PCpheno: assessing the role cellular organizational units in determining phenotype

Nolwenn Le Meur and Robert Gentleman

October 17, 2016

1 Introduction

We propose computational methods and statistical paradigms to explore the relationships between phenotypic data and cellular organizational units, such as multi-protein complexes or pathways. Indeed, while proteins are often the primary unit used by cells to carry out the many different functions that the cell requires for life, they seldom accomplish important tasks alone, but rather assemble into organizational units. Recent studies suggest that some control of phenotype can be usefully attributed to multi-protein complexes rather than genes (**??**) and hence may help provide elucidation of the underlying roles or mechanisms that directly control changes in phenotype.

2 Data sources

In this package, we currently present yeast phenotypic datasets and use of the yeast cellular organizational units defined in the Bioconductor package *ScISI* package and the KEGG pathways listed in the *org.Sc.sgd.db* package (formerly the *YEAST* package).

Nevertheless our methods can easily be applied to other species and other estimates of organizational units within the genome, or proteome, and in no way rely on the particular choices we have made here.

2.1 Phenotypic data

We currently propose 7 Yeast phenotypic datasets, downloaded from the the literature and the *Saccharomyces Genome Database* (http://www.yeastgenome.org/).

- **?** collection of single gene-deletion mutants under 6 different experimental conditions.
- **?** collection of single gene-deletion mutants under 21 different experimental conditions.
- **?** collection of haploinsufficient genes.
- **?** network of genetic interactions.
- **?** collection single gene-deletion mutants under 5 different experimental conditions.
- **?** collection of overexpression 600 C-terminal tagged integral membrane under 4 different experimental conditions.
- **?** list of phenotypes and associated genes from several published experiments.

While our approach focuses on understanding the functional roles that underly phenotypic changes when manipulating single genes, we hope that these methods will also form the basis for the analysis of more complex gene manipulation experiments.

To illustrate this vignette, we will use the data by **?**.

```
> library(PCpheno)
> data(DudleyPhenoM)
> ##Number of genes sensitive at each condition
> colSums(DudleyPhenoM)
```

```
 benomyl    CaCl2      CAD     Caff cyclohex      DTT     EtOH    FeLim
      34      180       83      208      164        5       75       35
      HU   HygroB   lowPO4      MPA     NaCl    Paraq      pH3      rap
      87      264       34       11       57       36       16      119
    Sorb       UV    YPGal    YPGly    YPLac   YPRaff
       8       32       41      206      159       31
```

```
> ##Retrieve the name of the sensitive genes in each condition
> DudleyPhenoL <- apply(DudleyPhenoM,2,function(x) names(which(x==1)))
> DudleyPhenoL[1]
```

```
$benomyl
 [1] "YLR085C" "YOR035C" "YPL241C" "YBR231C" "YPL129W" "YDR388W" "YER007W"
 [8] "YDR120C" "YER083C" "YDR138W" "YHL020C" "YHR012W" "YOR139C" "YJL179W"
[15] "YDR195W" "YBL031W" "YNL148C" "YGL086W" "YJL030W" "YJR074W" "YDR028C"
[22] "YJR053W" "YLR447C" "YCR063W" "YNR052C" "YER110C" "YDL020C" "YGR162W"
[29] "YMR055C" "YLR399C" "YLR244C" "YLR338W" "YLR304C" "YDR532C"
```

2.2 Cellular organizational units

As previously mentioned, here we are interested in yeast datasets and the yeast cellular organizational units defined in the Bioconductor package *ScISI* package and the KEGG pathways relevant to the yeast genome and available in the *YEAST* annotation package. However our methods can easily be applied to other species and other estimates of organizational units within the genome, or proteome.

