tmod:

An R Package for General and Multivariate Enrichment Analysis"

January Weiner and Teresa Domaszewska

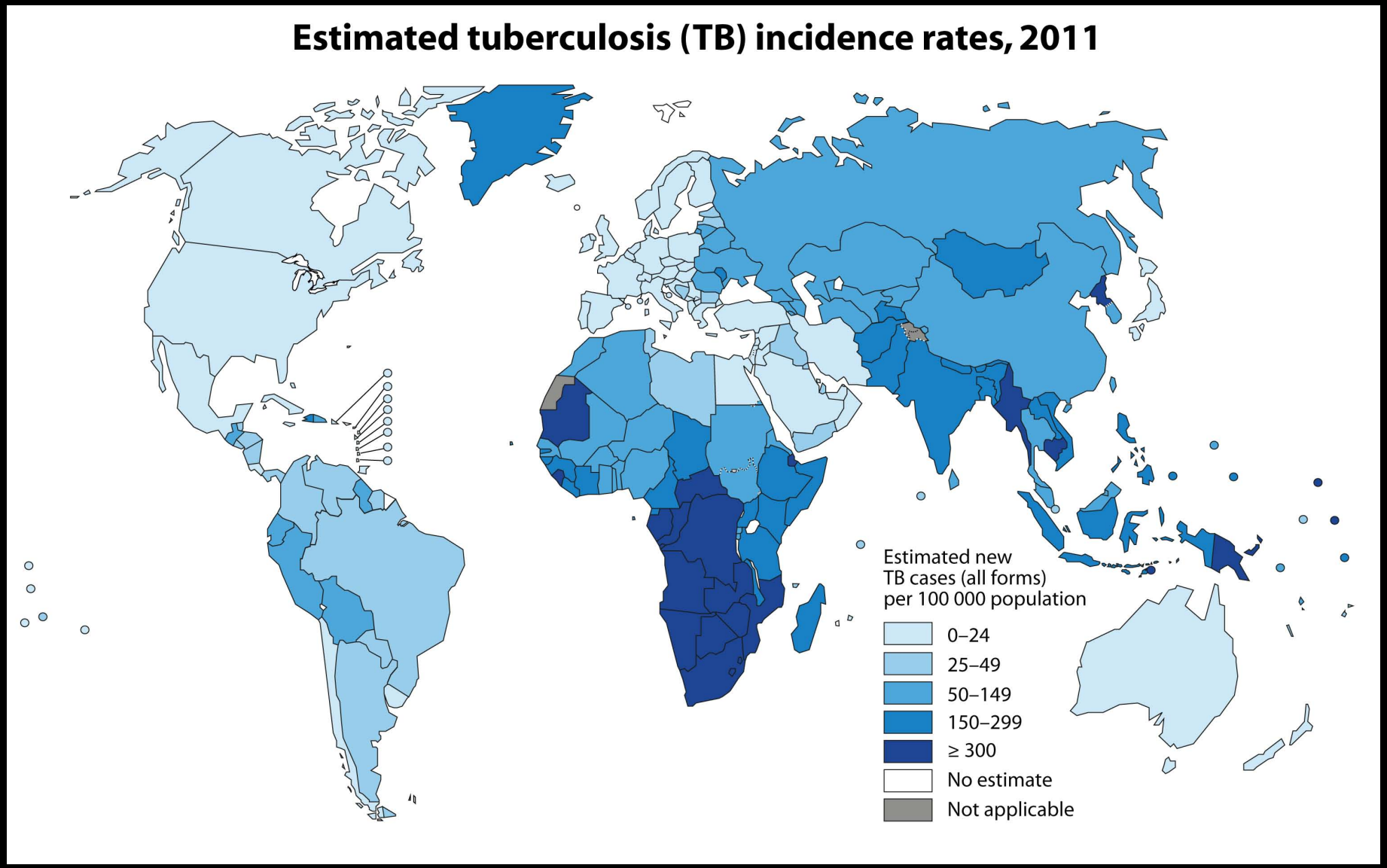
2016-09-13

Overview

About this presentation

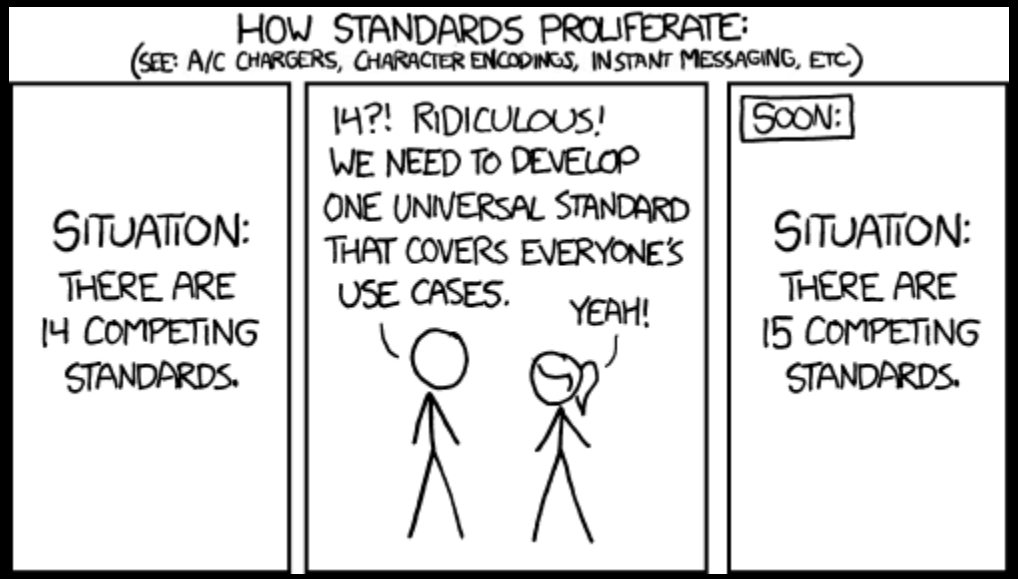
This is an Rmarkdown document; you can view the code on http://bioinfo.mpiib-berlin.mpg.de/tmod/

What do we work on



We needed a package for enrichment analysis

- Custom variable sets (expression modules, metabolic profiling)
- Statistical test(s):
  - not relying on arbitrary thresholds
  - preferably no bootstrapping
  - detached from differential analysis
  - pluggable into ML and MDS (e.g. PCA)
- Integrating in our R pipelines
- Highly flexible – diverse projects with different problem settings
- “Bird’s eye” visualizations for multiple analyses


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

(xkcd.com/927)

Introducing tmod

> *"Although there is no particular novelty in the methods, the package addresses the right questions and appears to do a good job on real biological analysis." (Anonymous reviewer)*

– Perfect!

tmod features

- CERNO: a new(-ish) statistical test for continuous enrichments (Yamaguchi et al. 2008, not implemented elsewhere)
- MSD: a new method for ordering genes
- “panel plots” – visualisation of gene set enrichment results
- prepackaged gene sets from Chaussabel et al. (2008) and Li et al. (2014) and metabolic profiling clustering from Weiner et al. (2012)

