Bioinformatics Advance Access published February 26, 2004

Bioinformatics © Oxford University Press 2004; all rights reserved.

GeneCluster 2.0: An advanced toolset for bioarray analysis

M. Reich, K. Ohm, M. Angelo, P. Tamayo, J.P. Mesirov
Center for Genome Research, Massachusetts Institute of Technology
Cambridge, MA 02141

ABSTRACT

Summary: GeneCluster 2.0 is a software package for analyzing gene expression and other bioarray data, giving users a variety of methods to build and evaluate class predictors, visualize marker lists, cluster data, and validate results. GeneCluster 2.0 greatly expands the data analysis capabilities of GeneCluster 1.0 by adding classification, class discovery, and permutation test methods. It includes algorithms for building and testing supervised models using weighted voting (WV) and k-nearest neighbors (kNN) algorithms, a module for systematically finding and evaluating clustering via self-organizing maps (SOM), and modules for marker gene selection and heat map visualization that allow users to view and sort samples and genes by many criteria. GeneCluster 2.0 is a standalone Java application and runs on any platform that supports the Java Runtime Environment version 1.3.1 or greater.

Availability:
http://www.broad.mit.edu/cancer/software

Contact: gc-info@broad.mit.edu

GeneCluster 1.0, released in June of 1999, implemented the Self-Organizing Maps (SOM) algorithm popularized in Tamayo et al. 1999 as well as various preprocessing methodologies that have since become standard in microarray analysis. Since its public release, GeneCluster 1.0 has been downloaded by over 3000 users. In June 2002, a greatly enhanced version, GeneCluster 2.0, was released. It enhances the SOM clustering capability of GeneCluster 1.0 and adds modules for feature selection, predictive modeling, validation, and annotated heat-map visualization.

To enter data into GeneCluster 2.0, users provide a file containing the gene intensity values for each sample scanned along with associated annotations. Any required image processing steps such as background subtraction and dye correction should already be done.

Once loaded, an experiment file can be filtered and normalized with a variety of algorithms. These include thresholding, scaling, normalizing to a given mean and variance, fold-change, and exclusion of high and low scoring features. Users can also randomize a dataset by bootstrap sampling columns from the data set with replacement. Additional available transformations are transposition and linear and logarithmic scaling.

Unsupervised learning, or clustering, is implemented by a SOM algorithm that has been extended to allow users to perform batch runs varying the number of clusters and cluster geometries. Results can be viewed in a visualizer that displays cluster profiles and relevant cluster member information.

GeneCluster 2.0 adds supervised learning methods (Golub et al. 1999, Slonim et al. 2000), in which a user trains an algorithm with labeled samples to obtain a model that can predict the class of an unknown sample. The user can choose between a weighted voting (WV) and a K-nearest neighbors (kNN) algorithm, as well as the number of features in the model and how those features are chosen.

GeneCluster 2.0 offers an extensive set of tools and methods to perform marker or feature selection. This module allows users to find all genes in the neighborhood of a single gene or the genes that most closely fit an expression profile. Distance metrics include Euclidean, Pearson correlation, and Manhattan distance, and statistical measures of correlation include the signal-to-noise score and t-test. Markers can be visualized as an interactive heat map (color-gram), described below, that allows users to view graphically the expression levels for a set of markers.

To determine the statistical significance of a marker, GeneCluster 2.0 provides a permutation test. This test compares the density of neighborhoods of genes correlated with a valid class distinction profile (for example, tumor vs. normal) to neighborhoods of profiles of randomly assigned distinctions (see Golub et al. (1999) and Slonim et al. (2000) for details). The user can set the significance level and number of permutations, or can use default values.

Once a predictive model is built, its accuracy can be measured through either leave-one-out or n-fold cross-validation, or via random train/test splits. The predictor's accuracy is given as the average across all the split-train-test cycles. GeneCluster 2.0 also provides a confusion matrix, which gives users a tabular view of the number of correct versus incorrect class predictions.