The cellular organizational units should be represented as an adjacency matrix. The row names are the gene names and the column names the cellular organizational units. A 1 means that this particular gene belongs to this particular organizational units. Below is an example of yeast KEGG pathways. The *org.Sc.sgd.db* annotation package contains KEGG pathways annotation for the yeast genes and the `PWAmat`, available in the *annotate* package, allows to build the adjacency matrix.

```
> library(org.Sc.sgd.db) ## new YEAST annotation package
> ##library(annotate)
> KeggMat <- PWAmat("org.Sc.sgd")
> KeggMat[1:5, 1:5]

        00250 00330 00910 01100 00650
YAL062W     1     1     1     1     0
YJL130C     1     0     0     1     0
YJR109C     1     0     0     1     0
YKL106W     1     1     0     1     0
YKL104C     1     0     0     1     0
```

To build such interactome for a particular species, one should first have an annotation package for its species of interest. For instance, one can create this annotation package using the Bioconductor package *AnnBuilder* which retrieves, among other annotation, the KEGG pathways associated with the genome of interest. For protein complexes, it might be slightly more complicated but one can use the GO categories that refer to complexes and

create a similar binary matrix. In the case of the yeast genome, we use the interactome available in the Bioconductor package *ScISI*.

The *ScISI* package or *In Silico Interactome for Saccharomyces cerevisiae* provides an interactome built for computational experimentation. The `ScISI` is binary incidence matrix where the rows are indexed by the gene locus names and the columns are indexed by the identification codes for the protein complexes based on the repository from where they are obtained. This interactome is currently built from the Intact, Gene Ontology and Mips curated databases, and estimated protein complexes from the *apComplex* package. In this vignette, we will make use of a subset of the `ScISI` interactome, the `ScISIC` data, that only contains the data from the curated databases.

```
> library(ScISI)
> data(ScISIC)
> ScISIC[1:5, 1:5]
```

```
            EBI-913756 EBI-876785 EBI-852570 EBI-866976 EBI-1180400
EBI-913756           0          0          0          0           0
EBI-876785           0          0          0          0           0
EBI-852570           0          0          0          0           0
EBI-866976           0          0          0          0           0
EBI-1180400          0          0          0          0           0
```

3 Computational and Statistical Methods

In order to test for association between 2 datasets or 2 phenomenon, one has to define a null hypothesis. In our case, our null hypothesis is that there is no association between the collection of genes that induce the phenotypic change and the organizational units (*e.g.*, multi-protein complexes, pathways). To test this hypothesis we consider a multi-faceted approach.

First, for any level of organization, we use a hypothesis test designed to determine whether there is an effect that can be attributed to that specific groupings of genes, without testing which cellular organizational units are involved. Then, if we reject the null hypothesis of no association between the collection of genes that induce the phenotypic change and the organizational units, the next step is to identify those specific organizational units.

3.1 Global testing

We currently have devised two different methods of performing the omnibus test. One test is based on density estimation (**?**) and provides valuable visual information, but for which we do not have an explicit P-value. The second approach is based on the permutation of graphs (**?**) and while it provides an explicit P-value, it provides little insight into the reasons for rejecting, or not, the hypothesis. See below for more details.

3.1.1 Density Estimation

For each cellular organizational unit, we compute the proportion of genes that affect the phenotype. We then compute the smoothed histogram of the proportions and compare it to a reference distribution. Our reference distribution is obtained by randomly permuting 1,000 times the gene labels for the interactome and computing, for each permutation, the new (simulated) proportions of genes that affects the phenotype and the associated smoothed histograms.

As example, we test whether the genes sensitive to paraquat in the **?** experiment are randomly distributed among multi-protein complexes.

```
> perm <- 10
> paraquat <- DudleyPhenoL[["Paraq"]]
> parDensity <- densityEstimate(genename=paraquat, interactome=ScISIC, perm=perm)
```

Then, we can visualize the result of this test using the `plot` function.

