the Matlab workspace: calcMapping - a one-dimensional vector that represents the calculated classification of the elements.

3 Results

We applied several of the most commonly used clustering algorithms for gene expression data. By analyzing the results of COMPACT we observe significant variations in performance. In the following we compare the performance on different datasets. We choose to use datasets that were heavily studied and for which an expert view is accepted.

3.1 COMPACT Tests of Leukemia Microarray Expression Data

We tested COMPACT on the dataset of Golub et al. [4] that has served already as a benchmark for several clustering tools (e.g. [2], [8], [9], [10], [11]). The experiment

Table 1. COMPACT based comparison for the Golub dataset [4]. For details see text.

Method	Jaccard	Purity	Efficiency
Raw data			
K Means	0.257	0.369	0.459
Fuzzy C Means (FCM)	0.272	0.502	0.372
Competitive Neural Network (NN)	0.297	0.395	0.547
Quantum Clustering (QC)	NA	NA	NA
Preprocessing (SVD)			
K Means	0.4	0.679	0.494
Fuzzy C Means (FCM)	0.316	0.584	0.408
Competitive Neural Network (NN)	0.442	0.688	0.553
Quantum Clustering (σ= 0.54)	0.707	0.77	0.898

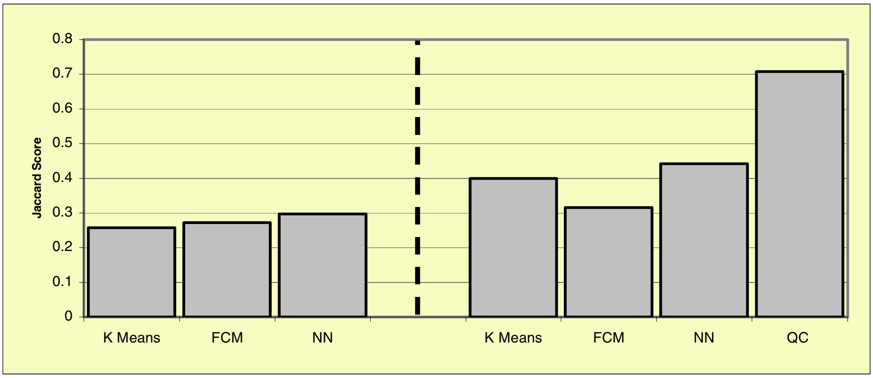


Fig. 2. Jaccard scores of the four algorithms tested by COMPACT on the Golub dataset. Left: before compression, Right: following application of the SVD compression step. Note that an improvement is detected for all methods by a preprocessing step.

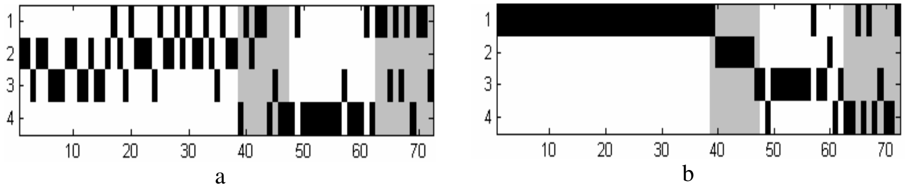


Fig. 3. A graphical comparison of COMPACT results on Leukemia dataset. The samples (patients) are ordered by their groups: samples 1-37: group #1, samples 38-47: group #2, samples 48-62: group #3 and samples 63-72: group #4. The four 'real' classes are distinguished by the background color (white, gray, white and gray), whereas black bars demonstrate the algorithm's classification. For example, in (a) the first sample belongs to the 'correct' first group (white background); while the algorithm placed it in the second group (the black bar is at group #2). Shown are the results of (a) K-means (K=4) and (b) QC (Quantum clustering, $\sigma = 0.54$) for clustering the AML/ALL cancer cells after SVD truncation to 5 dimensions.

sampled 72 leukemia patients with two types of leukemia, ALL and AML. The ALL set is further divided into T-cell leukemia and B-cell leukemia and the AML set is divided into patients who have undergone treatment and those who did not. For each patient an Affymetrix chip measured the expression of 7129 genes.

The clustering results for four selected clustering algorithms are shown in Table 1. A comparison of the Jaccard scores for all algorithms is displayed in Figure 2 and two clustering assignments are compared in Figure 3. Applying the selected algorithms to the raw data (i.e., without an SVD preprocessing) yields poor outcomes.

Next we applied the SVD preprocessing step selecting and normalizing the 5 leading SVD components ('eigengenes' according to Alter, [12]) thus reducing the matrix from 7129X72 to 5X72. Clustering has improved after dimensional truncation, yet not all algorithms correctly cluster the samples. Note that only QC shows a substantial degree of consistency with the 'real' classification (Jaccard. = 0.707, Purity = 0.77 and Efficiency = 0.898, for discussion see Horn & Axel [13]).

