Introduction to Transcriptomics Analysis

Class 15 - Downstream Analysis I
GO Term Analysis

INSTRUCTOR:
Aureliano Bombarely
Department of Bioscience
Universita degli Studi di Milano
aureliano.bombarely@unimi.it

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

1. Basics about GO Terms.

Gene ontologies:

Structured controlled vocabularies (ontologies) that describe **gene products** in terms of their associated

biological processes,

cellular components and

molecular functions

in a species-independent manner

http://www.geneontology.org/GO.doc.shtml

Ontology

Property	Value
Valid terms	44411 (Δ = -97)
Obsoleted terms	2947 (Δ = 23)
Merged terms	2056 (Δ = 91)
Biological process terms	29112
Molecular function terms	11118
Cellular component terms	4181

Annotations

Property	Value
Number of annotations	7,975,639
Annotations for biological process	3,069,526
Annotations for molecular function	2,455,089
Annotations for cellular component	2,451,024
Annotations for evidence PHYLO	4,163,423
Annotations for evidence IEA	1,978,576
Annotations for evidence EXP	759,654
Annotations for evidence OTHER	791,743
Annotations for evidence ND	241,978
Annotations for evidence HTP	40,265
Number of annotated scientific publications	159,963

1. Basics about GO Terms.

Biological processes,

Recognized series of events or molecular functions. A process is a collection of molecular events with a defined beginning and end.

Cellular components,

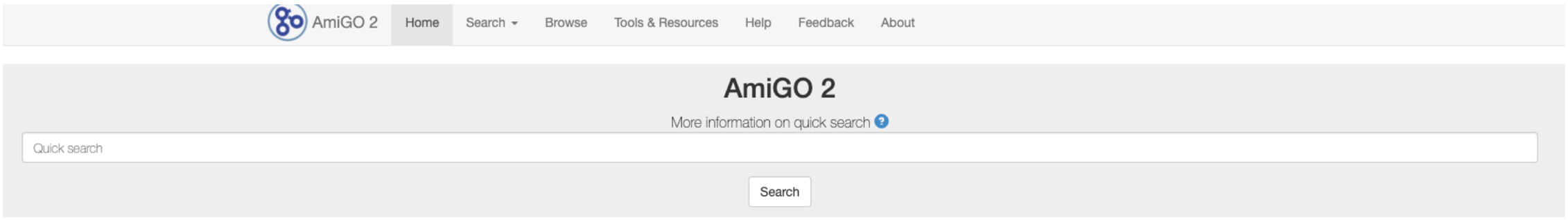
Describes locations, at the levels of subcellular structures and macromolecular complexes.

Molecular functions

Describes activities, such as catalytic or binding activities, that occur at the molecular level.

1. Basics about GO Terms.

The information about the Gene Ontology terms can be retrieved at:
http://amigo.geneontology.org/amigo/landing



1. Basics about GO Terms.

The information about the Gene Ontology terms can be retrieved at:
http://amigo.geneontology.org/amigo/landing

Term Information

Accession	GO:0007165
Name	signal transduction
Ontology	biological_process
Synonyms	signaling pathway, signalling pathway, signaling cascade, signalling cascade
Alternate IDs	GO:0023033
Definition	The cellular process in which a signal is conveyed to trigger a change in the activity or state of a cell. Signal transduction begins with reception of a signal (e.g. a ligand binding to a receptor or receptor activation by a stimulus such as light), or for signal transduction in the absence of ligand, signal-withdrawal or the activity of a constitutively active receptor. Signal transduction ends with regulation of a downstream cellular process, e.g. regulation of transcription or regulation of a metabolic process. Signal transduction covers signaling from receptors located on the surface of the cell and signaling via molecules located within the cell. For signaling between cells, signal transduction is restricted to events at and within the receiving cell. *Source:* GOC:mtg_signaling_feb11, GOC:go_curators
Comment	Note that signal transduction is defined broadly to include a ligand interacting with a receptor, downstream signaling steps and a response being triggered. A change in form of the signal in every step is not necessary. Note that in many cases the end of this process is regulation of the initiation of transcription. Note that specific transcription factors may be annotated to this term, but core/general transcription machinery such as RNA polymerase should not.
History	See term history for GO:0007165 at QuickGO
Subset	goslim_metagenomics goslim_aspergillus goslim_chembl goslim_plant goslim_generic goslim_candida
Related	Link to all **genes and gene products** annotated to signal transduction. Link to all direct and indirect **annotations** to signal transduction. Link to all direct and indirect **annotations download** (limited to first 10,000) for signal transduction.

Data health

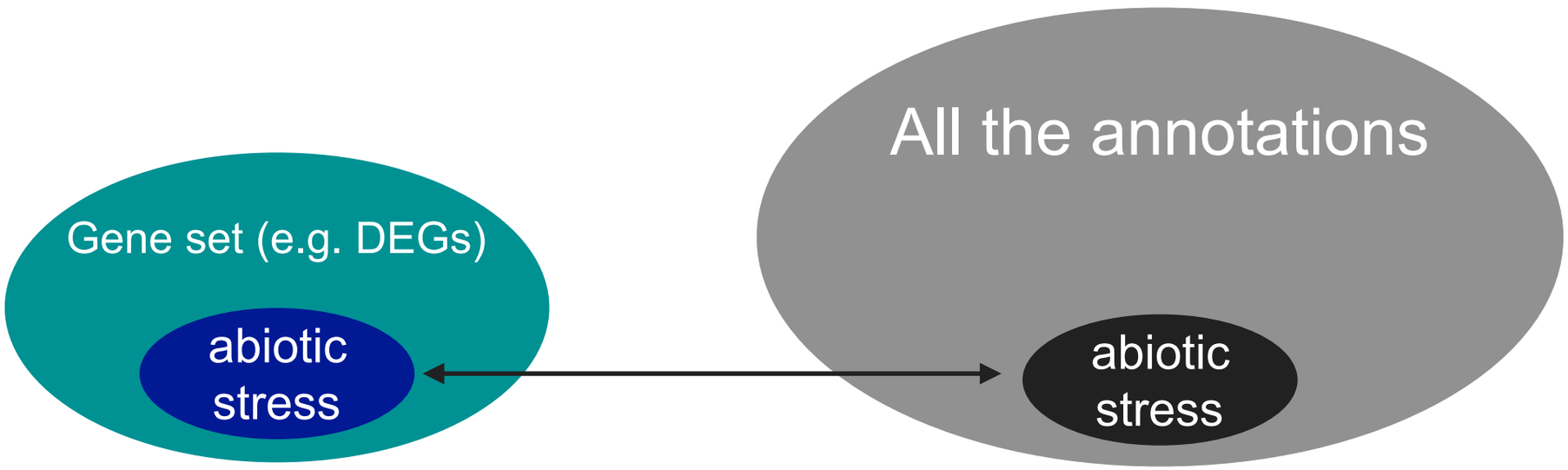
- R GO:0008150 biological_process
  - I GO:0065007 biological regulation
    - R GO:0009987 cellular process
    - I GO:0050789 regulation of biological process
    - P GO:0050896 response to stimulus
      - P GO:0007154 cell communication
      - P GO:0051716 cellular response to stimulus
      - I GO:0050794 regulation of cellular process
      - P GO:0023052 signaling
        - **GO:0007165 signal transduction**
          - I GO:0097190 apoptotic signaling pathway
          - I GO:0038183 bile acid signaling pathway
          - I GO:0099004 calmodulin dependent kinase signaling pathway
          - I GO:0009756 carbohydrate mediated signaling
          - I GO:0007166 cell surface receptor signaling pathway
          - I GO:0010019 chloroplast-nucleus signaling pathway
          - I GO:0007212 dopamine receptor signaling pathway