3.1.2 Graph Theory

The graph theory procedure is based on the permutation of graphs proposed by **?**. Two distinct graphs, $G_i = (V, E_i)$, $i = 1, 2$, are formed. The nodes, V, are in our case the *S. cerevisiae* genes, and they are common to both graphs. In one graph G_1 two proteins have an edge between them if, and only if, they are co-members of one, or more, cellular organizational units. In the second graph G_2 edges are created between all proteins that are associated with a phenotype of interest, so that if there are k genes associated with the phenotype of interest then there is $k(k-1)/2$ edges. We exclude self-loops in both graphs. We then compute the intersection of these two graphs and count the edges in common. To test whether the number of edges in the third graph is unexpectedly large, a permutation analysis is performed. A

```
> plot(parDensity, main="Effect of paraquat on S. cerevisiae genes")
```

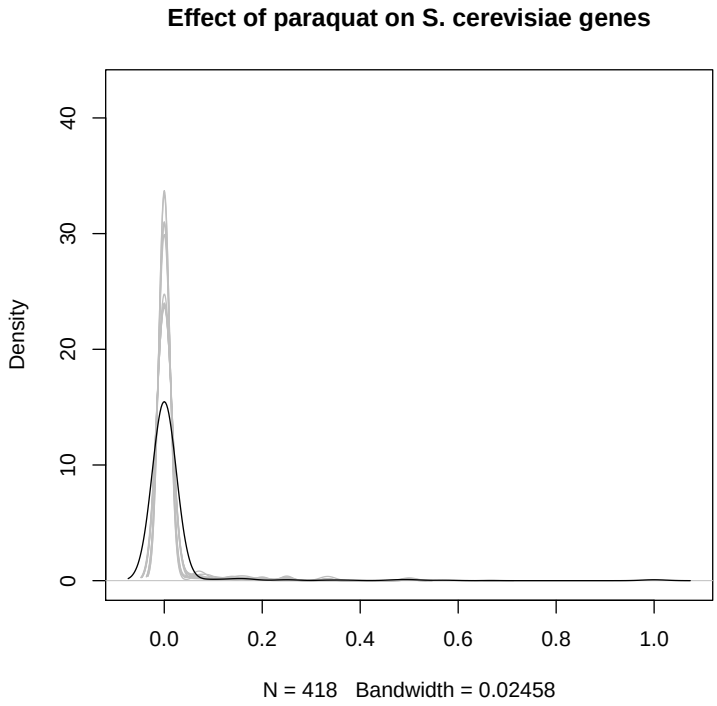

Figure 1: **Genes sensitive to paraquat are not well represented in our interactome.** A high frequency of protein complexes have zero "paraquat-sensitive" genes. Grey lines represent the permutation data and the black line is the observed data.

reference distribution is computed by permuting n times the labels on either G_1 or G_2 and counting the number of edges in common obtained. A p-value can be obtained by comparing the observed test statistic to the observed distribution of the counts of intersecting edges from the permutations.

```
> parGraph <- graphTheory(genename=paraquat, interactome=ScISIC, perm=perm)
```

Then, we can visualize the result of this test using the `plot` function.

```
> plot(parGraph, main="Effect of paraquat S. cerevisiae genes")
```