CERNO test: a variant of Fisher's exact test

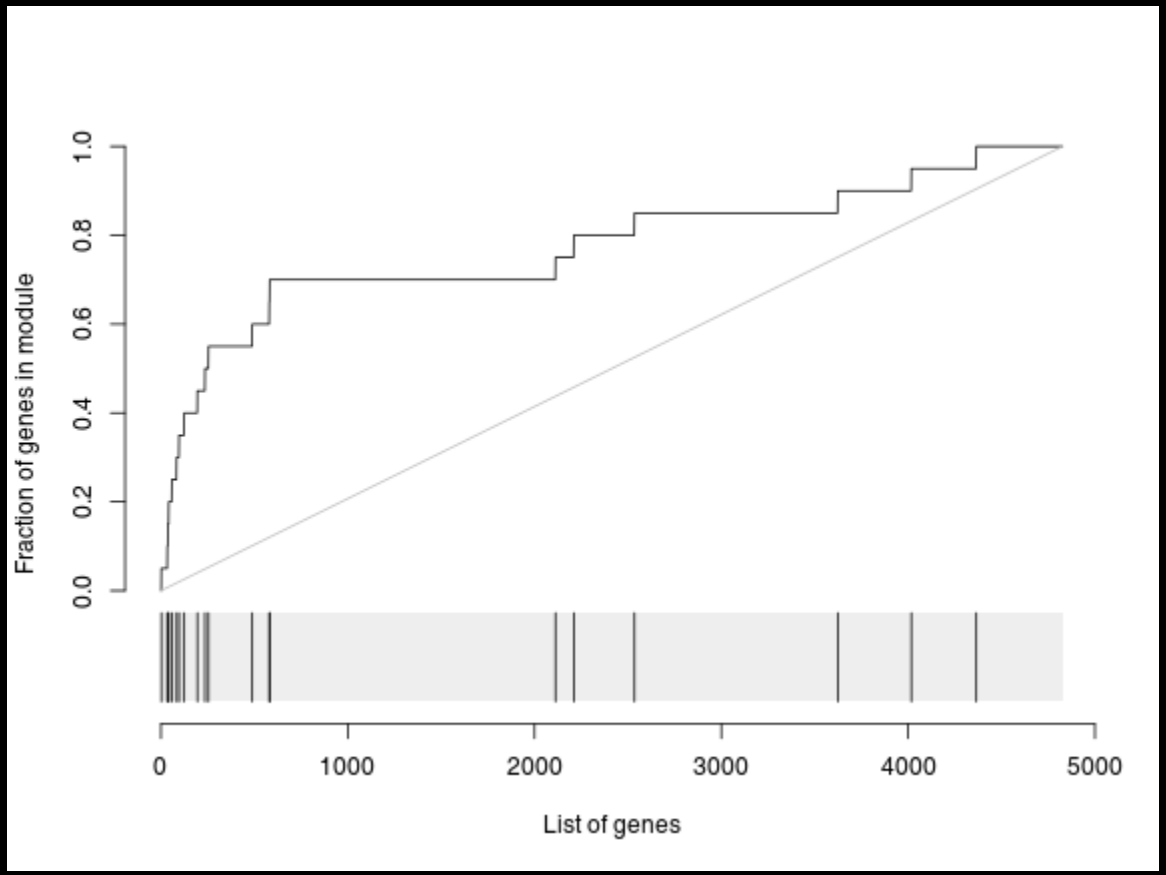
Some enrichment tests (such as the hypergeometric test) rely on arbirtrary tresholds to divide the genes into “differentially expressed” and “background” (or equivalent sets). It is easy to run a statistical test on such a setup, however it is problematic: the number of significantly regulated genes depends on the statistical power, i.e. for example on the number of samples.

Better tests yet are independent of arbitrary thresholds. Examples include

- Randomization approaches (such as GSEA)
- ANOVA-like approaches
- Mann-Whitney U statistic

How does this work?

```
evidencePlot(l=tt$GENE_SYMBOL, m="LI.M11.0")
```



In an U-test, the U statistic is (almost) the same as the Area Under Curve:

$$r = 1 - \frac{2 \cdot U}{n_1 \cdot n_2} = 1 - 2 \cdot \mathrm{AUC}$$

(r is the effect size for an U-test)

CERNO: Ranks can be treated as probabilities

$$P(rank(g_j) < rank(g_i)) = \frac{rank(g_i)}{N}$$

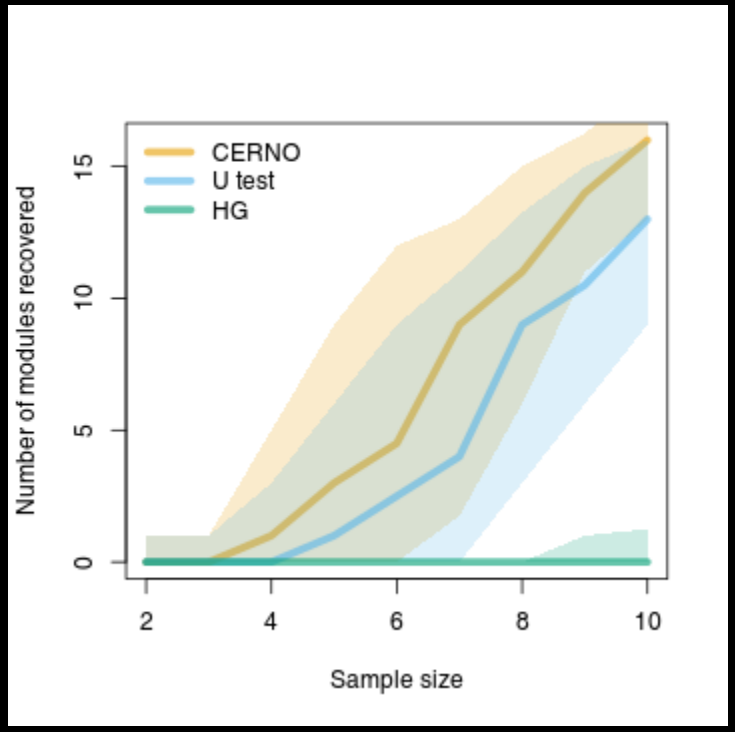
Where N is the total number of genes.

We apply Fisher's method to ranks

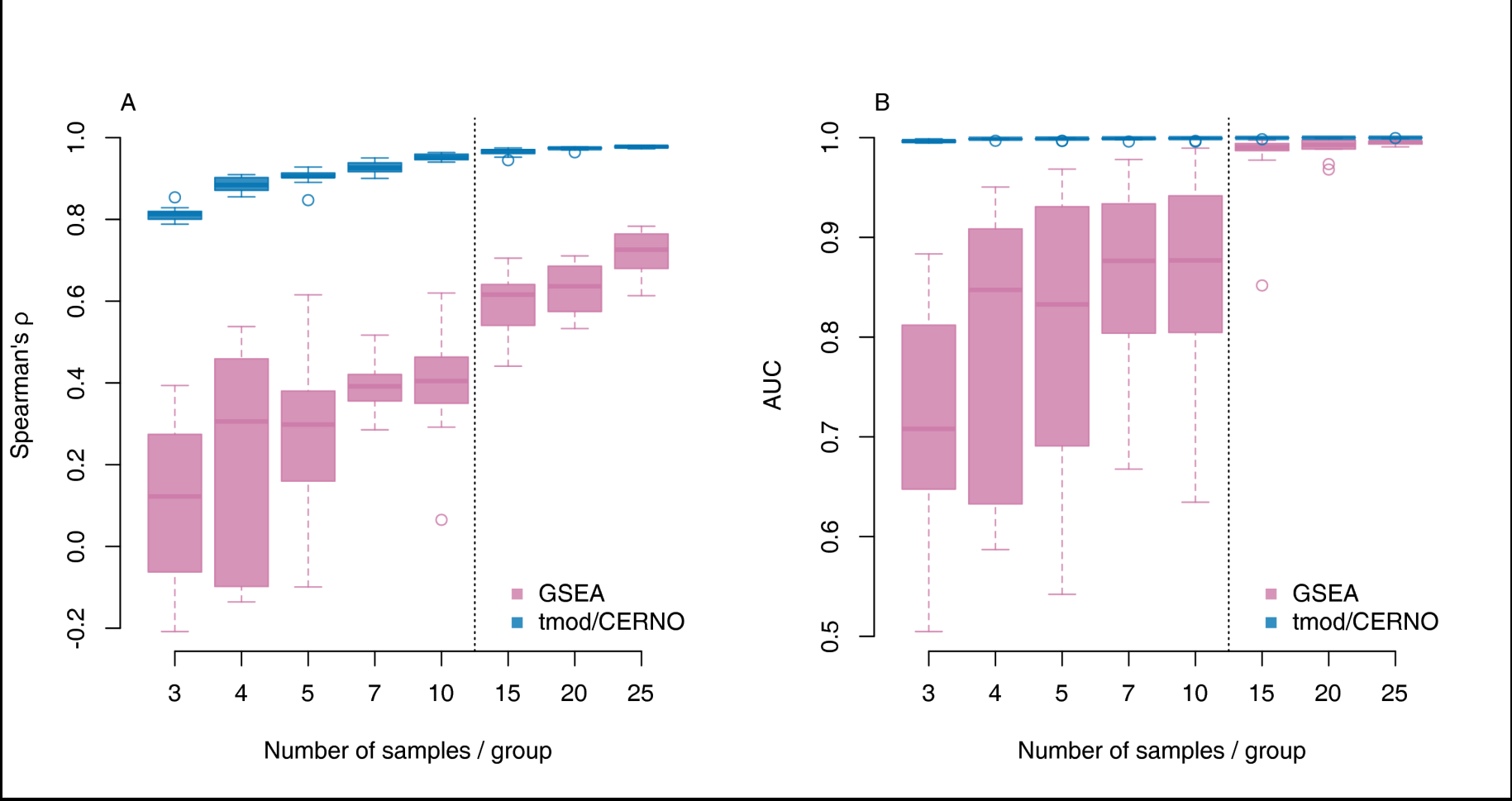
$$\mathbf{CERNO} = -2 \cdot \sum_{i=1}^{N} \ln(\frac{rank(g_i)}{N}) \Big|$$

The statistics has a $\chi^2 \Big|$ distribution with $2 \cdot N \Big|$ degrees of freedom.