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

2. Gene Set Enrichment Analysis

One of the main uses of the GO is to perform enrichment analysis on gene sets. For example, given a set of genes that are up-regulated under certain conditions, an enrichment analysis will find ***which GO terms are over-represented (or under-represented) using annotations for that gene set***.



Is the subgroup proportion significant?

http://geneontology.org/docs/go-enrichment-analysis/

2. Gene Set Enrichment Analysis

One of the main uses of the GO is to perform enrichment analysis on gene sets. For example, given a set of genes that are up-regulated under certain conditions, an enrichment analysis will find ***which GO terms are over-represented (or under-represented) using annotations for that gene set***.

Interpretation:

Background frequency is the number of genes annotated to a GO term in the entire background set.

Sample frequency is the number of genes annotated to that GO term in the input list.

Overrepresented (+) or ***underrepresented*** (-)

P-value is the probability or chance of seeing at least x number of genes out of the total n genes in the list annotated to a particular GO term, given the proportion of genes in the whole genome that are annotated to that GO Term.

http://geneontology.org/docs/go-enrichment-analysis/

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3. Tools to analyse GO Terms.

There are different tools to perform a GO term enrichment analysis. Some popular ones such as Blast2GO runs on a standalone software and run these analysis as a part of a bigger pipeline.

Tools can be divided as:

- Web applications.
- Standalone applications.
- R (and other programs) packages.

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.1. Web based tools

Most of the web tools are quite intuitive but the lack in the flexibility that it is needed to analyse non-model organisms. ***The have the "universe" pre-set, so the identifiers should be the same that the gene_ids of these pre-sets***.

Some examples are:

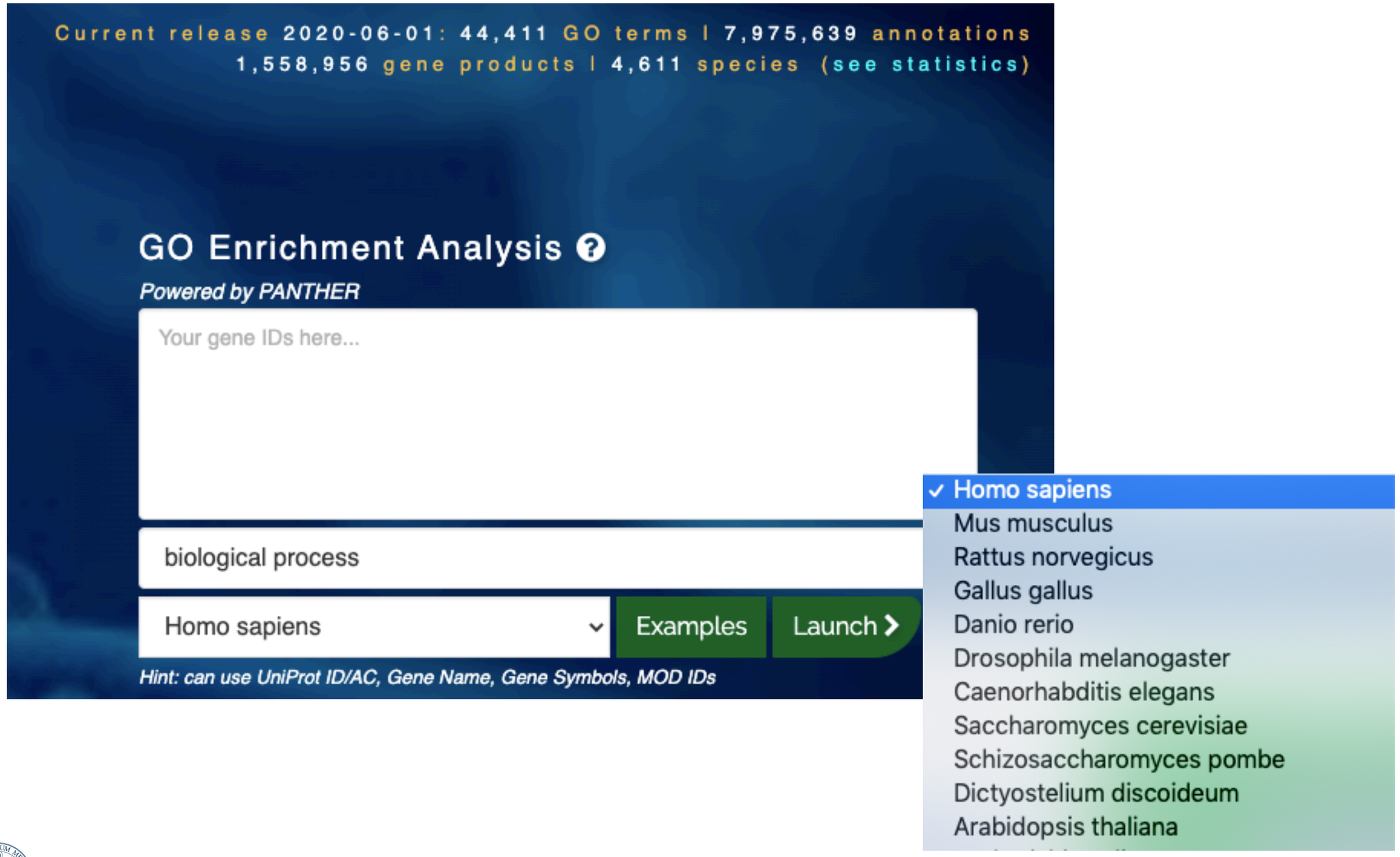
- BINGO (https://www.psb.ugent.be/cbd/papers/BiNGO/Home.html). It is a Java-based tool implemented as a plugin of Cytoscape.
- GeneWeaver (https://www.geneweaver.org/). Web application for the integrated cross-species analysis of functional genomics data from heterogeneous sources.
- gProfiler (http://biit.cs.ut.ee/gprofiler/gost). Web application with ENSEMBL genomes, including several plants.
- Ontologizer (http://ontologizer.de/). It is a Java Webstart application.
- GOrilla (http://cbl-gorilla.cs.technion.ac.il/). Web application with most of the classical models.
- Gene Ontology (http://geneontology.org/).

