goProfiles: an \textit{R} package for the Statistical Analysis of Functional Profiles

Alex Sánchez, Jordi Ocaña and Miquel Salicrú

April 30, 2008

1 Introduction

This document presents an introduction to the use of \textit{goProfiles}, an \textit{R} package for the analysis of lists of genes based on their projection at a given level of the Gene Ontology following the methodology developed by [2] [?].

2 Requirements

To use the package one must have \textit{R} (2.7 or greater) installed. Bioconductor 2.2 or greater is also needed.

3 Quick start

For the inpatient user the following lines provide a quick and simple example on the use of the package to explore and compare two experimental datasets obtained from two prostate cancer experiments ([4][1]). Detailed explanations about the computations can be found in the following chapters and in the package help.

The analysis proceeds as follows:

- First a dataset is loaded into memory. This dataset contains several lists of genes, from two different studies, selected as being differentially expressed in prostate cancer. \texttt{help(prostateIds)} will provide extra information about the lists of genes.

  \begin{verbatim}
  > require(goProfiles)
  > data(prostateIds)
  \end{verbatim}

- Next a functional profile is build for each list. For simplicity it is build for the MF ontology only.

  \begin{verbatim}
  > welsh.MF <- basicProfile(welsh01EntrezIDs, onto = "MF",
  + level = 2, orgPackage = "org.Hs.eg.db")
  > singh.MF <- basicProfile(singh01EntrezIDs, onto = "MF",
  + level = 2, orgPackage = "org.Hs.eg.db")
  > welsh.singh.MF <- mergeProfilesLists(welsh.MF,
  + singh.MF, profNames = c("Welsh", "Singh"))
  > printProfiles(welsh.singh.MF, percentage = TRUE)
  \end{verbatim}

  Functional Profile
  ================
  [1] "MF ontology"
  Description      GOID Welsh Singh
  7 antioxidant activity GO:0016209 0.3 0.4
  6 auxiliary transport prote... GO:0015457 0.5 0.7
Welsh (2001). Prostate cancer data. MF ontology

Figure 1: Basic profile for a dataset at the second level of the MF ontology

Changing the parameter onto to 'ANY' instead of 'MF' will compute profiles for the three ontologies.

> welsh <- basicProfile(welsh01EntrezIDs, onto = ”ANY”, + level = 2, orgPackage = ”org.Hs.eg.db”)

• A visual comparison of profiles can be useful to give hints about the difference between them.
• Finally a numerical comparison of both profiles is performed and its summary is printed. Notice that the comparison rebuilds the profiles, that is the input for the computation are the two lists of genes not the profiles.
> plotProfiles(welsh.singh.MF, percentage = T, aTitle = "Welsh vs Singh", + legend = T)

Welsh vs Singh. MF ontology

Figure 2: Comparison between two profiles at the second level of the MF ontology
> compared.welsh.singh.01.MF <- compareGeneLists(welsh01EntrezIDs, + singh01EntrezIDs, onto = "MF", level = 2, + orgPackage = "org.Hs.eg.db")
> print(compSummary(compared.welsh.singh.01.MF))

```
Sqr.Euc.Dist StdErr  pValue  0.95CI.low  0.95CI.up
0.002416  0.001363  0.075000 -0.000255  0.005087
```

• The reason to compare two lists similar studies may be to combine them. In this case what is more meaningful to do is an equivalence test, instead of a difference test as above.

> eqws01 <- equivalentGOProfiles(compared.welsh.singh.01.MF, + simplify = FALSE)
> print(es <- equivSummary(eqws01))

```
Sqr.Euc.Dist StdErr  pValue CI.up
0.002416  0.001363  0.000000  0.004658
```

4 User guide

This introduction has simply shown the possibilities of the program. A more complete user guide can be found at the program’s web site:


References