3.2 COMPACT Tests of Yeast Cell Cycle Data

Next we test the performance of COMPACT for clustering of genes rather than samples. For this goal we explore the dataset of yeast cell cycle presented by Spellman et al. [6]. This dataset was used as a test-bed for various statistical and computational methods 14]. The expression levels of 798 genes were collected from 72 different

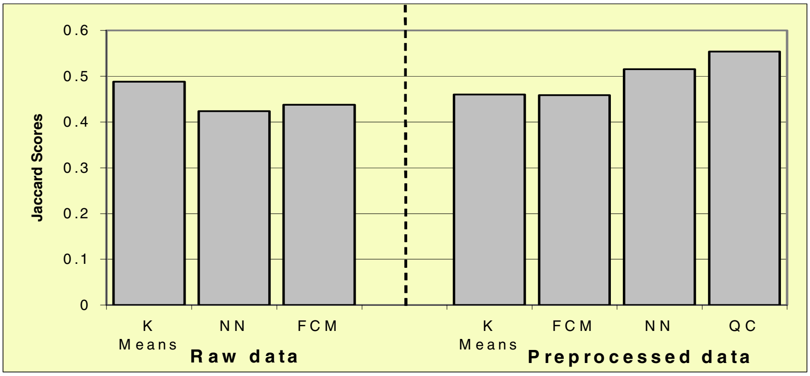


Fig. 4. Jaccard scores of the algorithms in the COMPACT based comparison for the Spellman dataset (shown are results for four clusters analysis)

Table 2. COMPACT based comparison to the Spellman dataset of Cell cycle in Yeast [6]

Method	Jaccard	Purity	Efficiency
Raw data			
K Means (5 clusters)	0.435	0.617	0.596
K Means (4 clusters)	0.488	0.64	0.673
Fuzzy C Means (5 clusters)	0.425	0.663	0.542
Fuzzy C Means (4 clusters)	0.438	0.458	0.912
Competitive Neural Network (4 clusters)	0.424	0.53	0.68
Quantum Clustering	NA	NA	NA
Preprocessing			
K means (5 clusters)	0.406	0.636	0.528
K means (4 clusters)	0.46	0.626	0.634
Fuzzy C means (5 clusters)	0.4	0.63	0.522
Fuzzy C means (4 clusters)	0.459	0.624	0.634
Competitive Neural Network (5 clusters)	0.33	0.55	0.458
Competitive Neural Network (4 clusters)	0.516	0.658	0.706
QC after SVD (σ =0.595)	0.554	0.664	0.77

conditions that reflect different time points in the yeast cell cycle. The task in this case is to cluster these 798 genes into five classes identified by Spellman et al. through functional annotations of individual genes.

We applied COMPACT both to ‘raw’ data and to SVD compressed data. In the latter case we selected two leading normalized SVD components ('eigensamples' according to Alter, [12]), thus reducing the matrix size from 798X72 to 798X2. All four clustering methods were tested as before. Once again the results obtained by the QC are moderately superior.

We have tested all methods for both 4 and 5 clusters (Table 2 and Figure 4). Interestingly enough, 4 clusters seem to be a better choice in all cases, although the 'real' classification defines 5 classes.

4 Discussion

In this paper we demonstrate how different clustering algorithms may lead to different results. The advantage of COMPACT is in allowing many algorithms to be viewed and evaluated in parallel on a common test set. Through COMPACT one can evaluate the impact of changing the algorithm or its parameters (e.g., sigma value in QC, number of iterations for the Competitive Neural Network, starting points of K-Means, Fuzzy C-Means and more). Being able to run a number of clustering algorithms, observe their results (quantitatively and graphically) and compare between them is beneficial for researchers using gene expression, proteomics, and other technologies that produce large datasets. We find it advisable to start with a problem that has a known classification (referred to as ‘real classification’) and use the statistical criteria (i.e., efficiency, purity and Jaccard score) to decide on the favorable clustering algorithm. For general research problems, where no known classification exists, the same statistical tools may be used to compare results of different clustering methods with one another. We presented here a comparative analysis of some well-known clustering methods with one relatively new method, QC. For the two datasets that we have explored, QC outperformed the other methods.