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.1.1. Gene Ontology

http://geneontology.org/



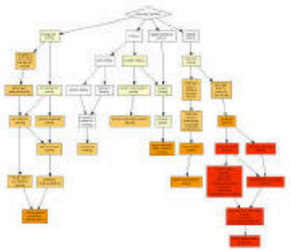
Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2.1. GOrilla

http://cbl-gorilla.cs.technion.ac.il/

***GO**RILLA*



Gene Ontology enRIchment anaLysis and visuaLizAtion tool

GOrilla is a tool for identifying and visualizing enriched GO terms in ranked lists of genes.
It can be run in one of two modes:

1. Searching for enriched GO terms that appear densely at the top of a ranked list of genes or
2. Searching for enriched GO terms in a target list of genes compared to a background list of genes.

For further details see References.

Running example Usage instructions GOrilla News References Contact

Step 1: Choose organism

Homo sapiens

Step 2: Choose running mode

(•) Single ranked list of genes ( ) Two unranked lists of genes (target and background lists)

Step 3: Paste a ranked list of gene/protein names

Names should be separated by an <ENTER>. The preferred format is gene symbol. Other supported formats are: gene and protein RefSeq, Uniprot, Unigene and Ensembl.

Or upload a file: Choose file No file chosen

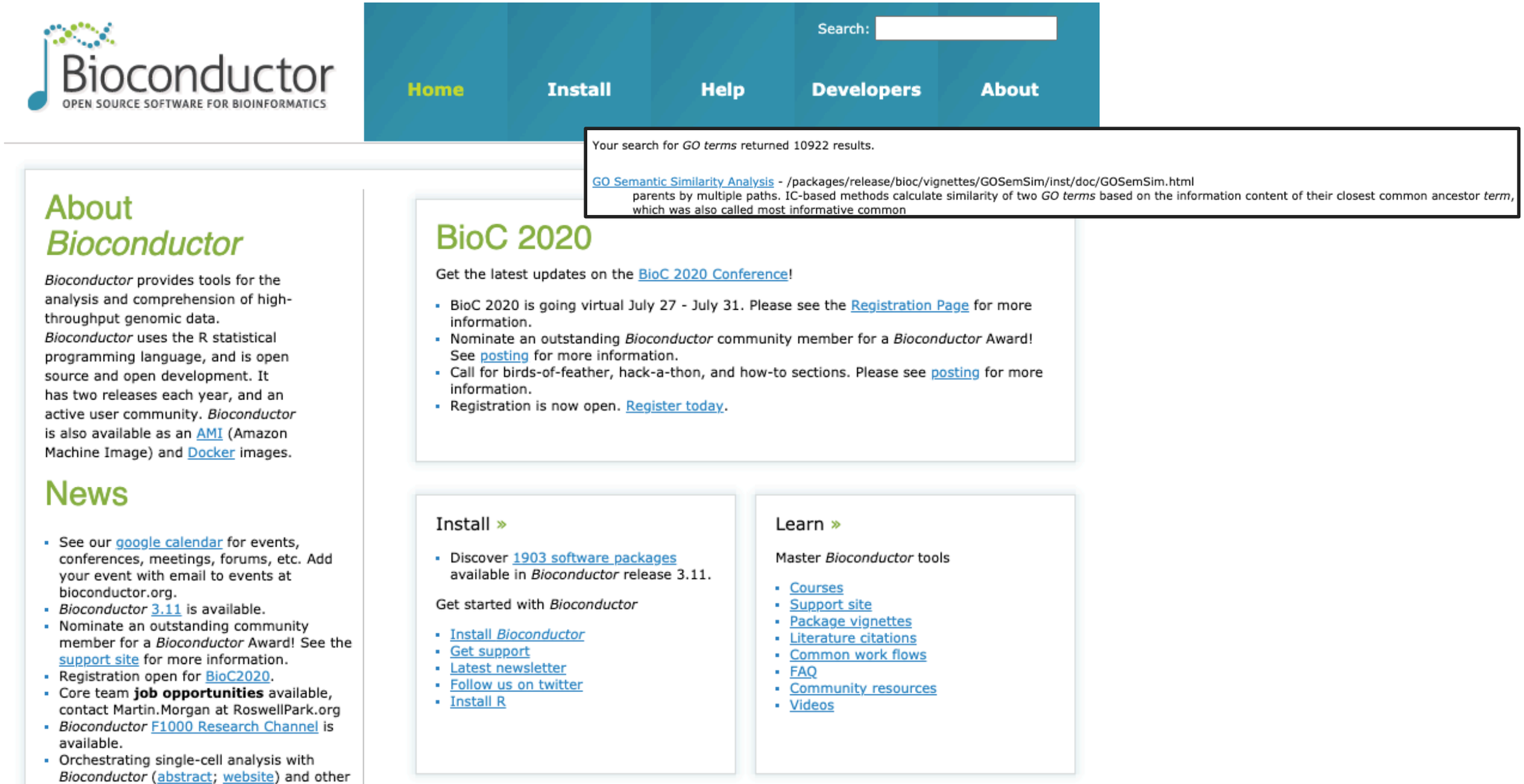
Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2. R packages

Most of the R packages used for the GO term analysis can be found in Bioconductor.

http://www.bioconductor.org/



3.2. R packages

Most of the R packages used for the GO term analysis can be found in Bioconductor.

http://www.bioconductor.org/

Bioconductor Packages for GO Terms:

GO.db	A set of annotation maps describing the entire Gene Ontology
Gostats	Tools for manipulating GO and microarrays
GOSim	functional similarities between GO terms and gene products
GOProfiles	Statistical analysis of functional profiles
TopGO	Enrichment analysis for Gene Ontology