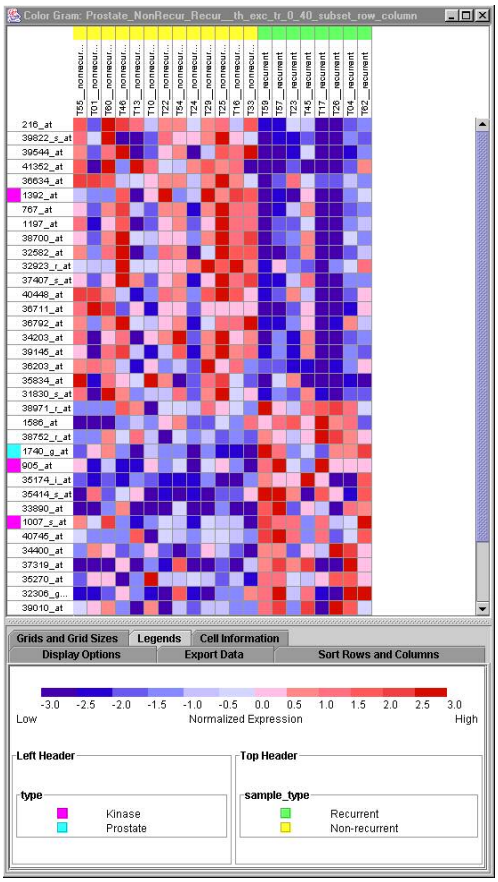

Figure 1. Colorgram Browser window.
Genes appear in a column to the left-hand side, and samples appear across the top. Colors in the heat map represent the up- or down-regulation of a gene in a given sample. Colors to the outside of the gene and sample labels represent user-defined annotation.

Complementing the analysis capabilities of GeneCluster 2.0 is its colorgram browser, which displays gene expression values in a two-dimensional matrix using a heat-map format. Users can enter their own annotation into the colorgram browser, via color-coding (see Figure 1), so that they can sort genes or samples and view the results visually. For example, a user can color-code all genes that code for nuclear proteins, give another color to those that code for cytoplasmic proteins, and another to those for extracellular proteins, and then sort the dataset by these categories so they can view the behavior of a custom-chosen class of genes.

GeneCluster 2.0 is implemented in Java and will run on any platform that supports the Java Runtime Engine 1.3.1 or greater. The installation includes sample datasets from the Cancer Genomics Program at the Whitehead Institute Center for Genome Research and a reference manual that describes the user interface, statistical methods used, and sample analyses. The software was made available in June, 2002, and has been downloaded by over 2000 users since its release. GeneCluster 2.0 is free for research purposes.

ACKNOWLEDGEMENTS

The authors wish to thank the other members of the Cancer Genomics Program at the Center for Genome Research / Massachusetts Institute of Technology: Jean-Philippe Brunet, Phil Febbo, Mike Gillette, Todd Golub, Christine Ladd-Acosta, D.R. Mani, Stefano Monti, Sayan Mukherjee, Nick Patterson, Sridhar Ramaswamy, Ken Ross, Donna Slonim, and Aravind Subramanian.

REFERENCES

Golub,T.R., Slonim,D.K., Tamayo,P., Huard,C., Gaasenbeek,M., Mesirov,J.P., Coller,H., Loh,M., Downing,J.R., Caligiuri,M.A., Bloomfield,C.D., Lander,E.S. (1999) Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression, Science 286, 531-537.

Slonim,D., Tamayo,P., Mesirov,J.P., Golub,T., Lander,E. (2000). Class prediction and discovery using gene expression data. Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB), Universal Academy Press, pp. 263-272.

Tamayo,P., Slonim,D., Mesirov,J., Zhu,Q., Kitareewan,S., Dmitrovski,E., Lander,E., Golub,T. (1999) Interpreting patterns of gene expression with self-organizing maps: methods and applications to hematopoietic differentiation, Proc. Natl. Acad. Sci. USA, 96, 2907-2912.