Gene expression

gcExplorer: interactive exploration of gene clusters
Theresa Scharl¹,* and Friedrich Leisch²
1Department of Biotechnology, Universität für Bodenkultur, Vienna, Austria and 2Department of Statistics, Ludwig-Maximilians-Universität, Munich, Germany

Received on October 15, 2008; revised on January 19, 2009; accepted on February 16, 2009
Advance Access publication February 18, 2009

Associate Editor: Trey Ideker

ABSTRACT

Summary: Cluster analysis plays an important role in the analysis of gene expression data since the early beginning of microarray studies and is routinely used to find groups of genes with common expression pattern. In order to make cluster analysis helpful for users, visualization of cluster solutions is of utmost importance. Here, we present the new R package gcExplorer for the interactive exploration of gene clusters. gcExplorer facilitates the interpretation of cluster results and allows to investigate extensive information about clusters.

Availability: The latest release of gcExplorer is always available at the Comprehensive R Archive Network CRAN: http://cran.R-project.org/package=gcExplorer. See the README file in the package for detailed installation instructions.

Supplementary information: Supplementary data are available at Bioinformatics online.

1 INTRODUCTION

Clusters of co-expressed genes can help to discover potentially co-regulated genes or association to conditions under investigation. However, the definition of gene clusters is not clear as genetic interactions are very complex. Microarray data are often very noisy and hence genes can easily end up in different clusters even if the expression profiles without noise are very similar. Therefore, the relationship between clusters plays an important role. Cluster methods with the ability to visualize the results [e.g. hierarchical clustering in (Eisen et al., 1998)] are preferred by many users. Visualization methods give an understanding of the relationships between segments of a partition and make it easier to interpret the overall results.

Neighborhood graphs (Leisch, 2006) can be used for visual assessment of the cluster structure of centroid-based cluster solutions. In a neighborhood graph each cluster is represented by a node. The similarity of two clusters is measured by the weighted percentage of data points that have one of the corresponding cluster centroids as closest and the other as second closest, respectively. Observations that have similar distances to both centroids get a larger weight than observations which are close to one and far away from the other. Note that as a percentage the cluster similarity is always between 0 and 1 [see, Leisch (2006) for details]. If we remove one of the two cluster centroids, all points contributing to the cluster similarity of the centroid pair would be reassigned to the remaining centroid. Thus, it can also be used as an indication which clusters are candidates for being merged. Similar concepts are used for creating topology representing networks (Martinez et al., 1993) or to define cluster silhouettes (Kaufman and Rousseeuw, 1990). In the graph, two clusters/nodes are connected if their similarity exceeds a user-specified threshold. The thickness of an edge between two clusters is proportional to their similarity. Neighborhood graphs can also be used to explore the number of clusters: too fine partitions show very high connectivity in the graph, while too coarse clusterings are identified if different expression profiles show up in one cluster (see below).

The main contribution of the new R (R Development Core Team, 2008) package gcExplorer is to scale up neighborhood graphs to the large number of clusters typically used for gene expression data. Instead of linear projections we now use non-linear layout algorithms allowing the display of very complex similarity structures. Gene clusters can be explored interactively using various display methods.

2 THE R PACKAGE

gcExplorer is based on infrastructure from R package flexclust (Leisch, 2006) which contains extensible implementations of K-centroids and QT-Clust (Heyer et al., 1999). However, arbitrary centroid-based cluster solutions using, for example, the standard functions kmeans or pam can also be used as input by converting them to flexclust objects. Bioconductor packages graph and Rgraphviz (Gentleman et al., 2005) are used for non-linear arrangement of the graph nodes. Rgraphviz is an interface to the open source software project GraphViz (http://www.graphviz.org).

gcExplorer offers several possibilities for the interactive exploration of gene clusters. There are different methods to include information about the clusters in the representation of nodes by using color coding defined by argument node.function. External information about pathways or functional groups (e.g. gene ontology terms) can be included in the call to gcExplorer. This allows to search for accumulation of functionally related genes in clusters. The association between clusters can be explored by using different cutoff values for the edges to be drawn. This can be used to find the appropriate number of clusters for a given problem.

The most important feature is that the data points in a given cluster can be explored interactively using panel functions. When clicking on the nodes of the neighborhood graph, the panel.function()
is executed for the observations in the corresponding cluster. Users can define arbitrary new panel functions, or use the ones already defined by us. Any R graphic can be used as panel function, the example below shows gene expression profiles. Another possibility are HTML tables of all genes in a cluster with links to databases.

3 EXAMPLE
Software usage is now demonstrated on Escherichia coli cultivation data (Dürrschmid et al., 2008) where stress response was measured during expression of the human recombinant protein SOD (human superoxide dismutase, hSOD). After preprocessing and filtering out genes that did not show any differential expression a dataset of 527 genes was clustered into 15 clusters using QT-Clust with a maximum radius of 2:

```r
R> library("gcExplorer"); data("hsod")
R> set.seed(1111)
R> cl1 <- qtclust(hsod, radius = 2,
+ save.data = TRUE)
```

We use color coding to distinguish small clusters (light) from larger clusters (dark), gene expression profiles are used as panel function:

```r
R> gcExplorer(cl1, theme = "blue",
+ legend.pos = "bottomright",
+ panel.function = gcProfile,
+ node.function = node.size)
```

The middle part of Figure 1 shows the corresponding neighborhood graph. Nodes connected by thick and dark edges indicate close relationship between the underlying clusters. The three outer plots and the table pop up when clicking on the nodes of the graph. The bottom left and right plots show that the genes in clusters 11 and 7 have similar expression patterns, with stress response being slightly stronger in cluster 7. Gene clusters with different expression pattern lie on the opposite end of the graph (e.g. cluster 3 in the top right plot). The HTML table in the top left corner can be created using `panel.function=gcTable`.

Due to space constraints only a small example is shown here, but `gcExplorer` is routinely used for more than 50 clusters. See the vignette of the R package or (Scharl and Leisch, 2008) for details and further examples. Static PDF versions of all graphics can be created using function `gcOffline`.

4 OUTLOOK
In the future the `gcExplorer` will be extended allowing small glyphs as node functions. Additionally, the user will be able to zoom into a subgraph of interest to have a more detailed view on the cluster structure and the underlying gene expression patterns.

Funding: Austrian Center of Biopharmaceutical Technology.
Conflict of Interest: none declared.

REFERENCES