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2.1. GO.db

GO.db http://www.bioconductor.org/packages/2.9/data/annotation/html/GO.db.html

A set of annotation maps describing the entire Gene Ontology

1) GO terms stored in **objects**

GOTERM,

GOBPPARENTS, GOCCPARENTS, GOMFPARENTS

GOBPANCESTOR, GOCCANCESTOR, GOMFANCESTOR

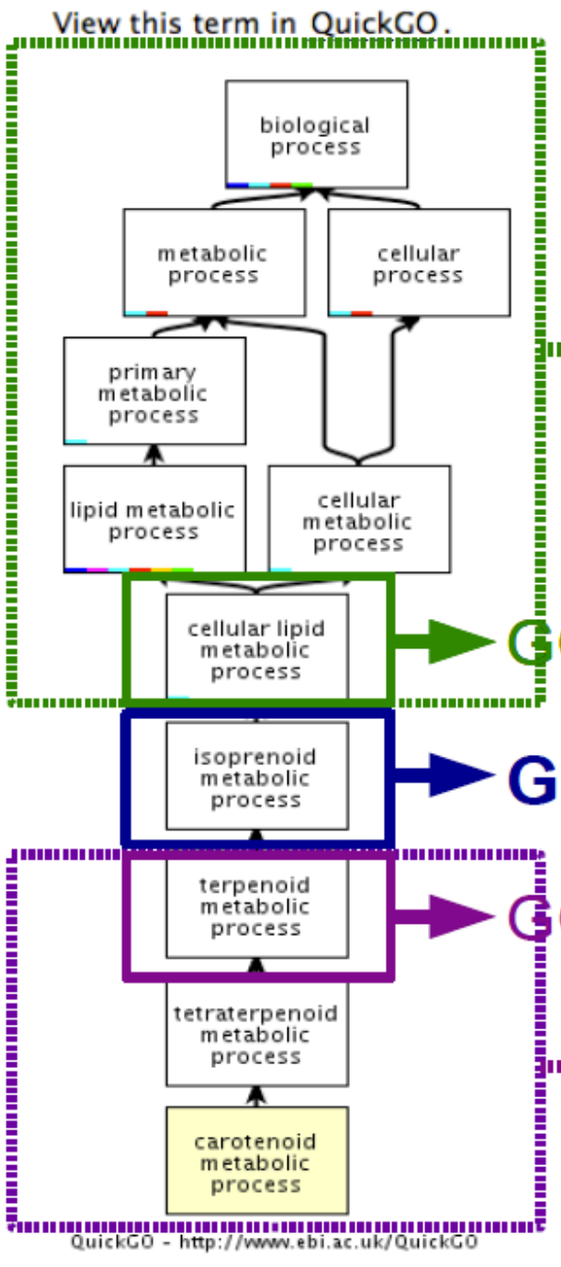
GOBPCHILDREN, GOCCCHILDREN, GOMFCHILDREN

GOBPOFFSPRING, GOCCOFFSPRING, GOMFOFFSPRING

More Information:

http://www.bioconductor.org/packages/2.6/bioc/vignettes/annotate/inst/doc/GOusage.pdf

3.2.1. GO.db



GOBPANCESTOR → GOBPANCESTOR$"GO:0006720"

GOBPPARENTS → GOBPPARENTS$"GO:0006720"

GOTERM → GOTERM$"GO:0006720"

GOBPCHILDREN → GOBPCHILDREN$"GO:0006720"

GOBPOFFSPRING → GOBPOFFSPRING$"GO:0006720"

3.2.1. GO.db

2) Mapping between gene and GO terms stored in **objects or dataframes.**

Bioconductor version 2.8 (Release)

- AnnotationData (593)
  - ChipManufacturer (356)
  - ChipName (190)
  - CustomArray (2)
  - CustomCDF (16)
  - CustomDBSchema (10)
  - FunctionalAnnotation (10)
  - Organism (424)
    - Anopheles_gambiae (4)
    - Apis_mellifera (2)
    - Arabidopsis_thaliana (11)

Packages

AnnotationData > Organism > Arabidopsis_thaliana

- ag.db • agcdf • agprobe • arabidopsis.db0 • ath1121501.db • ath1121501cdf • ath1121501probe
- BSgenome.Athaliana.TAIR.04232008 • BSgenome.Athaliana.TAIR.TAIR9 • hom.At.inp.db • org.At.tair.db

org.At.tair.db

Genome wide annotation for Arabidopsis, primarily based on mapping using TAIR identifiers.

3.2.1. GO.db

2) Mapping between gene and GO terms stored in **objects (annotate package) or dataframes.**

```
> library("org.At.tair.db")
> org.At.tairGO[["AT5G58560"]]
```

Use a list:

```
> org.At.tairGO[["AT5G58560"]][[1]]$Ontology
```

Functions (“annotate” package):

+ getOntology(inlist, gocategorylist)

+ getEvidence(inlist)

```
> getOntology(org.At.tairGO[["AT5G58560"]])
> getEvidence(org.At.tairGO[["AT5G58560"]])
```

Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2.2. TopGO

topGO http://www.bioconductor.org/packages/release/bioc/html/topGO.html

topGO: Enrichment analysis for Gene Ontology

	fisher	ks	t	globaltest	sum
classic	✓	✓	✓	✓	✓
elim	✓	✓	✓	✓	✓
weight	✓	–	–	–	–
weight01	✓	✓	✓	✓	✓
lea	✓	✓	✓	✓	✓
parentchild	✓	–	–	–	–

Table 1: *Algorithms currently supported by* **topGO**.

