Using *clusterProfiler* to identify and compare functional profiles of gene lists

Guangchuang Yu
School of Biological Sciences
The University of Hong Kong, Hong Kong SAR, China
email: guangchuangyu@gmail.com

October 13, 2014

Contents

1 Introduction

In recently years, high-throughput experimental techniques such as microarray, RNA-Seq and mass spectrometry can detect cellular moleculars at systems-level. These kinds of analyses generate huge quantitaties of data, which need to be given a biological interpretation. A commonly used approach is via clustering in the gene dimension for grouping different genes based on their similarities [1].

To search for shared functions among genes, a common way is to incorporate the biological knowledge, such as Gene Ontology (GO) and Kyoto Encyclopedia of genes and Genomes (KEGG), for identifying predominant biological themes of a collection of genes.

After clustering analysis, researchers not only want to determine whether there is a common theme of a particular gene cluster, but also to compare the biological themes among gene clusters. The manual step to choose interesting clusters followed by enrichment analysis on each selected cluster is slow and tedious. To bridge this gap, we designed *clusterProfiler* [2], for comparing and visualizing functional profiles among gene clusters.

2 Citation

Please cite the following articles when using *clusterProfiler*.

G Yu, LG Wang, Y Han, QY He. clusterProfiler: an R package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*. 2012, 16(5), 284-287.

3 Supported organisms

At present, *clusterProfiler* about 20 species as shown below:

- *Arabidopsis*
- *Anopheles*
- *Bovine*
- *Canine*
- *Chicken*
- *Chimp*
- *E coli strain K12*

- *E coli strain Sakai*
- *Fly*
- *Human*
- *Malaria*
- *Mouse*
- *Pig*
- *Rat*
- *Rhesus*
- *Worm*
- *Xenopus*
- *Yeast*
- *Zebrafish*

These species are all supported by GO and KEGG analyses.

GO analyses also support *Coelicolor* and *Gondii*.

4 Gene Ontology Classification

In *clusterProfiler*, `groupGO` is designed for gene classification based on GO distribution at a specific level.

```
require(DOSE)
data(geneList)
gene <- names(geneList)[abs(geneList) > 2]
head(gene)

## [1] "4312"  "8318"  "10874" "55143" "55388" "991"

ggo <- groupGO(gene = gene, organism = "human", ont = "BP",
    level = 3, readable = TRUE)
head(summary(ggo))

##                    ID                                Description Count GeneRatio
## GO:0019953 GO:0019953                        sexual reproduction    10    10/138
## GO:0019954 GO:0019954                      asexual reproduction     0     0/138
## GO:0032504 GO:0032504      multicellular organism reproduction    11    11/138
## GO:0032505 GO:0032505 reproduction of a single-celled organism     0     0/138
```

```
## GO:0051321 GO:0051321                        meiotic cell cycle     5     5/138
## GO:0006807 GO:0006807      nitrogen compound metabolic process    76    76/138
##
## GO:0019953
## GO:0019954
## GO:0032504
## GO:0032505
## GO:0051321
## GO:0006807 CDC45/MCM10/S100A9/FOXM1/KIF23/CENPE/MYBL2/S100A8/TOP2A/NCAPH/E2F8/CX
```

5 Enrichment Analysis

5.1 Hypergeometric model

Enrichment analysis [3] is a widely used approach to identify biological themes. Here we implement hypergeometric model to assess whether the number of selected genes associated with disease is larger than expected.

To determine whether any terms annotate a specified list of genes at frequency greater than that would be expected by chance, *clusterProfiler* calculates a p-value using the hypergeometric distribution:

$$p = 1 - \sum_{i=0}^{k-1} \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{n}}$$

In this equation, *N* is the total number of genes in the background distribution, *M* is the number of genes within that distribution that are annotated (either directly or indirectly) to the node of interest, *n* is the size of the list of genes of interest and *k* is the number of genes within that list which are annotated to the node. The background distribution by default is all the genes that have annotation.

P-values were adjusted for multiple comparison, and q-values were also calculated for FDR control.

5.2 Gene set enrichment analysis

A common approach in analyzing gene expression profiles was identifying differential expressed genes that are deemed interesting. The enrichment analysis we demonstrated previous were based on these differential expressed genes. This approach will find genes where the difference is large, but it will not detect a situation where the difference is small, but evidenced in coordinated way in a set of related genes. Gene Set Enrichment Analysis (GSEA) [4] directly addresses this limitation. All genes can be used in GSEA; GSEA aggregates the per gene statistics across genes within a gene set, therefore making it possible to detect situations where all genes in a predefined set change in a small but coordinated

way. Since it is likely that many relevant phenotypic differences are manifested by small but consistent changes in a set of genes.