First, second and third quartiles of number of modules recovered by the different statistical tests in dependence of the sample size in 100 random sample replicates.

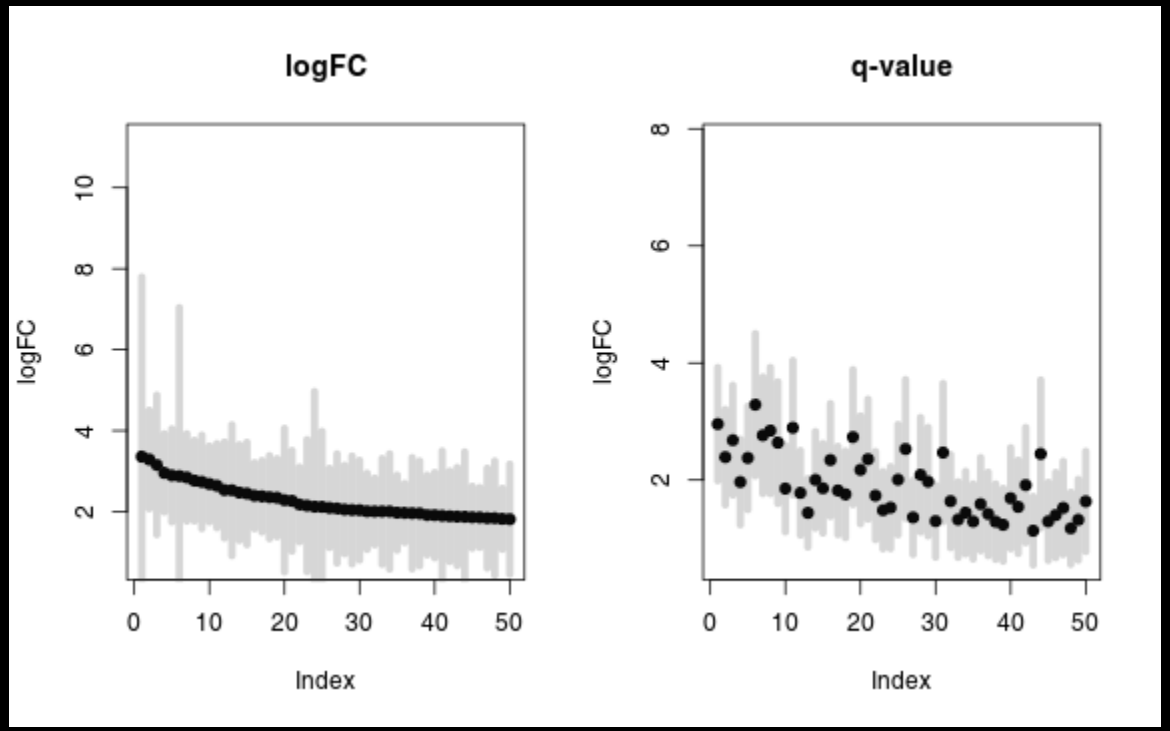


Sample size dependent recovery of results for tmod/CERNO and GSEA



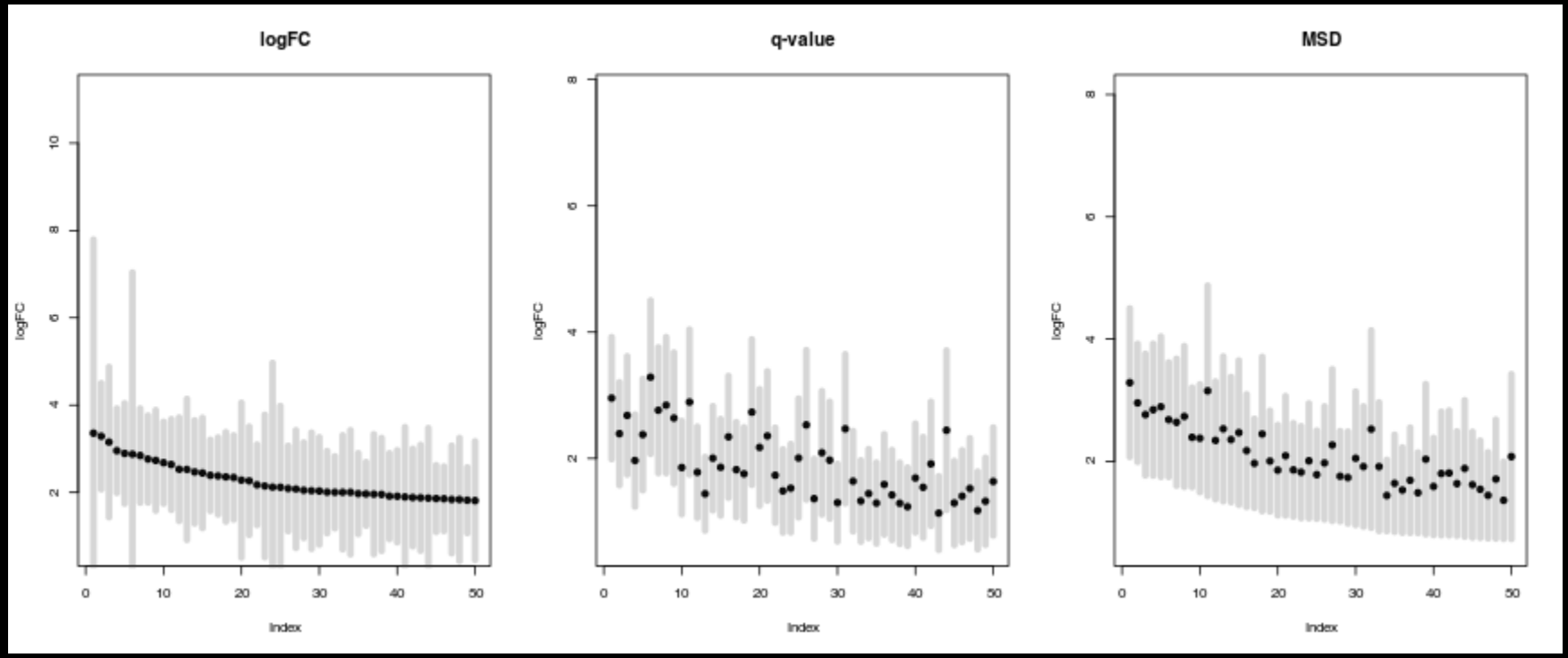
How to order genes?

- Order by p-values (common approach).
  - Genes with strong expression tend to have lower p-values even if log-fold changes are small
- Order by (absolute) log fold change
  - Genes with weak expression (near background) can have huge log fold changes despite lack of significance

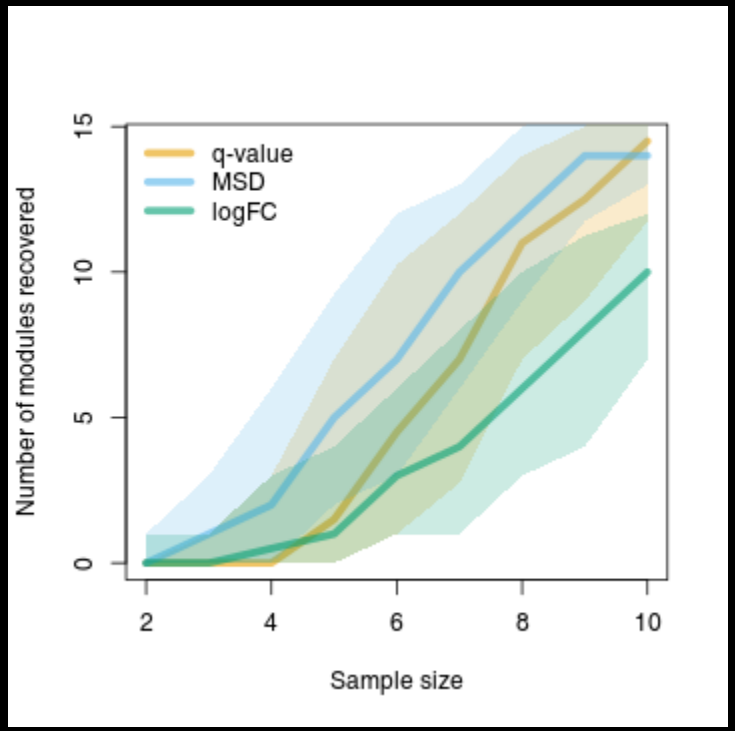

logFC
logFC
Index
q-value
logFC
Index

MSD – Minimal Significant Difference

$$\text{MSD} = \begin{cases} CI.L & \text{if logFC} > 0 \\ -CI.R & \text{if logFC} < 0 \end{cases}$$