3.2.2. TopGO

Different algorithms are adequate only to some specific test

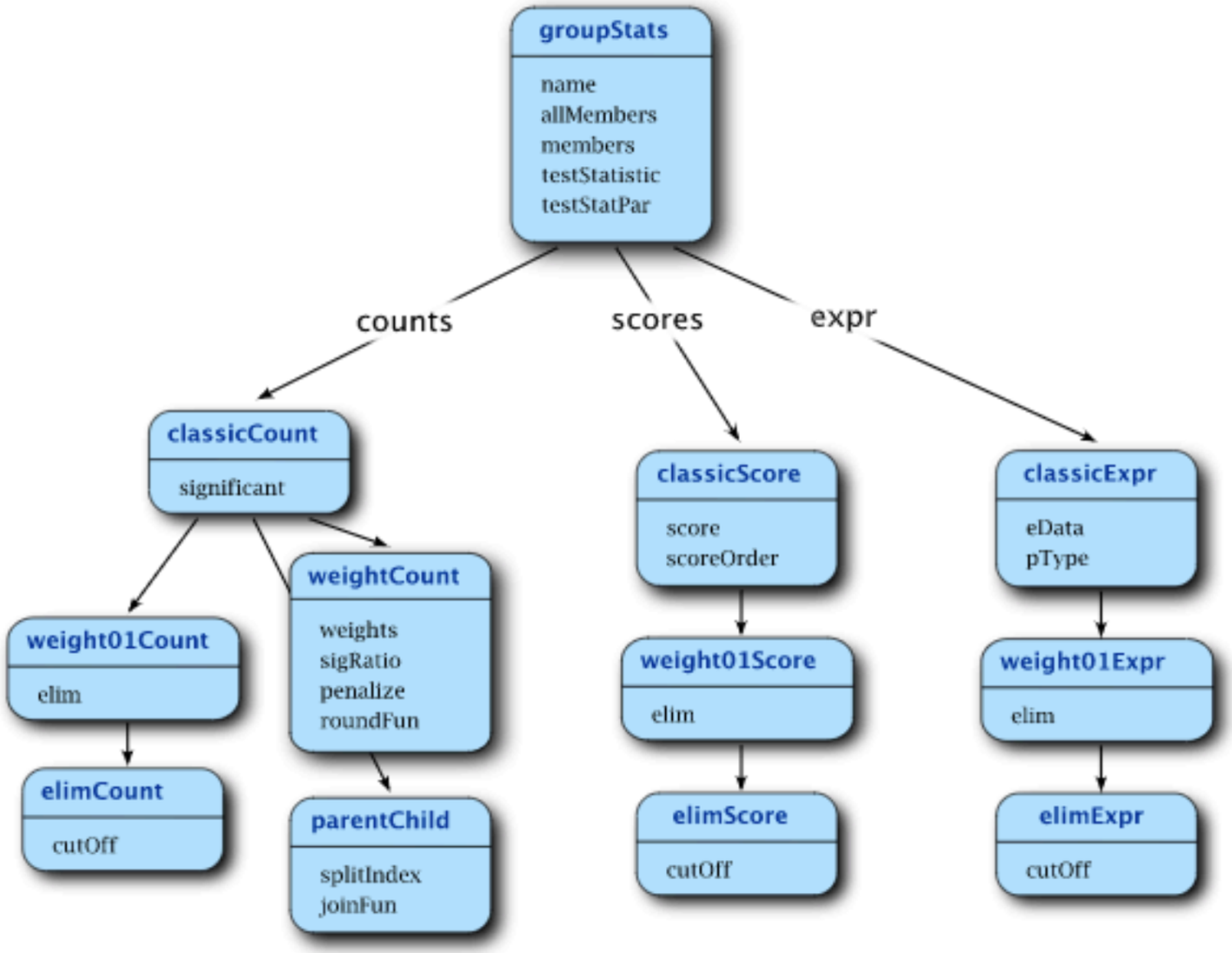


Figure 4: *The test statistics class structure.*

3.2.2. TopGO

Different algorithms are adequate only to some specific test

Methodology:

1) Data preparation:

a) Gene Universe.

b) GO Annotation.

c) Criteria to select interesting genes.

OBJECT:

topGOdata

```
> sampleGOdata <- new("topGOdata",
      description = "Simple session",
      ontology = "BP",
      allGenes = geneList,
      geneSel = topDiffGenes,
      nodeSize = 10,
      annot = annFUN.db,
      affyLib = "hgu95av2.db")
```

- `allGenes` → Gene Universe
- `geneSel` → Selected Genes
- `annot` → Function to map data provided in the annotation data packages
- `affyLib` → Data package

3.2.2. TopGO

Different algorithms are adequate only to some specific test

Methodology:

2) Running the enrinchment test: (**runTest** function)

```
> resultFisher <- runTest(sampleGOdata,
                          algorithm = "classic",
                          statistic = "fisher")

> resultFisher

Description: Simple session
Ontology: BP
'classic' algorithm with the 'fisher' test
672 GO terms scored: 66 terms with p < 0.01
Annotation data:
    Annotated genes: 316
    Significant genes: 50
    Min. no. of genes annotated to a GO: 10
    Nontrivial nodes: 607
```

whichAlgorithms()

whichTest()

```
"fisher"
"ks"
 "t"
 "globaltest"
"sum"
"ks.ties"
```

3.2.2. TopGO

Different algorithms are adequate only to some specific test

Methodology:

3) Analysis of the results: (**GenTable** function)

```
> allRes <- GenTable(sampleGOdata,
                     classicFisher = resultFisher,
                     classicKS = resultKS,
                     orderBy = "classicKS",
                     ranksOf = "classicFisher",
                     topNodes = 10)

> allRes
```

	GO.ID	Term	Annotated	Significant	Expected	Rank in classicFisher	classicFisher	classicKS
1	GO:0007049	cell cycle	204	33	32.28	324	0.48	2.8e-11
2	GO:0022403	cell cycle phase	189	28	29.91	435	0.78	8.2e-11
3	GO:0022402	cell cycle process	192	29	30.38	423	0.73	1.5e-10
4	GO:0000278	mitotic cell cycle	191	28	30.22	442	0.81	2.0e-10
5	GO:0000087	M phase of mitotic cell cycle	185	24	29.27	566	0.96	2.9e-10
6	GO:0000279	M phase	185	24	29.27	567	0.96	2.9e-10
7	GO:0000280	nuclear division	176	18	27.85	601	1.00	3.7e-09
8	GO:0007067	mitosis	176	18	27.85	602	1.00	3.7e-09
9	GO:0048285	organelle fission	176	18	27.85	603	1.00	3.7e-09
10	GO:0006996	organelle organization	190	21	30.06	600	1.00	2.1e-08

3.2.2. TopGO

R Command example:

1- UPLOAD THE ANNOTATION (UNIVERSE)

```
geneID2GO <- readMappings(file =“Genes.GOTerms.txt")
GO2geneID <- inverseList(geneID2GO)
```

2- SELECT THE GROUP OF TARGET GENES

```
DEG_GeneIDList = DEG[DEG$qval < 0.05,2]
```

3- GENERATE THE GO DATA OBJECT

```
geneNames = names(geneID2GO)
geneList = factor(as.integer(geneNames %in% DEG_GeneIDList))
names(geneList) = geneNames

GOData4_BP = new("topGOdata", ontology = "BP", allGenes = geneList, annot =
annFUN.gene2GO, gene2GO = geneID2GO)

GOData_CC = new("topGOdata", ontology = "CC", allGenes = geneList, annot =
annFUN.gene2GO, gene2GO = geneID2GO)

GOData_MF = new("topGOdata", ontology = "MF", allGenes = geneList, annot =
annFUN.gene2GO, gene2GO = geneID2GO)
```

3.2.2. TopGO

R Command example:

```
## 4- RUN THE TEST

resultFis_IAC_BP <- runTest(GOData4_BP, algorithm = "classic", statistic =
"fisher")

resultKS_IAC_BP <- runTest(GOData4_BP, algorithm = "weight01", statistic =
"fisher")

resultWeight_IAC_BP <- runTest(GOData4_BP, algorithm = "elim", statistic =
“ks")
```

```
## 5- PRESENT THE RESULTS INTO A TABLE

allRes_IAC_BP <- GenTable(GOData4_BP, classic = resultFis_IAC_BP, KS =
resultKS_IAC_BP, weight = resultWeight_IAC_BP, orderBy = "weight", ranksOf =
"classic", topNodes = 30)
```