Genes are ranked based on their phenotypes. Given a priori defined set of gens *S* (e.g., genes shareing the same *GO* or *KEGG* category), the goal of GSEA is to determine whether the members of *S* are randomly distributed throughout the ranked gene list (*L*) or primarily found at the top or bottom.

There are three key elements of the GSEA method:

- Calculation of an Enrichment Score.
  The enrichment score (*ES*) represent the degree to which a set *S* is over-represented at the top or bottom of the ranked list *L*. The score is calculated by walking down the list *L*, increasing a running-sum statistic when we encounter a gene in *S* and decreasing when it is not. The magnitude of the increment depends on the gene statistics (e.g., correlation of the gene with phenotype). The *ES* is the maximum deviation from zero encountered in the random walk; it corresponds to a weighted Kolmogorov-Smirnov-like statistic [4].
- Esimation of Significance Level of *ES*.
  The *p-value* of the *ES* is calculated using permutation test. Specifically, we permute the gene labels of the gene list *L* and recompute the *ES* of the gene set for the permutated data, which generate a null distribution for the *ES*. The *p-value* of the observed ES is then calculated relative to this null distribution.
- Adjustment for Multiple Hypothesis Testing.
  When the entire *GO* or *KEGG* gene sets is evaluated, *clusterProfiler* adjust the estimated significance level to account for multiple hypothesis testing and also *q-values* were calculated for FDR control.

5.3 GO enrichment analysis

```
ego <- enrichGO(gene = gene, universe = names(geneList),
    organism = "human", ont = "CC", pvalueCutoff = 0.01,
    readable = TRUE)
head(summary(ego))
```

```
##                    ID                                Description GeneRatio
## GO:0005819 GO:0005819                                    spindle    24/196
## GO:0015630 GO:0015630                  microtubule cytoskeleton    37/196
## GO:0005876 GO:0005876                       spindle microtubule    10/196
## GO:0000793 GO:0000793                      condensed chromosome    16/196
## GO:0000779 GO:0000779 condensed chromosome, centromeric region    12/196
## GO:0044430 GO:0044430                        cytoskeletal part    43/196
##               BgRatio   pvalue p.adjust   qvalue
```

```
## GO:0005819  211/11884 6.96e-14 6.68e-12 4.39e-12
## GO:0015630  710/11884 3.09e-10 1.49e-08 9.77e-09
## GO:0005876   40/11884 6.53e-10 2.09e-08 1.37e-08
## GO:0000793  141/11884 1.29e-09 3.09e-08 2.03e-08
## GO:0000779   73/11884 2.21e-09 4.24e-08 2.79e-08
## GO:0044430 1005/11884 4.18e-09 6.68e-08 4.40e-08
##
## GO:0005819
## GO:0015630                                        KIF20A/TACC3/CENPE/CHEK1/KIF18B/
## GO:0005876
## GO:0000793
## GO:0000779
## GO:0044430 KIF20A/TACC3/CENPE/CHEK1/KIF18B/SKA1/ABLIM3/TPX2/PSD3/KIF4A/ASPM/AK5/
##            Count
## GO:0005819    24
## GO:0015630    37
## GO:0005876    10
## GO:0000793    16
## GO:0000779    12
## GO:0044430    43
```

```
ego2 <- gseGO(geneList = geneList, organism = "human",
    ont = "CC", nPerm = 100, minGSSize = 120, pvalueCutoff = 0.01,
    verbose = FALSE)
head(summary(ego2))
```

```
##                    ID                    Description setSize enrichmentScore
## GO:0000228 GO:0000228              nuclear chromosome     291           0.414
## GO:0000775 GO:0000775 chromosome, centromeric region     133           0.636
## GO:0000785 GO:0000785                      chromatin     300           0.341
## GO:0000790 GO:0000790               nuclear chromatin     175           0.318
## GO:0000793 GO:0000793           condensed chromosome     141           0.625
## GO:0005575 GO:0005575             cellular_component   11479           0.201
##            pvalue p.adjust qvalues
## GO:0000228      0        0       0
## GO:0000775      0        0       0
## GO:0000785      0        0       0
## GO:0000790      0        0       0
## GO:0000793      0        0       0
## GO:0005575      0        0       0
```

