Statistical data analysis

Learning from Gene Expression Data

A biologically-directed final assignment

Fall 2019/2020

Introduction

You are provided with gene expression data. The dataset gives expression measurements of a large pool of genes over a reasonable sample of individuals. The individuals are split into several well-defined groups such as healthy controls, stationary patients, chronically diseased patients and recovered patients. The goal is to learn from the gene expression data, i.e., to understand relationships among activity of the individual genes and the health condition of an individual (the group, in general it is referred to as phenotype).

1 Background

Hereditary information encoding the development and functioning of an organism is stored in a macromolecule called *deoxyribonucleic acid* (DNA). The information is stored as a sequence of nucleotides also called *bases*, namely adenine, cytosine, guanine and thymine. The information carried by DNA is held in the sequence of distinguishable regions of DNA called *genes*. *Gene expression* (GE) is the cellular process by which information from a gene is used in the synthesis of a functional product, most often a protein. A gene is first transcribed into a *ribonucleic acid* (RNA) that serves for passing the genetic instructions from the cell nucleus to the cytoplasm where the RNA is *translated* into a *protein*. Proteins already perform a large scale of biological functions. To sum up, the genes expressed into proteins specify the structure and function of the biological system. The other genes remain silent. The above-described flow of genetic information is referred to as *the central dogma of molecular biology*.

Cell functioning is based on which genes get expressed under what circumstances. The expression is influenced by many factors, including external signals coming through the membrane from outside the cell. In this way, a cell responds to external stimuli and produces specific proteins under specific circumstances. The simplified view suggests that cell can be modelled as a feedforward linear system that proceeds from DNA towards proteins and phenotype in the end. In fact it should rather be

modelled as an extremely complex dynamic and non-linear regulatory network containing frequent feedback loops. Nevertheless, even simple statistical gene expression models may help to understand molecular processes accompanying disease onset and recovery.

High-throughput technologies, like DNA microarrays and RNA-Seq for transcriptome profiling, examine the gene expression level in a given cell population simultaneously, i.e., they conduct a huge number of measurements in parallel. The data usually show an inconveniently low ratio of samples (individuals, biological situations, typically tens of them) against variables (genes, typically tens of thousands of them). Datasets are often noisy and they contain a great part of variables irrelevant in the context under consideration. Even simple statistical inference could be challenging in this context. To exemplify, robust statistical testing must consider multiple testing correction at least.

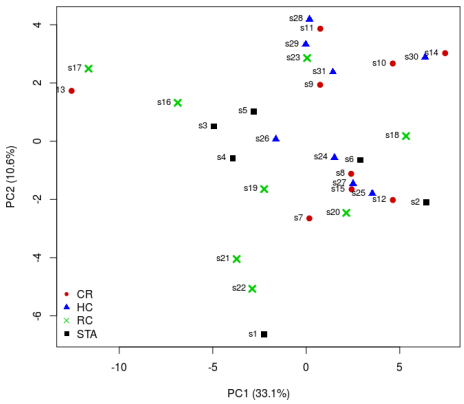
2 Data

You are given a data matrix that contains 10,000 genes measured in 31 different samples, the file is *ge.csv*. Each line starts with a unique gene name, then 31 normalized expression values follow. Each value gives expression level of the given gene in the given sample. The phenotype, i.e., the assignment between the sample id and a group is provided in the file *annot.csv*. HC stands for healthy controls, STA for stationary patients, CR for chronically diseased patients and RC for recovered patients.

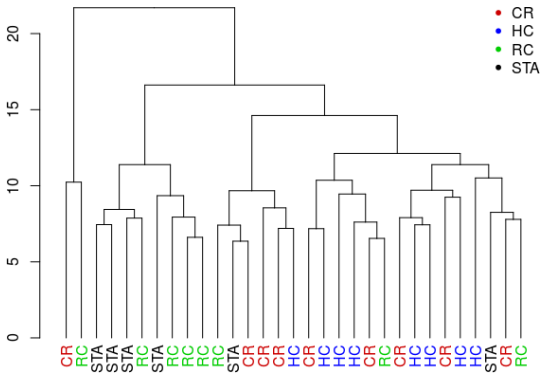
3 Tasks

The assignment can be decomposed into the following well-defined tasks:

1. **Exploratory analysis of the gene expression dataset**. Load the dataset, carry out dimensionality reduction and clustering. The main goal is to see whether the phenotype manifests in samples' gene expression profile clusters. Show the explanatory plots and explain the relationship between the observed sample distribution and the expected distribution. Let us assume that the individuals in the same group should have similar gene expression profiles. An expected result could look similar to following (run on different data, only structure matters):

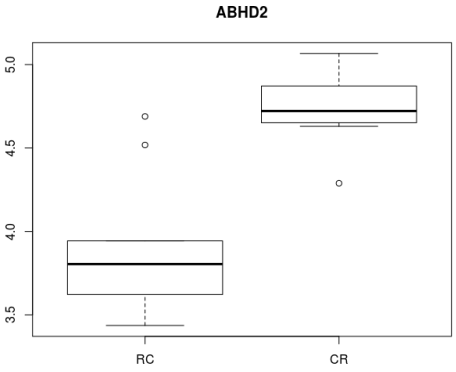


Principal component analysis

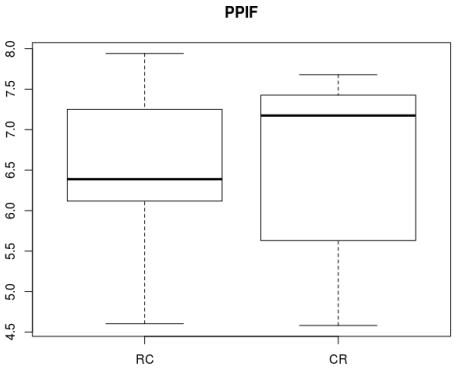


Hierarchical clustering, dendrogram

2. **Differential expression.** Find all the genes that are differentially expressed (DE), i.e., the genes whose expression differs under different conditions. In our case, different conditions mean different groups of individuals. Work with a selected pair of groups (e.g., recovered vs chronically diseased patients) as well as with the whole set of groups. Test the assumptions of the used statistical methods (t-test, ANOVA), do not forget to do multiple testing correction. An example of potentially differentially expressed gene ABHD2 is shown below in the left pane. In the right pane, there is an example of PPIF gene whose expression is not likely to change between the groups RC and CR.



Boxplot for a DE gene



Boxplot for a non DE gene

3. **Phenotype predictive model.** Create a model that predicts the phenotype (group) based on gene expression profile. Definitely consider linear discriminant analysis and logistic regression, but you can try any other model too. Evaluate and compare the performance of the models, employ cross-validation to get an unbiased estimate. Utilize feature selection to simplify the models and improve their performance.

4 Submission and evaluation

Submit your solution to the upload system. Submit only the rmarkdown file named geneExpression$YOURFELUSERNAME.Rmd. This file should be considered as a report containing a definition of the task, description of your implementation details, graphical outcomes and your **detailed answers** to the required tasks.

You can obtain up to 15 points for this assignment, 5 points per each subtask. Approximately 60% will be given for the concept of the solution (selection of statistical methods and their correct application, depth of the solution), 30% for the answers that summarize your solution in the individual subtasks (interpretation of your results and explanation of their practical impact in natural language), 10% for formal issues (clarity of the code, comments, readability of the rmarkdown as a whole).