3.2.2. TopGO

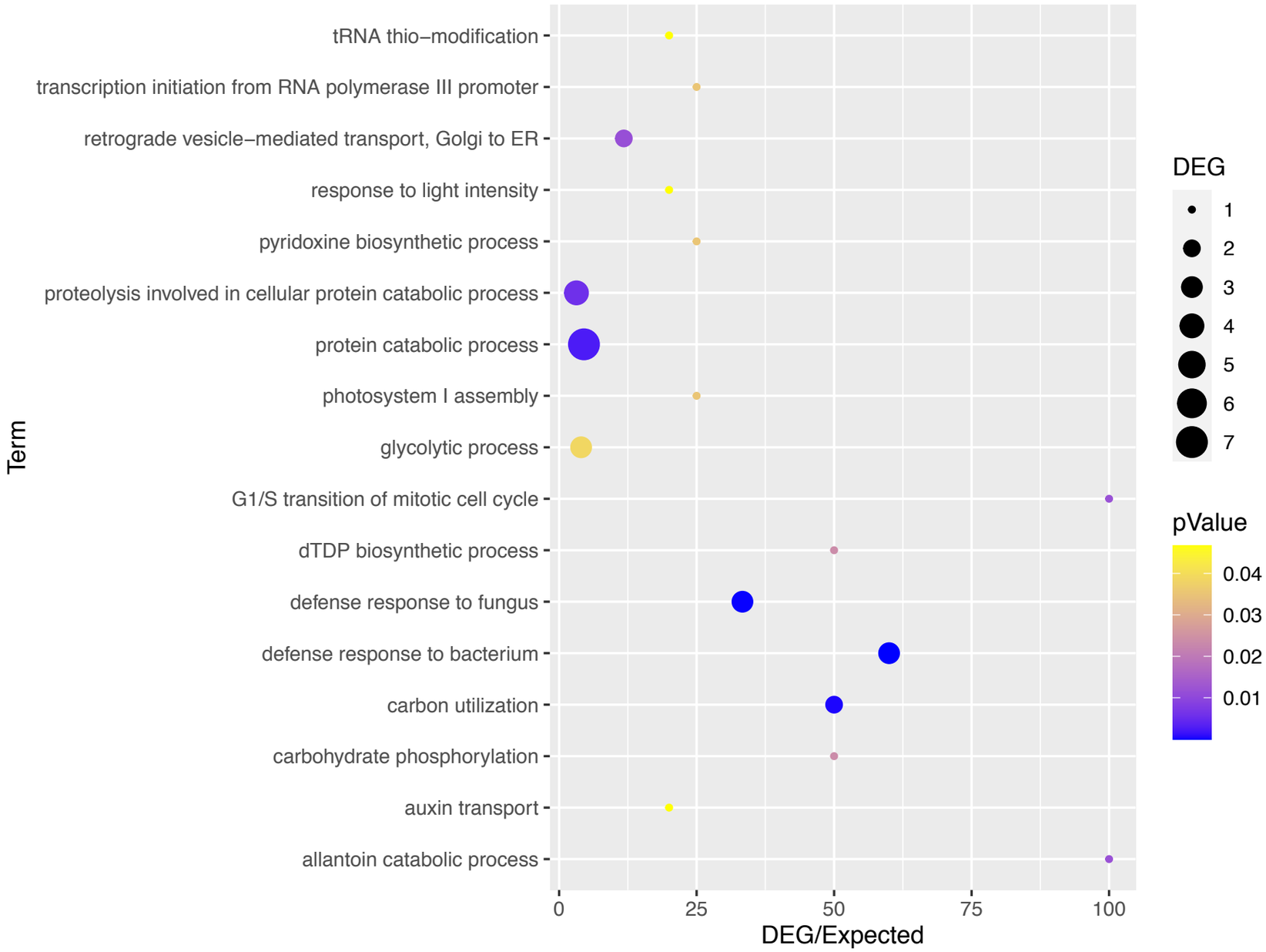
R Command example:

```
## 6- VISUALIZATION (e.g. using GGPLOT)

pVals_Fis_BP = score(resultFis_IAC_BP)[score(resultFis_IAC_BP) <= 0.05]

GOData4DE = termStat(object = GOData4_BP, whichGO = names(pVals_Fis_BP))

GOData4DE$DEG = GOData4DE$Significant

GOData4DE$pValue = pVals_GSEA

GOData4DE$Term = Term(rownames(GOData4DE))

ggplot(GOData4DE, aes(x = DEG/Expected, y = Term)) +
geom_point(aes(color=pValue, size=DEG)) + scale_color_gradient(low="blue",
high="yellow")
```

3.2.2. TopGO

R Command example:

```
## 6- VISUALIZATION (e.g. using GGPLOT)
```



Outline of Topics

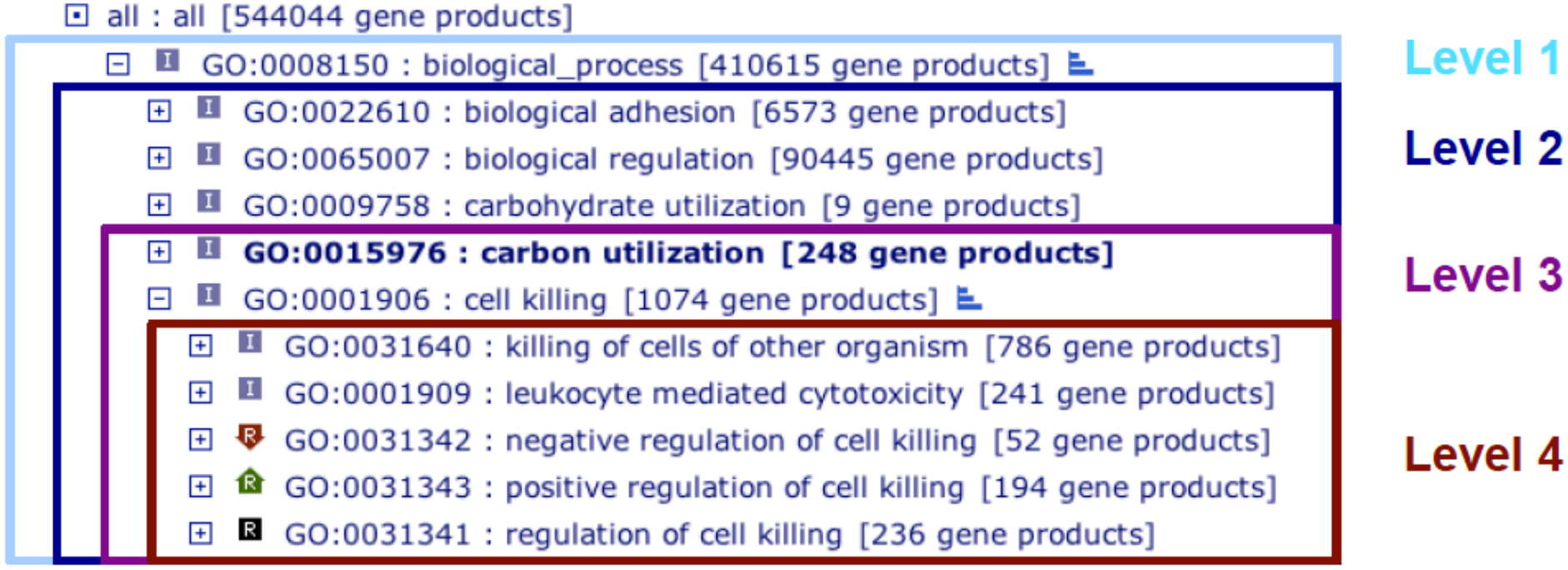
1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2.3. GOProfiles