We have shown that dimensionality reduction improves the clustering quality. This observation is highly relevant when handling genomic data. Recall that for Affymetrix microarrays the number of genes tested reaches all known transcripts from the selected organism, producing 20,000-30,000 data points for a mammalian genome. Similarly, the application of the new SNP discovery chip produces a huge number of noisy data points in a single experiment. Besides its computational complexity, one of the major challenges when using massive data is to identify features and to filter out noise. Often handling such high dimensional noisy inputs can be a barrier. Hence it is important to develop more efficient and accurate tools to tackle these problems (see examples in [3], [4], [15], [16]). Thus, constructing a method that can significantly reduce data volume, and at the same time keep the important properties of that data, is obviously required.

COMPACT offers easy-to-use graphical controls for users to select and determine their own preferences, and graphical displays where the results can be presented or saved for later usage. It offers several clustering algorithms and allows the user to compare them to one another.

Although similar tools have already been proposed (e.g., [17], or [18]), the novelties of COMPACT are: (i) presenting an integrative, light package for clustering and visualization, (ii) integrating an efficient compression method and (iii) introducing the QC algorithm as part of the available clustering options.

The beginners will find this user-friendly tool with its graphical and textual displays useful in their data analysis. The experts will benefit from its flexibility and customizability that enables expanding the tool and modifying it for advanced, specialized applications.

Acknowledgment

We thank Inon Axel for helping in the initial stages of the development. We thank Noam Kaplan for critical reading the manuscript. We thank the ProtoNet team in the Hebrew University of Jerusalem for suggestions and helpful discussions. This work is partially supported by the BioSapiens NoE of the EU framework VI. R.V. awarded a fellowship by the SCCB, The Sudarsky Center for Computational Biology. His research is partially supported by the Israel Science Foundation.

Availability: COMPACT is available at http://www.protonet.cs.huji.ac.il/compact and at http://adios.tau.ac.il/compact . A detailed description of the application can be found on these websites.

References

1. Jain, A. K., Dubes R. C.: Algorithms for Clustering Data. Englewood Cliffs, NJ, Prentice Hall, Englewood Cliffs, NJ; 1988
2. Sharan, R., Maron-Katz A., Shamir, R.: CLICK and EXPANDER: a system for clustering and visualizing gene expression data. Bioinformatics. 2003, 19(14): 1787-99.
3. Eisen, M. B., Spellman, P. T., Brown P. O., Botstein D.: Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci U S A. 1998, 95(25): 14863-14868.
4. Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Coller H., Loh M.L., Downing J.R., Caligiuri M.A., Bloomfield C.D., Lander E.S.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science. 1999, 286: 531-537.
5. Cheng Y., Church G. M.: Biclustering of Expression Data. Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology; AAAI; 2000:93-103.
6. Spellman P.T., Sherlock G., Zhang M.Q., Iyer V.R., Anders K., Eisen M.B., Brown P.O., Botstein D., Futcher B. P. T.: Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol Biol Cell. 1998, 9(12): 3273-97.
7. Horn, D., Gottlieb A.: Algorithm for data clustering in pattern recognition problems based on quantum mechanics. Phys Rev Lett. 2002, 88(1): 018702.
8. Yeang C.H., Ramaswamy S., Tamayo P., Mukherjee S., Rifkin R.M., Angelo M., Reich M., Lander E., Mesirov J., Golub T. C. H., Ramaswamy S.: Molecular classification of multiple tumor types. Bioinformatics. 2001, 17 Suppl 1: S316-22.

9. Pan, W.: A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. Bioinformatics. 2002, 18(4): 546-54.
10. Mukherjee S., Tamayo P., Rogers S., Rifkin R., Engle A., Campbell C., Golub T.R., Mesirov J.P. S.: Estimating dataset size requirements for classifying DNA microarray data. J Comput Biol. 2003, 10(2): 119-42.
11. Pan, W.: A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. Bioinformatics. 2002, 18(4): 546-54.
12. Alter, O., Brown P. O, Botstein D.: Singular value decomposition for genome-wide expression data processing and modeling. Proc Natl Acad Sci U S A. 2000, 97: 10101-10106.
13. Horn, D., Axel I.: Novel clustering algorithm for microarray expression data in a truncated SVD space. Bioinformatics. 2003, 19(9): 1110-5.
14. Friedman, N., Linial M., Nachman I., Pe'er D.: Using Bayesian networks to analyze expression data. J Comput Biol. 2000, 7: 601-20.
15. Sasson, O., Linial N., Linial M.: The metric space of proteins-comparative study of clustering algorithms. Bioinformatics. 2002, 18 Suppl 1: S14-21.
16. Sasson O., Vaaknin A., Fleischer H., Portugaly E., Bilu Y., Linial N., Linial M.: ProtoNet: hierarchical classification of the protein space. Nucleic Acids Res. 2003, 31(1): 348-52.
17. The Eisen Lab software page [http://rana.lbl.gov/EisenSoftware.htm]
18. The R project for statistical computing [http://www.r-project.org/]