BIOINFORMATICS MSc
PROBABILITY AND STATISTICS

SPLUS EXERCISE SHEET 4

Things to download:

SPLUS script
`http://stats.ma.ic.ac.uk/~das01/BioinformaticsMSc/SPLUS/sheet04.ssc`

`R` script
`http://stats.ma.ic.ac.uk/~das01/BioinformaticsMSc/SPLUS/sheet04.R`

Data:
`http://stats.ma.ic.ac.uk/~das01/BioinformaticsMSc/Data/Cluster.zip`

You need to download this zipped file, and unzip it to a suitable directory on your own hard-disk. The SPLUS script assumes that you have unzipped the file to some directory, say

h:\Cluster

There should be eight directories **CWRU-A,...,CWRU-H**. Each directory contains a range of microarray experiments (in *Anopheles* mosquitos). The data are either comparative (different experiments) or time course (data recorded in the same experiment but over a number of hours/days).

First, inspect the file formats in the different directories; these files are raw output files derived from the microarray imaging software. The files are quite informative, containing much more information than we will utilize. Then **attach** a new SPLUS "chapter" in which to store the SPLUS data as follows. Using the pull-down menus

File -> Chapters -> Attach/Create Chapter

Then in the **Chapter Folder** dialog box, type **h:\Cluster**, change **Position** to 1 by clicking on the **Down** arrow, type **Cluster** in the label box. This will place all your subsequent SPLUS defined objects into this chapter, which you can "detach" at the end of the session, without losing the data, by typing `detach(1)` or `detach("Cluster")`

NOTE: In `R`, the procedure is somewhat different; the `setwd` function sets the working directory, In the code, the `read.table` function is used to input data from the flat text files, and the `list.files` function is used to produce the required directory listings.

1. PRINCIPAL COMPONENTS ANALYSIS

Your objective is to use principal components to carry out a data reduction exercise. In this case we have a large number of genes (**observations)** (more than 10000 in most cases) and often seven time points. However, there is little evidence of up- or down-regulation at all time points, so principal components may be able to summarize the data in a lower dimension by looking at linear combinations of the time points (here acting as **predictors**). In addition the PCA may allow those genes that are regulated (rather than having no differential expression).

The code to carry out the PCA, and produce the required plots, is given in the scripts on the website. Use the **Help** menus to found out what each command does.

EXERCISE: Implement PCA for the data for the different experiments. Note that the file formats in different directories are sometimes slightly different, so you may need to adjust the code that reads in the data.

2. CLUSTERING OF GENE EXPRESSION MEASUREMENTS

The objective of cluster analysis is to identify those subsets of genes that have common patterns of expression and regulation. The key splus commands that implement **hierarchical** clustering, and produce the relevant plots are

- the **dist** command that computes the similarity matrix
- the **hclust** command that produces the hierarchical clustering object
- the **plclust** command that produces the **dendrogram** for the clustering
- the **cutree** command that returns the clustering labels, if a specific number of clusters is chosen

For a given selected number of clusters, the data in each cluster can be plotted out using the **plot** command. Using such a plot, the integrity of each cluster can be verified.

EXERCISES:

(1) Use the dendrogram to try to identify how many clusters are present in the data, and to choose **ncuts** appropriately

(2) Experiment with different clustering specifications for **hclust**. For example, try different clustering **methods** such as

```
plclust(hclust(dist(gene.courses.F,metric="euclidean"),method="ave"))

plclust(hclust(dist(gene.courses.F,metric="maximum"),method="single"))
```

Examine how the clusters change as the parameters are varied.