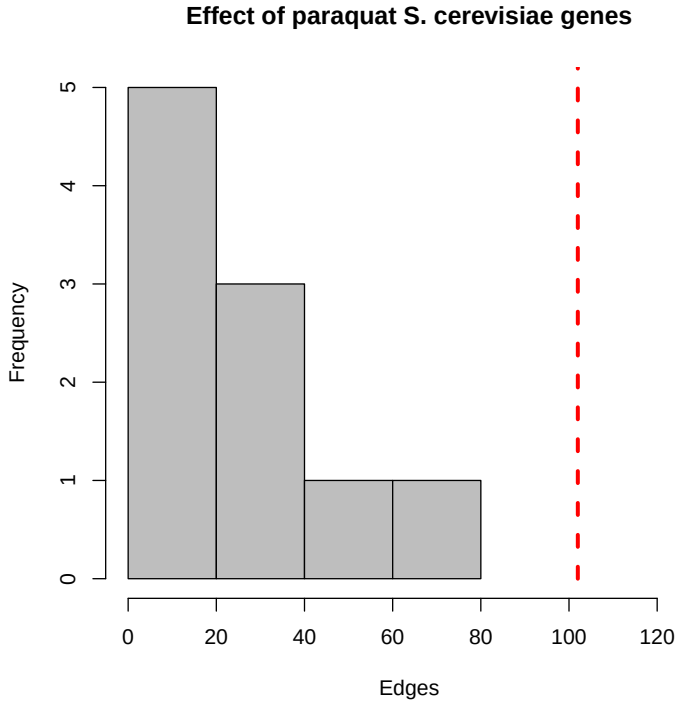

Figure 2: **Genes sensitive to paraquat are not randomly distributed in our interactome.** The "paraquat-sensitive" genes gather among complexes more than expected by chance. the grey histogram represent the observed distributions and the red dashed line the observed data.

3.2 Hypergeometric Test

A Hypergeometric test can be used to assess whether a cellular organizational unit contains more genes that affect the phenotype than expected by chance. We rank the multi-protein complexes by their p-value and classify a complex as being associated with the phenotype if the Hypergeometric p-value was less than a threshold, *e.g.,* 0.01. The Hypergeometric test is the equivalent of Fisher's exact test for two-by-two tables and the function repot the p-value, expected values and odds ratio for all tests.

```
> params <- new("CoHyperGParams",
+               geneIds=paraquat,
+               universeGeneIds=rownames(ScISIC),
+               annotation="org.Sc.sgd",
+               categoryName="ScISIC",
+               pvalueCutoff=0.01,
+               testDirection="over")
> paraquat.complex <- hyperGTest(params)
```

We can display the results using the `summary` function (Note that for visualization purposes we only show the first 6 columns).

```
> summary(paraquat.complex)[,1:6]
```

```
          ID       Pvalue OddsRatio   ExpCount Count Size
1   MIPS-220 3.082381e-11 121.68750 0.15941296     7   15
2 GO:0016469 7.909231e-08  43.04444 0.25506073     6   24
3 GO:0000220 3.239645e-07 153.09804 0.07439271     4    7
4 GO:0000814 1.035862e-06       Inf 0.03188259     3    3
5 GO:0000221 5.605086e-05  65.00000 0.08502024     3    8
6 MIPS-90.30 1.076206e-04       Inf 0.02125506     2    2
7 GO:0000813 6.374648e-04 102.78947 0.04251012     2    4
8 GO:0000815 6.374648e-04 102.78947 0.04251012     2    4
```

Finally, you can classify the complexes as significant or not and annotate them if they are a GO, MIPS or KEGG term.

```
> status <- complexStatus(data=paraquat.complex,
+                                phenotype=paraquat,
+                                interactome=ScISIC, threshold=0.01)
> descr <- getDescr(status$A, database= c("GO","MIPS"))
> data.frame( descr,"pvalues"=paraquat.complex@pvalues[status$A])
```

```
                                        descr      pvalues
MIPS-220   H+-transporting ATPase, vacuolar 3.082381e-11
GO:0016469                             FALSE 7.909231e-08
GO:0000220                             FALSE 3.239645e-07
GO:0000814                             FALSE 1.035862e-06
GO:0000221                             FALSE 5.605086e-05
MIPS-90.30              ER assembly complex 1.076206e-04
GO:0000813                             FALSE 6.374648e-04
GO:0000815                             FALSE 6.374648e-04
```

Those results are in concordance with the known effect of paraquat on H+-transporting. Indeed, the mechanisms of the toxic effects of paraquat are largely the result of a metabolically catalyzed single-electron reduction-oxidation reaction, resulting in depletion of cellular NADPH and the generation of potentially toxic forms of oxygen such as the superoxide radical. It also highlight the critical role of the ESCRT complexes (Endosomal Sorting Complex Required for Transport).

4 Conclusion

This package offers computational methods and statistical paradigms to explore the relationships between phenotype and cellular organizational units. We demonstrated its usefulness in *S. cerevisiae* using **?** dataset and multi-protein complexes. While this is one example, we believe that those approaches are powerful enough to investigate many other phenotypes and estimates of organizational units within the genome, or proteome.