goProfiles http://www.bioconductor.org/packages/2.8/bioc/html/goProfiles.html

goProfiles: an R package for the statistical analysis of functional profiles

1) Profiles are built by slicing the GO graph at a given level



3.2.3. GOProfiles

2) Functional profile at a given GO level is obtained by counting the number of identifiers having a hit in each category of this level

GO Term	*gene1*	*gene2*	*gene 3*	*gene 4*
GO:0005488	1	1	0	0
GO:0030234	0	1	1	1
GO:0045182	0	0	0	1

Table 2.1: A simple list of 4 identifiers considered at level 2 of the MF ontology illustrates how some genes may have hits in several categories simultaneously.

3) Profiles comparissons:

1. How different or representative of a given gene universe is a given set of genes?
   - Universe: All genes analyzed,
     Gene Set: Differentially expressed genes in a microarray experiment
   - Universe: All genes in a database,
     Gene Set: Arbitrarily selected set of genes

2. How biologically different are two given sets of genes?
   - Differentially expressed genes in two experiments
   - Arbitrarily chosen lists of genes

3.2.3. GOProfiles

Methodology:

1) Data preparation:
   a) Gene GO term map (**object** or **dataframe**).
      Dataframe with 4 columns:
      + GeneID
      + Ontology
      + Evidence
      + GOID

2) Profile creation:
   a) **basicProfile** function

```
BasicProfile( genelist,
              idType = "Entrez",
              onto = "ANY",
              Level = 2,
              orgPackage=NULL,
              anotPackage=NULL,
              ...)
```

- idType → "Entrez" (default), "BiocProbes", "GoTermsFrame"
- orgPackage=NULL → Requested for "Entrez"
- anotPackage=NULL → Requested for "BiocProbes"

3.2.3. GOProfiles

Methodology:

2) Profile creation:

a) **basicProfile** function

```
> printProfiles(bpprofile)

Functional Profile
==================
[1] "BP ontology"
```

	Description	GOID	Frequency
25	cellular component biogen...	GO:0044085	5
14	cellular component organi...	GO:0016043	6
12	cellular process	GO:0009987	122
32	establishment of localiza...	GO:0051234	4
4	immune system process...	GO:0002376	1
31	localization	GO:0051179	4
2	metabolic process	GO:0008152	3
33	multi-organism process...	GO:0051704	2
30	response to stimulus	GO:0050896	4

3.2.3. GOProfiles

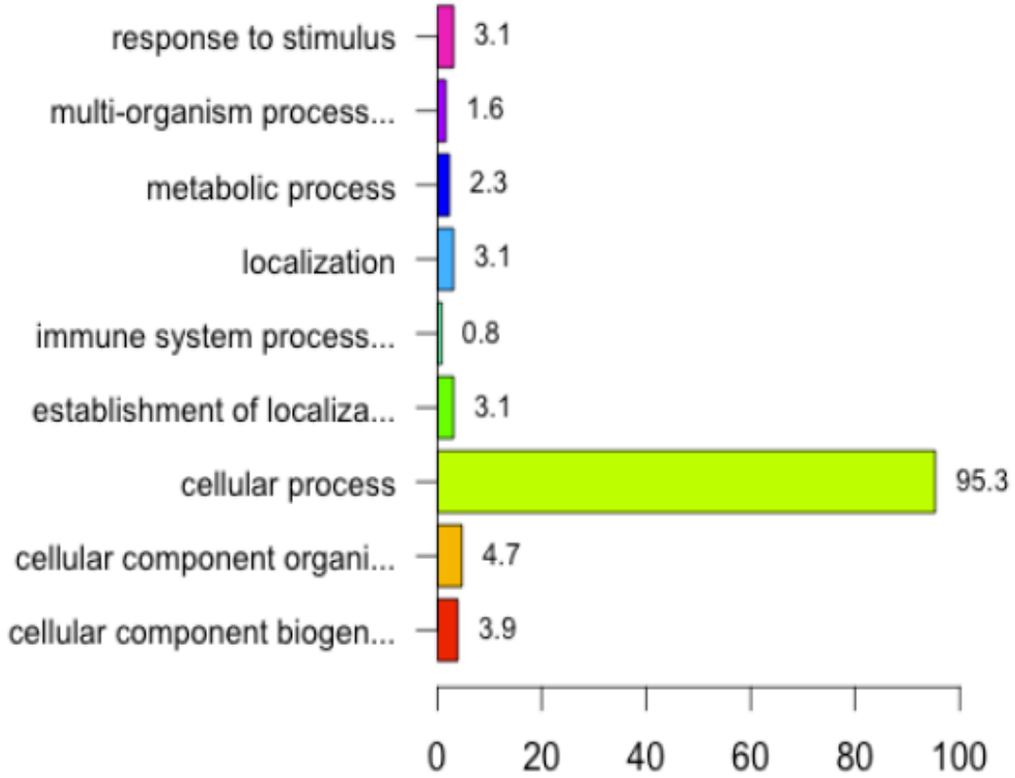
Methodology:

2) Profile creation:

a) **basicProfile** function

```
> plotProfiles(bpprofile)
```

Functional Profile. BP ontology



3.2.3. GOProfiles

Methodology:

2) Profile creation:

b) **expandedProfile** function

Used mainly for comparisons of profiles.

```
ExpandedProfile( genelist,
                 idType = "Entrez",
                 onto = "ANY",
                 Level = 2,
                 orgPackage=NULL,
                 anotPackage=NULL
                 ...)
```

3) Profile comparisons:

- Case I (INCLUSION): One list included in the other.