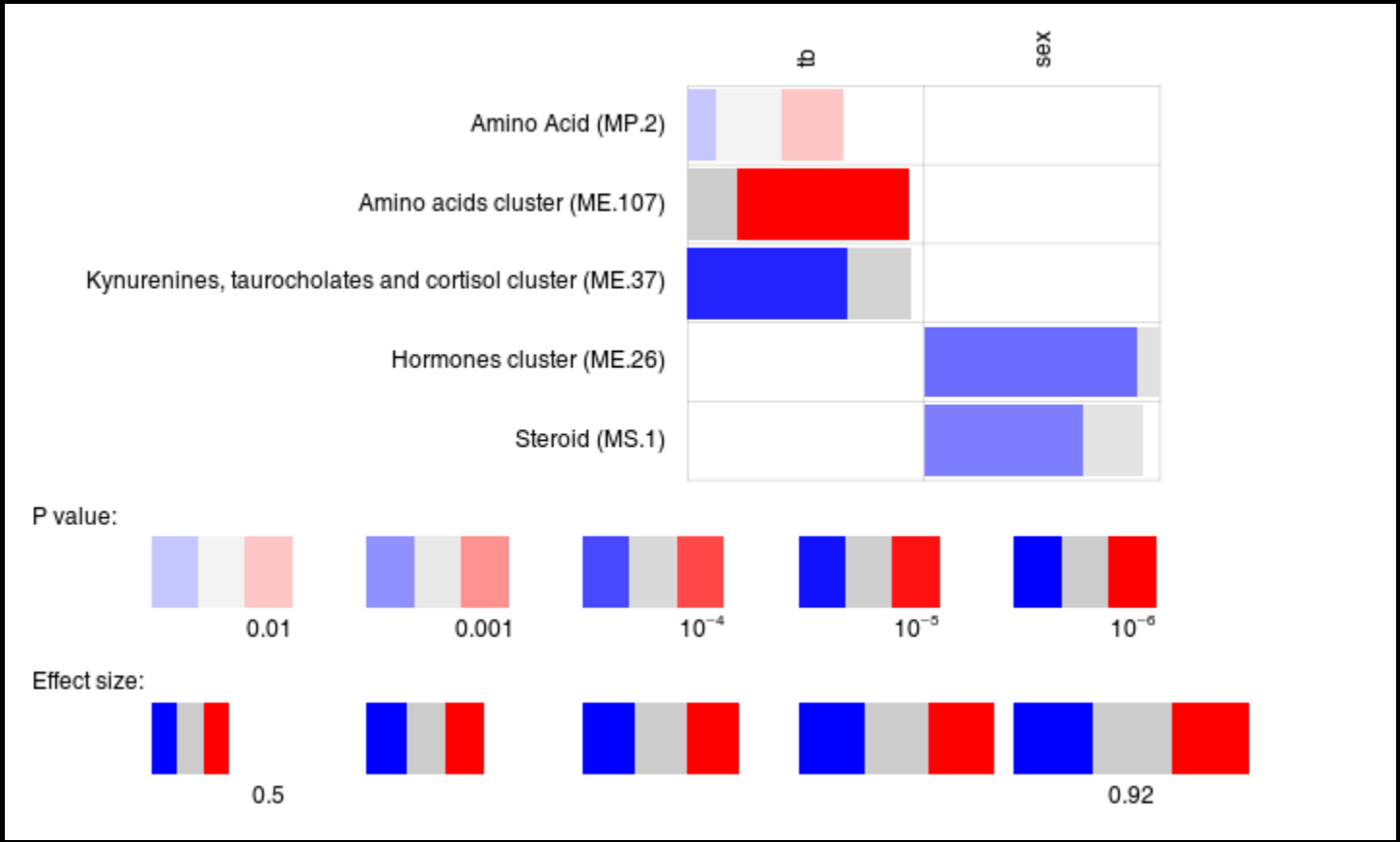
First, second and third quartiles of number of modules recovered by the different approaches in dependence of the sample size in 100 random sample replicates.



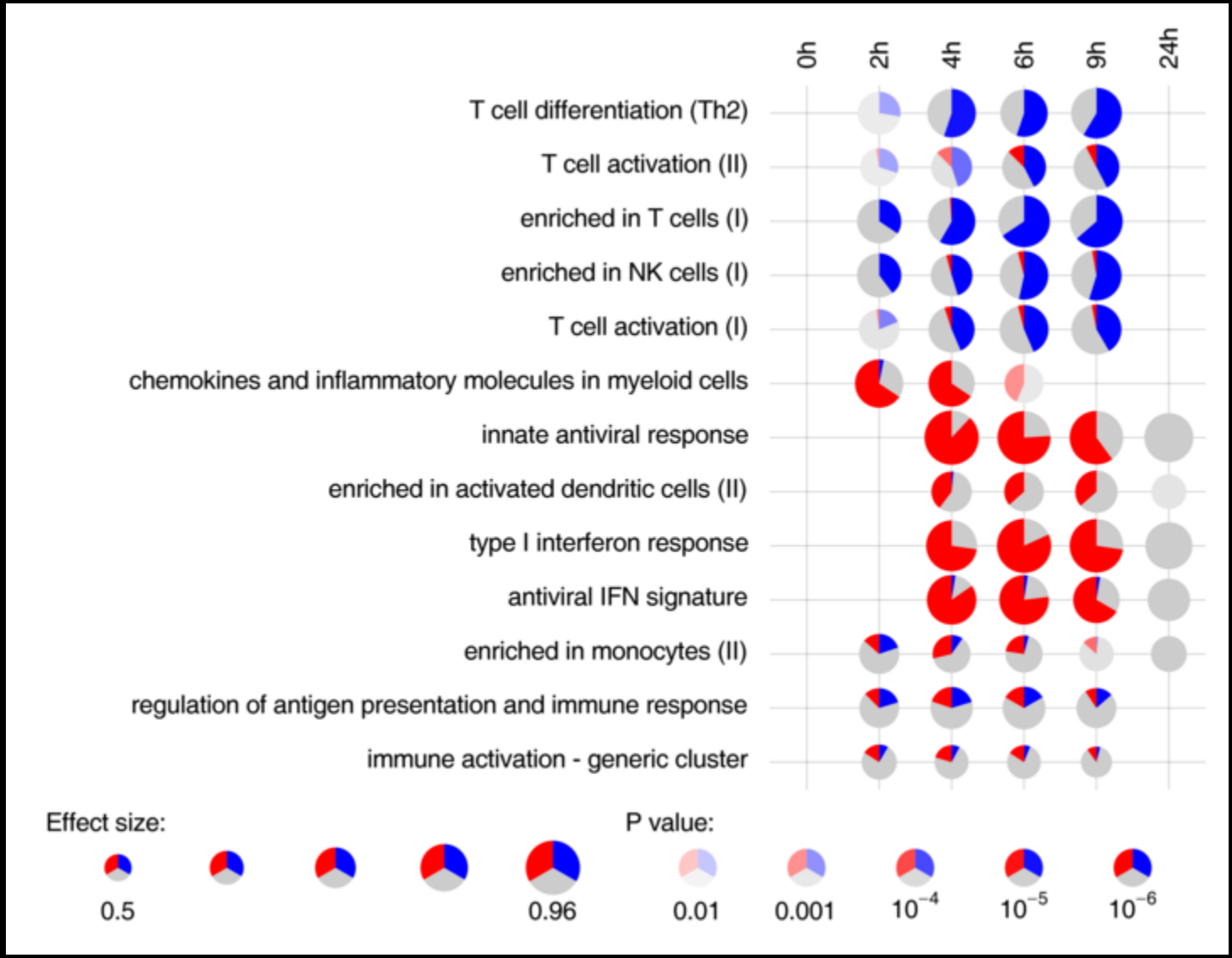
In comparisons with other ordering techniques, MSD has an exceptional specificity, while maintaining good sensitivity (Joanna Żyła, personal communication).

