Conditional Clustering of Temporal Expression Profiles
User Manual
Version 1.0, released September, 2007

Ling Wang (wangling@bu.edu)
Paola Sebastiani (sebas@bu.edu)
Department of Biostatistics
Boston University School of Public Health

I. Introduction

Welcome to use the Bayesian conditional clustering program! This program is designed to perform clustering analysis of short temporal gene expression data in several experimental conditions.

This program is written in the software R (www.r-project.org), which is freely available, and can be downloaded at the CRAN site by following the links on the R homepage.

II. Citing the conditional clustering program

To cite the clustering method for this conditional clustering program, please make reference to the paper:
Wang, L., Montano, M., Rarick, M. and Sebastiani, P. Conditional clustering of temporal expression profiles. Submitted

III. Preliminaries

Currently this conditional clustering program runs on Windows operating system. You can download the zipped folder of the program to a directory that you want to work at.

Data format: the function that performs the clustering automatically takes the data for clustering. Preferably, the dataset should be tab-delimited txt files, in which the first column contains the gene names, the second column contains their accession numbers or descriptions. The remaining columns, in their orders, are the gene expression data (contain multiple columns), the time of the experiment (contain multiple columns) and the experimental condition. The dataset are assumed to have been processed so that each row contains only the expressions of one gene at one experimental condition.

IV. Clustering procedure

1. Prepare the data in the right format.
2. Create a directory under the directory of the conditional clustering program where you want to move the data and store the results.
3. Modify the provided 'sample R code.txt' to your needs, i.e., modify the working directory to where the conditional clustering program is and decide the parameters for the clustering function.

4. Open an R console, copy and paste, or source, the R code you modified in R. The function will automatically perform the conditional clustering, search for the optimal polynomial order and return a list of figures and results.

V. Parameters in the function iterative.cluster():

```
results.main.dir: the main directory in which the results are to be
saved.
data.all: the data.
name.col: column index of the gene names
id.col: column index of the gene names, descriptions and accession
numbers, etc.
data.col: column index of the gene expression data.
time.col: column index of the time of the experiment.
cs.cov.col: column index cross-sectional covariates. Useful for
clustering longitudinal data.
int.time.cs.cov.col: column index of the interaction between time
and cross-sectional covariates. Useful for clustering longitudinal
data.
lgtnl.cov.col: column index of the time-varying covariates. Useful
for clustering longitudinal data.
n.lgtnl.cov: number of time-varying covariates.
cat.var.col: column index of the categorical covariate for
longitudinal data, or column index of the experimental condition
for gene expression data
alpha.1: the hyper-parameter alpha.1
alpha.2: the hyper-parameter alpha.2
eta<-1: the hyper-parameter eta
normalize: whether to normalize the data
transform: what type of transformation needs to be done for data
type.distance: type of distance.
intercept: whether to include the intercept in the model. When the
data is normalized and log transformed, there should be no
intercept in the model
```

Default settings of some of the parameters:

```
alpha.1=2, eta=1, normalize=T, transform=1, filter=c(1,1),
type.distance=1, intercept=T
```

Options for some parameters:

```
normalize: T=normalize as the ratios of the first value in each
time series, F=don't normalize
transform: 0=None, 1=natural log, 2=square, 3=cubic, 4=square root,
5=cubic root
type.distance: the type of distance: 1=Euclidean,2=Correlation
intercept: whether to include intercept in the model
```

VI. Generated results

1. The number of unique and common clusters, the best polynomial order are stored in the file "results.txt" in the main results directory.
2. The folder 'initial-results' contains the conditional clustering results before the iterative procedure. There are data for each cluster, series and profile plots for

each cluster. The file ‘results.txt’ in this folder contains the fit statistics for the initial analysis.

3. If there are clusters of GCP, the folder ‘common’ contains results of the clusters of GCP. These are data for each cluster of GCP, profile and series plot, and a heatmap for each cluster.
4. If there are clusters of GUP, the folder ‘unique’ contains results of the clusters of GUP. These are data for each cluster of GUP, profile and series plot, and a heatmap for each cluster.