- Case || (DISJOINT): Non overlapping gene sets

- Case III (INTERSECTION): Overlapping genes

3.2.3. GOProfiles

Methodology:

3) Profile comparisons:

- Case I (INCLUSION): compareProfilesLists()
- Case || (DISJOINT): compareGeneLists()
- Case III (INTERSECTION): compareGeneLists()

```
> comp_ath_genes <- compareGeneLists(
        genelist1=ath_chl_list,
        genelist2=ath_mit_list,
        idType="Entrez", orgPackage="org.At.tair.db",
        onto="BP", level=2)

> print(compSummary(comp_ath_genes))

  Sqr.Euc.Dist  StdErr     pValue     0.95CI.low   0.95CI.up
  0.031401      0.024101   0.005000   -0.015837    0.078639
```

3.2.3. GOProfiles

Methodology:

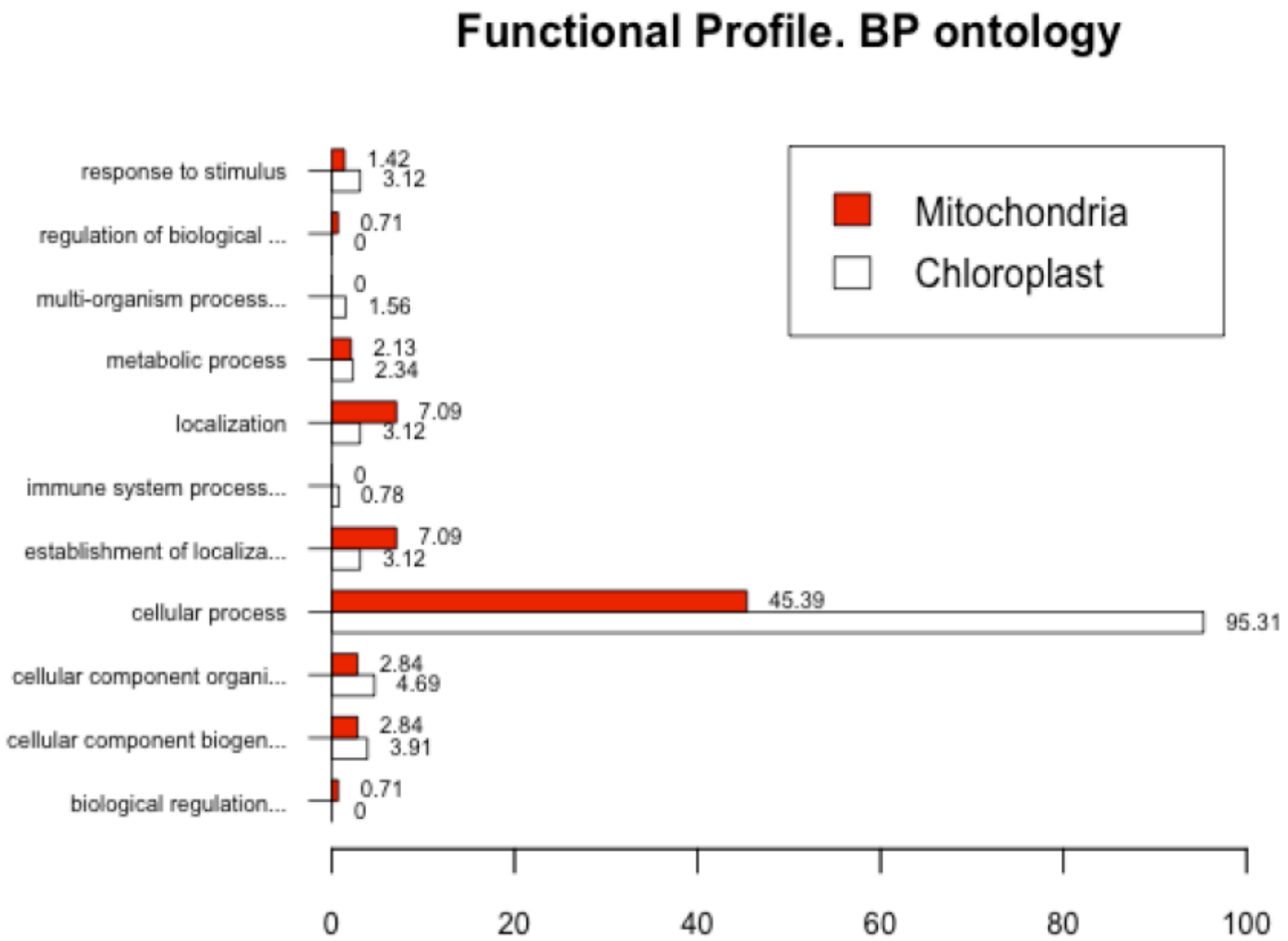
3) Profile comparisons:

```
> basic_mitprof <- basicProfile(ath_mit_list, idType="Entrez",
        onto="BP", level=2, orgPackage="org.At.tair.db",
        empty.cats=TRUE)

> basic_chloprof <- basicProfile(ath_chl_list, idType="Entrez",
        onto="BP", level=2, orgPackage="org.At.tair.db",
        empty.cats=TRUE)

> merged_prof <- mergeProfilesLists(basic_chloprof, basic_mitprof,
        profNames=c("Chloroplast", "Mitochondria"))

> plotProfiles(merged_prof, percentage=TRUE, legend=TRUE)
```

3.2.3. GOProfiles

Methodology:

3) Profile comparisons:



Outline of Topics

1. Basics about GO Terms.
2. GO Enrichment Analysis
3. Tools to analyse GO Terms.
    - 3.1. Web based tools
        - 3.1.1. Gene Ontology
        - 3.1.2. GOrilla
    - 3.2. R packages
        - 3.2.1. GO.db
        - 3.2.2. TopGO
        - 3.2.3. GOProfiles
        - 3.2.4. GOSim

3.2.4. GOSim

GOSim

platforms all | rank 442 / 1905 | posts 0 | in Bioc 6.5 years
build ok | updated before release | dependencies 47

DOI: 10.18129/B9.bioc.GOSim

Computation of functional similarities between GO terms and gene products; GO enrichment analysis

Bioconductor version: Release (3.11)

This package implements several functions useful for computing similarities between GO terms and gene products based on their GO annotation. Moreover it allows for computing a GO enrichment analysis

Author: Holger Froehlich <frohlich at bit.uni-bonn.de>

Maintainer: Holger Froehlich <frohlich at bit.uni-bonn.de>

Citation (from within R, enter `citation("GOSim")`):

> Important note to the maintainer of the GOSim package: An error occured while trying to generate the citation from the CITATION file. This typically occurs when the file contains R code that relies on the package to be installed e.g. it contains calls to things like `packageVersion()` or `packageDate()` instead of using `meta$Version` or `meta$Date`. See R documentation for more information.

Installation

To install this package, start R (version "4.0") and enter:

```
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install("GOSim")
```

For older versions of R, please refer to the appropriate Bioconductor release.