Visualisations in tmod

Panel plots showing effect sizes, p-values and direction of change



A more complex example



Functional multivariate analysis

Multivariate analysis + enrichment = Functional multivariate analysis (FMA)

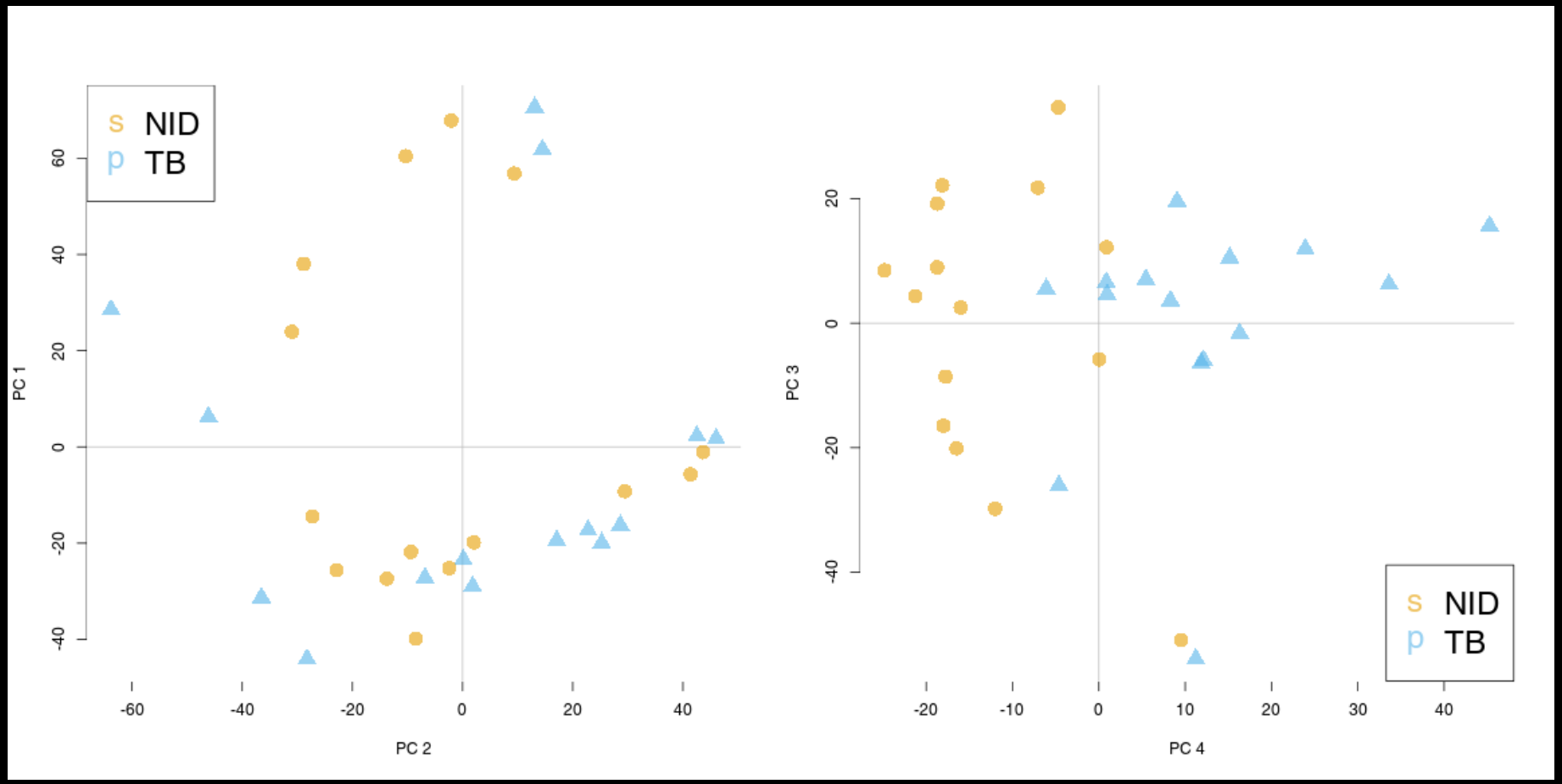
Combination of multivariate techniques such as PCA and functional enrichment analysis can circumvent the need for analysis of differential expression. A primer on FMA will be presented here.

Functional Principal Component Analysis (PCA)

In PCA, the $N \times K|$ matrix $\mathbf{X}|$ of $N|$ samples and $K|$ variables (e.g. genes) is rotated, which results in a new matrix, $\mathbf{Y}|$, with $N|$ samples and $J|$ principal components (PCs).

Effectively, a $K \times J|$ matrix $\mathbf{W}|$ is calculated, such that

$$\mathbf{X} \times \mathbf{W} = \mathbf{Y}|$$



Question in FMA: *What do these components mean?*

$$\mathbf{X} \times \mathbf{W} = \mathbf{Y}$$

Each column of $\mathbf{X}$ is a principal component. Each row corresponds to one sample.

A value for a given PC j and a given sample n is calculated as

$$y_{n,j} = \sum_{k=1}^{K} w_{j,k} \cdot x_{k,n}$$

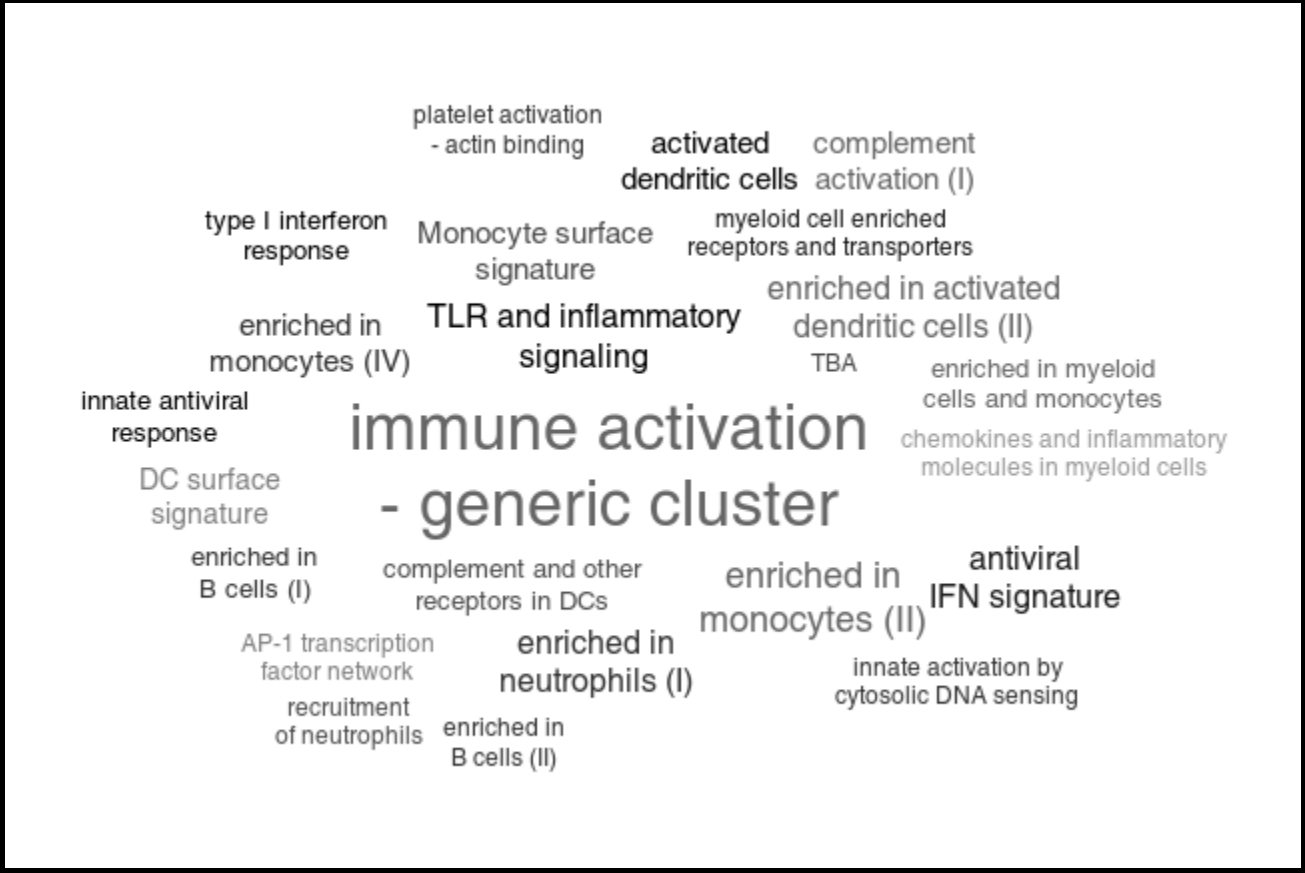
The terms $w_{j,k}$ are variable- (or: gene-) specific *weights* or *loadings* for each component j.

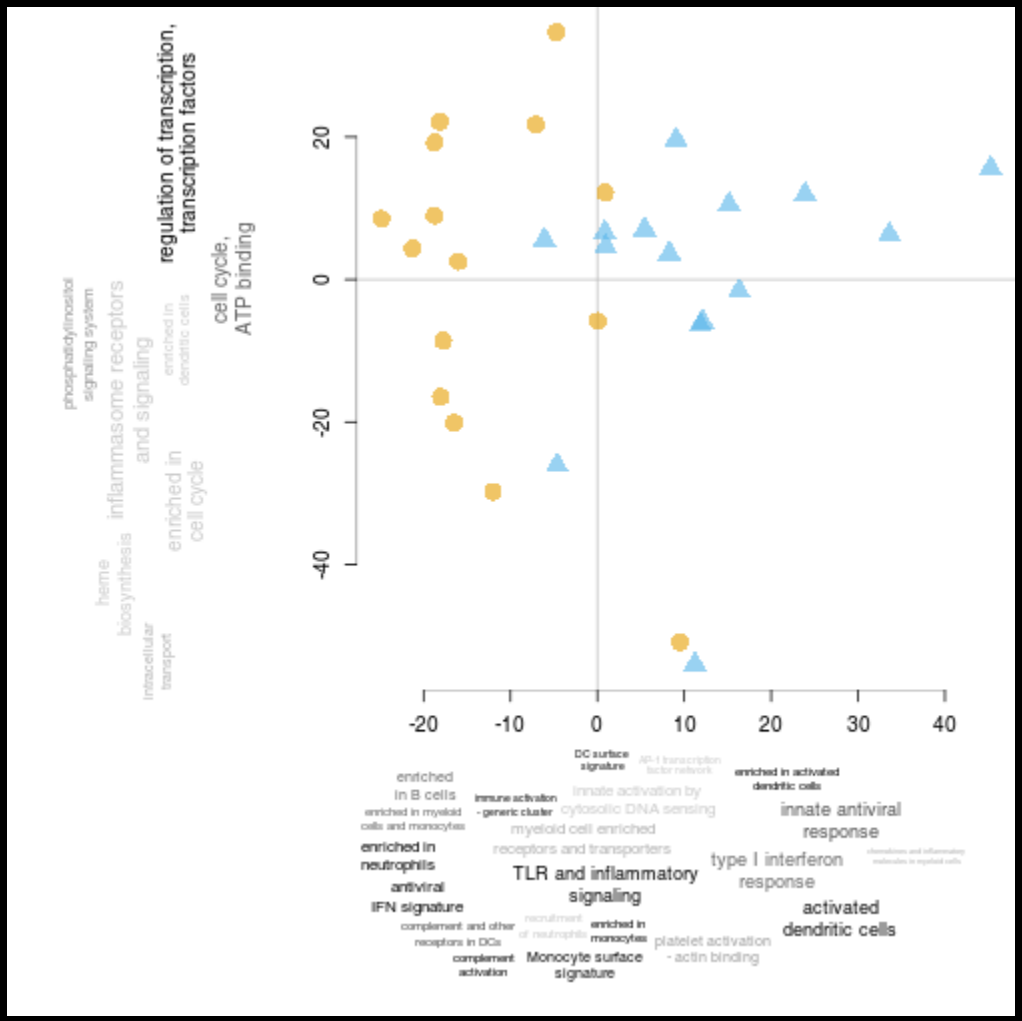
$$y_{(n,j)} = \sum_{k=1}^{K} w_{k,j} \cdot x_{k,n} \Big|$$

The larger the absolute value of $w_{k,j}|$, the more impact this gene has on the j-th principal component.

We can sort the genes by their weight in a component. Since as a result we get a sorted list of genes, we can apply a continuous enrichment algorithm.

Enrichment in component 4




regulation of transcription, transcription factors
cell cycle, ATP binding
phosphatidylinositol signaling system
inflammasome receptors and signaling
enriched in dendritic cells
enriched in cell cycle
heme biosynthesis
intracellular transport
20
0
-20
-40
-20
-10
0
10
20
30
40
DC surface signature
AP-1 transcription factor network
enriched in activated dendritic cells
enriched in B cells
innate activation by cytosolic DNA sensing
innate antiviral response
immune activation - generic cluster
enriched in myeloid cells and monocytes
myeloid cell enriched receptors and transporters
enriched in neutrophils
type I interferon response
chemokines and inflammatory molecules in myeloid cells
TLR and inflammatory signaling
antiviral IFN signature
activated dendritic cells
recruitment of neutrophils
enriched in monocytes
complement and other receptors in DCs
platelet activation - actin binding
complement activation
Monocyte surface signature

This approach works well also with other multivariate analyses such as independent component analysis (ICA), partial least squares (PLS) or correspondance analysis (CA).

Directly combining multivariate analyses with gene set enrichment allows us to achieve the same results without involving a direct group - to - group comparison. This makes it especially suitable for exploratory analyses.

Serial analysis of enrichment with *tmod*

tmod has been designed as a package for testing the enrichment of blood transcriptional modules. Therefore, *tmod* contains two sets of blood transcriptional module definitions; however, it can be used with any arbitrary gene set definition (e.g. GSEA/MSigDB) or high throughput data type (e.g. metabolomics)

tmod implements HG / U / CERNO tests, functional multivariate analyses, serial analysis / visualization and more.

Availability: http://bioinfo.mpiib-berlin.mpg.de/tmod/

Example: MFA with R and tmod

Data set Egambia: GEO GSE28623.

```
library(tmod)
data(Egambia)
head(Egambia)
```

```
##    GENE_SYMBOL                                    GENE_NAME     EG        NID
## 34    C19orf15           chromosome 19 open reading frame 15  57828  3.2618218
## 36     UNQ9368                                    RTFV9368 643036  1.5671748
## 41      ADORA3                       adenosine A3 receptor    140  6.2246027
## 44        CDH6 cadherin 6, type 2, K-cadherin (fetal kidney)"   1004  0.8328559
## 52       VASH1                                  vasohibin 1  22846 11.3952226 1
## 62     MAB21L2                    mab-21-like 2 (C. elegans)  10586  5.7530317
##          NID       NID        NID        NID       NID       NID       NID
## 34  4.617986  3.033595  3.1866326  3.6506719  3.787375  3.019342  2.795293  3.020
## 36  4.786995  3.091925  2.2736422  4.1327518  3.934754  3.077131  6.428547  4.655
## 41  6.878103  4.702415  7.6848512  5.2048066  4.836591  4.965997  8.234983  5.072
## 44  2.589377  3.307486  0.7026353  0.8349973  3.951534  2.112500  1.223633  1.477
## 52 11.376962 13.061029 13.0915988 12.0304966 11.980200 12.323327 11.076847 13.187
## 62  7.167419  6.299295  5.8910289  5.4252899  5.265659  6.367774  6.691451  6.351
##           TB        TB         TB        TB        TB        TB        TB
## 34  3.962293  2.080173  3.750405  2.248475  4.148280  4.203384  4.319223
## 36  5.551801  5.021816  5.338259  6.258222  6.383069  5.995486  5.203686
## 41  7.689227  6.004437  5.928957  5.178725  5.661376  6.611350  6.008429
## 44  0.977040  1.174805  1.764985  3.400484  2.486234  1.115761  1.454525
```

```r
pca <- prcomp(t(Egambia[,-c(1:3)]), scale.=TRUE)
names(pca)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

```r
head(pca$x[,1:5])
```

```
##              PC1        PC2        PC3          PC4        PC5
## NID   -27.40722 -13.745073  21.755484  -7.09238147   3.427730
## NID.1  60.44502 -10.323684 -20.091148 -16.49059391  28.610291
## NID.2  67.86063  -2.049866  -5.840748   0.03206066  -4.409519
## NID.3  -5.69328  41.378405 -16.466891 -18.00896080  -5.135397
## NID.4 -25.59922 -22.852422  22.133902 -18.13982412 -13.619485
## NID.5 -39.83765  -8.461881 -50.897837   9.51198824   3.981510
```

```r
head(pca$rotation[,1:5])
```

```
##              PC1           PC2          PC3          PC4          PC5
## 34 -0.018481790  0.0024852364 -0.005385593  0.002711500 -0.025291642
## 36  0.003759772 -0.0010608658 -0.018658252  0.041998920  0.006653159
## 41 -0.016103961  0.0002934259 -0.009930961 -0.008813419 -0.011689654
## 44  0.021992983 -0.0073125000  0.005115505  0.015148515  0.001656177
## 52  0.014524608  0.0258657287  0.015226449 -0.009626087 -0.001871263
## 62 -0.001791174  0.0081258050 -0.016757250  0.001834450 -0.009573289
```

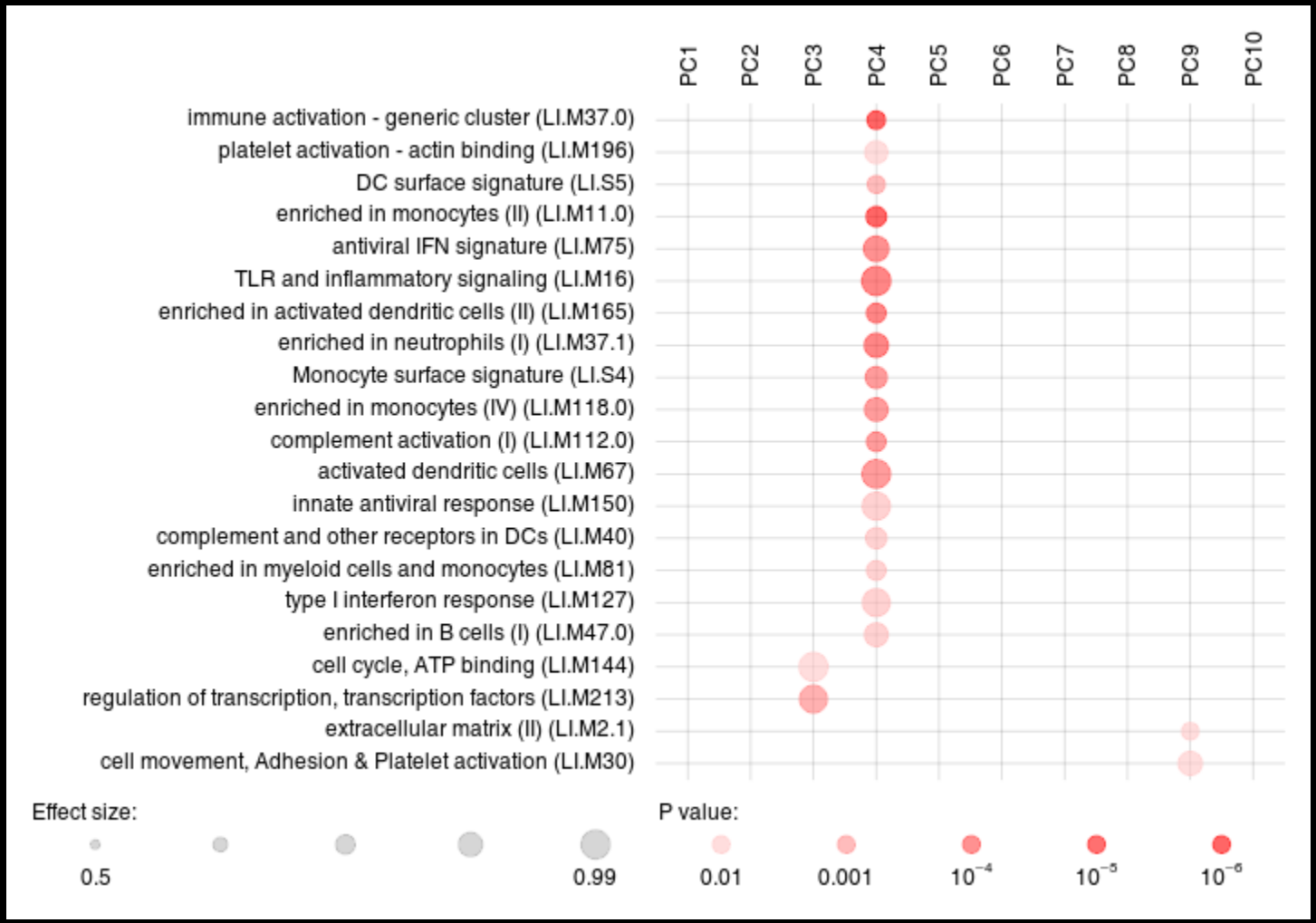
Enrichment for each component

```
l <- Egambia$GENE_SYMBOL
encfunc <- function(r) {
  o <- order(abs(r), decreasing=TRUE)
  tmodCERNOtest(l[o])
}
res <- apply(pca$rotation[,1:10], 2, encfunc)
head(res[[4]])
```

```
##                ID                                     Title     cerno  N1        
## LI.M37.0 LI.M37.0        immune activation - generic cluster 454.88172 100 0.7188
## LI.M11.0 LI.M11.0                  enriched in monocytes (II) 118.06755  20 0.7734
## LI.M165   LI.M165 enriched in activated dendritic cells (II) 101.09999  19 0.7562
## LI.M37.1 LI.M37.1                enriched in neutrophils (I)  77.04015  12 0.8671
## LI.M16     LI.M16             TLR and inflammatory signaling  50.11235   5 0.9923
## LI.M75     LI.M75                   antiviral IFN signature  67.61164  10 0.9007
```

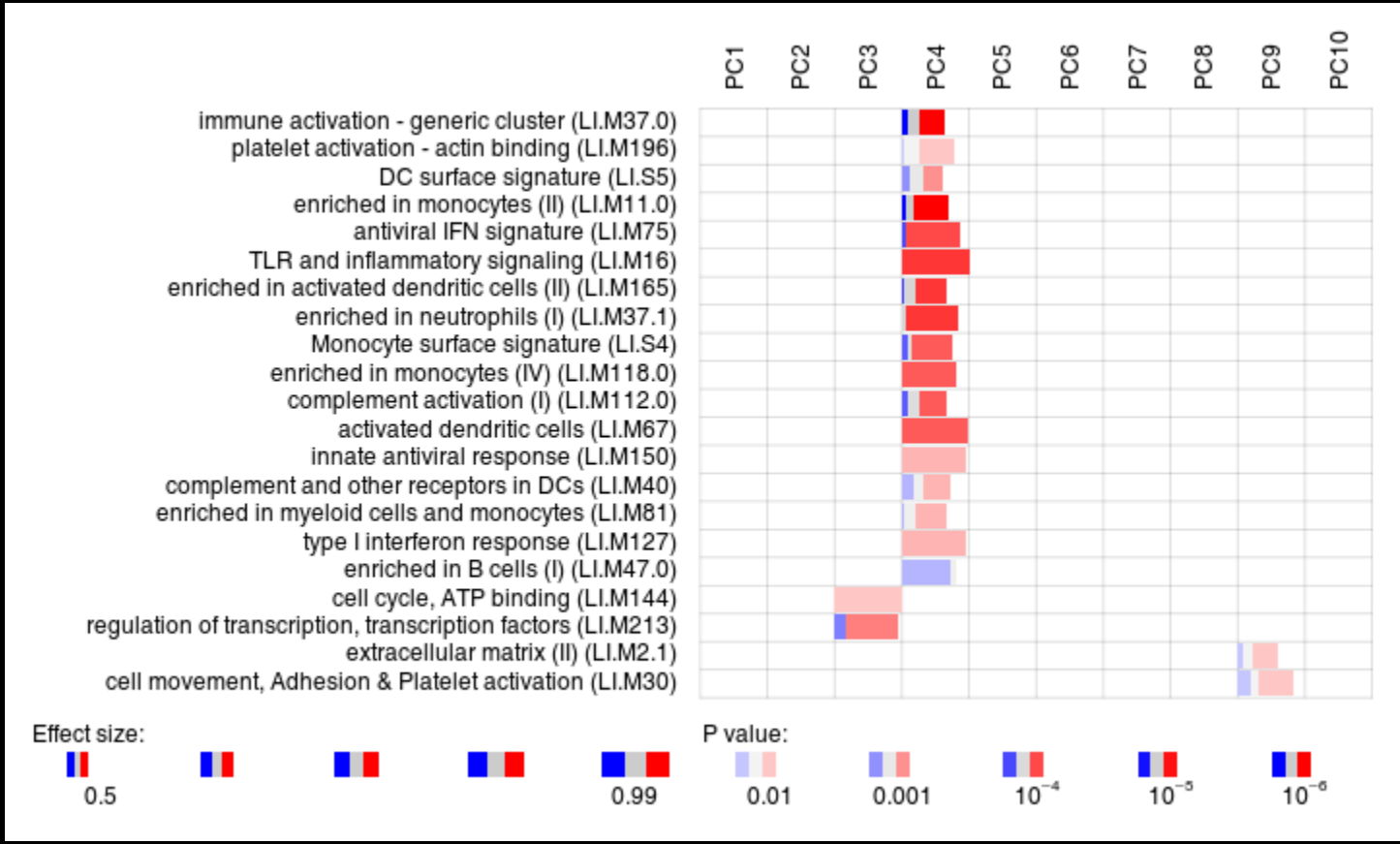
Visualization

```
tmodPanelPlot(res, filter.empty.rows=TRUE)
```



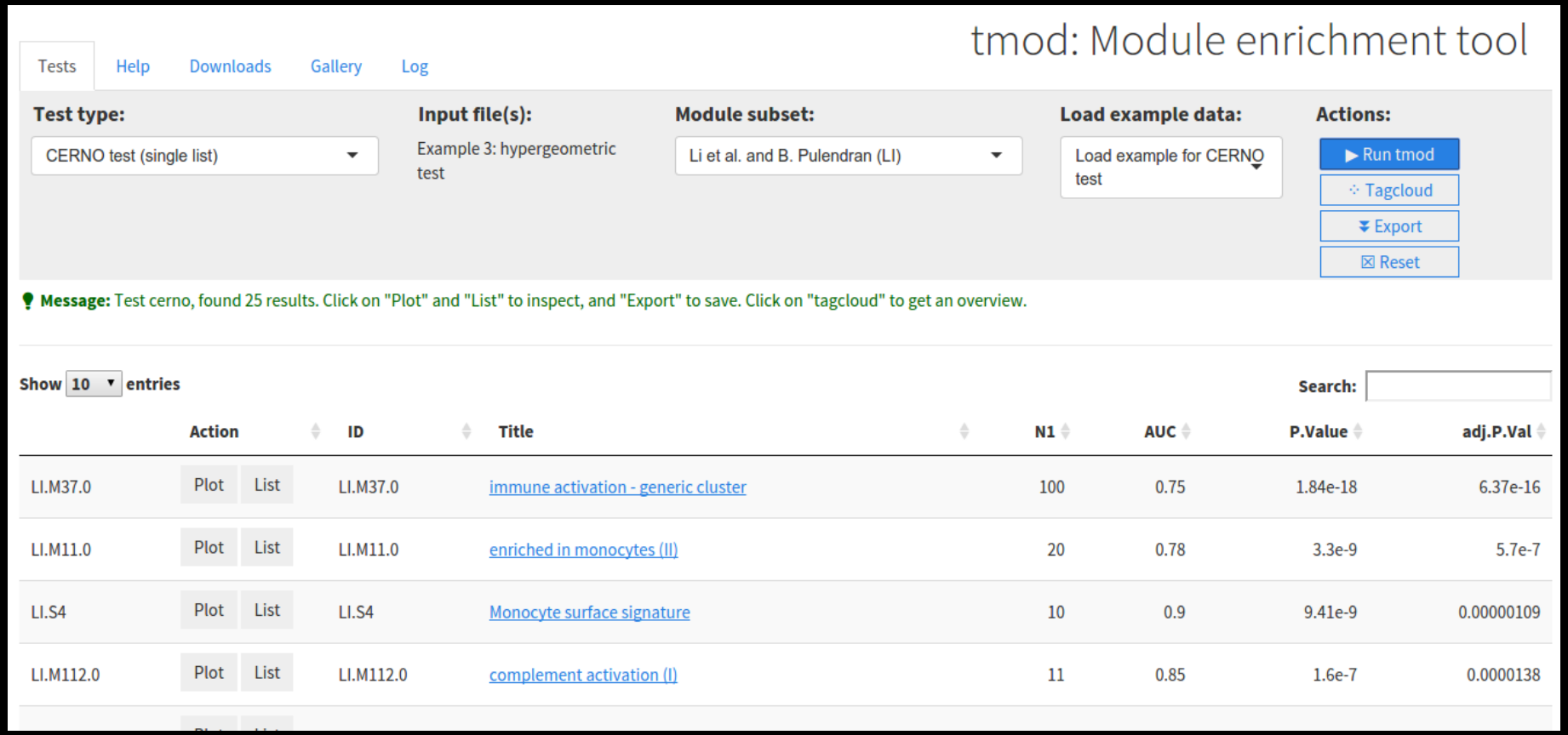
Genes with positive / negative weights?

```
qfnc <- function(r) quantile(r, 0.75)
qqs <- apply(pca$rotation[,1:10], 2, qfnc)
pie <- tmodDecideTests(l, lfc=pca$rotation[,1:10], lfc.thr=qqs)
tmodPanelPlot(res, pie=pie, pie.style="rug", grid="between")
```



tmod Web Interface

http://bioinfo.mpiib-berlin.mpg.de/tmod/.



Concluding remarks

- Gene set enrichment analysis is a versatile tool for functional annotation
- Functional multivariate analysis can replace differential expression analysis
- *tmod*: R package for BTM and GS enrichment analysis, available from http://bioinfo.mpiib-berlin.mpg.de/tmod/ and CRAN
- *tmod* allows functional multivariate analysis and serial enrichment analyiss
- features several visualization tools
- Where to find me: MPIIB january@mpiib-berlin.mpg.de

Conributors

- Teresa Domaszewska (MPIIB) – co-author (see our poster)

We would like to thank Emilio Siena (GSK Vaccines) for new ideas, as well Joanna Żyła, Michał Marczyk and Joanna Polańska (Politechnika Śląska) for helpful discussions and the BMBF / InfectControl2020 for supporting Teresa Domaszewska.

ZIBI GRADUATE SCHOOL BERLIN
RESEARCH IN INFECTION BIOLOGY AND IMMUNOLOGY

zwanzig20
PARTNERSCHAFT FÜR INNOVATION

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

UNTERNEHMEN REGION
Die BMBF-Innovationsinitiative Neue Länder

BIOVACSAFE
Biomarkers for enhanced Vaccine immunoSafety

Appendix

You can download the source code of this presentation on the tmod web page, http://bioinfo.mpiib-berlin.mpg.de/tmod/.

To recreate this presentation, download the full presentation package and unzip it. Install the required packages (knitr for R and pandoc). Run the following command from inside the package archive.

Commands:

```
Rscript -e 'knitr::knit("weiner_bioinfo_2015_06_23.Rmd")'
pandoc -s -S -t revealjs weiner_bioinfo_2015_06_23.md -o weiner_bioinfo_2015_06_23.h
  --mathjax='http://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AMS-MML_HTM
  --css css/mytheme.css \
  --slide-level 2 -V theme=blood
```

(Note: for an offline version, download MathJax and modify the –mathjax option)

To extract the code from this presentation, save it as “test.Rmd” and run

```
Rscript -e 'knitr::purl("tmod.Rmd")'
```

Printing:

This only works properly in Google Chromium; see reveal.js documentation

To print, follow the link below and press Ctrl-P; don't worry if the slides appear to overlap – they will look fine on the print preview.

Print