5.4 KEGG pathway enrichment analysis

```
kk <- enrichKEGG(gene = gene, organism = "human", pvalueCutoff = 0.01,
    readable = TRUE)
head(summary(kk))
```

```
##                ID                                Description GeneRatio  BgRatio
## hsa04110 hsa04110                                  Cell cycle     11/74 128/5894
## hsa04114 hsa04114                              Oocyte meiosis     10/74 114/5894
## hsa03320 hsa03320                   PPAR signaling pathway      7/74  70/5894
## hsa04914 hsa04914 Progesterone-mediated oocyte maturation      6/74  87/5894
## hsa04062 hsa04062              Chemokine signaling pathway      8/74 189/5894
## hsa04060 hsa04060  Cytokine-cytokine receptor interaction      9/74 265/5894
##             pvalue p.adjust   qvalue
## hsa04110 4.31e-07 3.02e-06 4.54e-07
## hsa04114 1.25e-06 4.38e-06 6.59e-07
## hsa03320 2.35e-05 5.49e-05 8.25e-06
## hsa04914 7.21e-04 1.26e-03 1.90e-04
## hsa04062 2.37e-03 3.32e-03 5.00e-04
## hsa04060 5.58e-03 6.51e-03 9.79e-04
##                                                                 geneID Count
## hsa04110 CDC45/CDC20/CCNB2/CCNA2/CDK1/MAD2L1/TTK/CHEK1/CCNB1/MCM5/PTTG1    11
## hsa04114     CDC20/CCNB2/CDK1/MAD2L1/CALML5/AURKA/CCNB1/PTTG1/ITPR1/PGR    10
## hsa03320                      MMP1/FADS2/ADIPOQ/PCK1/FABP4/HMGCS2/PLIN1     7
## hsa04914                                CCNB2/CCNA2/CDK1/MAD2L1/CCNB1/PGR     6
## hsa04062            CXCL10/CXCL13/CXCL11/CXCL9/CCL18/CCL8/CXCL14/CX3CR1     8
## hsa04060      CXCL10/CXCL13/CXCL11/CXCL9/CCL18/IL1R2/CCL8/CXCL14/CX3CR1     9
```

```
kk2 <- gseKEGG(geneList = geneList, organism = "human",
    nPerm = 100, minGSSize = 120, pvalueCutoff = 0.01,
    verbose = FALSE)
head(summary(kk2))
```

```
##                ID                                Description setSize
## hsa01100 hsa01100                        Metabolic pathways     920
## hsa04062 hsa04062              Chemokine signaling pathway     166
## hsa04510 hsa04510                          Focal adhesion     193
## hsa03013 hsa03013                           RNA transport     124
## hsa04145 hsa04145                               Phagosome     136
## hsa04060 hsa04060 Cytokine-cytokine receptor interaction     233
##          enrichmentScore pvalue p.adjust qvalues
## hsa01100           0.236      0        0       0
## hsa04062           0.383      0        0       0
## hsa04510          -0.446      0        0       0
## hsa03013           0.427      0        0       0
## hsa04145           0.313      0        0       0
## hsa04060           0.339      0        0       0
```

5.5 DO enrichment analysis

Disease Ontology (DO) enrichment analysis is implemented in *DOSE*, please refer to the package vignettes. The `enrichDO` function is very useful for identifying disease association of interesting genes, and function `gseAnalyzer` function is designed for gene set enrichment analysis of *DO*.

5.6 Reactome pathway enrichment analysis

With the demise of KEGG (at least without subscription), the KEGG pathway data in Bioconductor will not update and we encourage user to analyze pathway using *ReactomePA* which use Reactome as a source of pathway data. The function call of `enrichPathway` and `gsePathway` in *ReactomePA* is consistent with `enrichKEGG` and `gseKEGG`.

5.7 Function call

The function calls of `groupGO`, `enrichGO`, `enrichKEGG`, `enrichDO` and `enrichPathway` are consistent. The input parameters of *gene* is a vector of entrezgene (for human and mouse) or ORF (for yeast) IDs, and *organism* should be supported species (please refer to the manual of the specific function).

For gene set enrichment analysis, the function of `gseGO`, `gseKEGG`, `gseAnalyzer` and `gsePathway` need extra paramter *nPerm* to specify the permutation number.

For GO analysis, *ont* must be assigned to one of "BP", "MF", and "CC" for biological process, molecular function and cellular component, respectively. In `groupGO`, the *level* specify the GO level for gene projection.

In enrichment analysis, the *pvalueCutoff* is to restrict the result based on their pvalues and the adjusted p values. *Q-values* were also calculated for controlling false discovery rate (FDR).

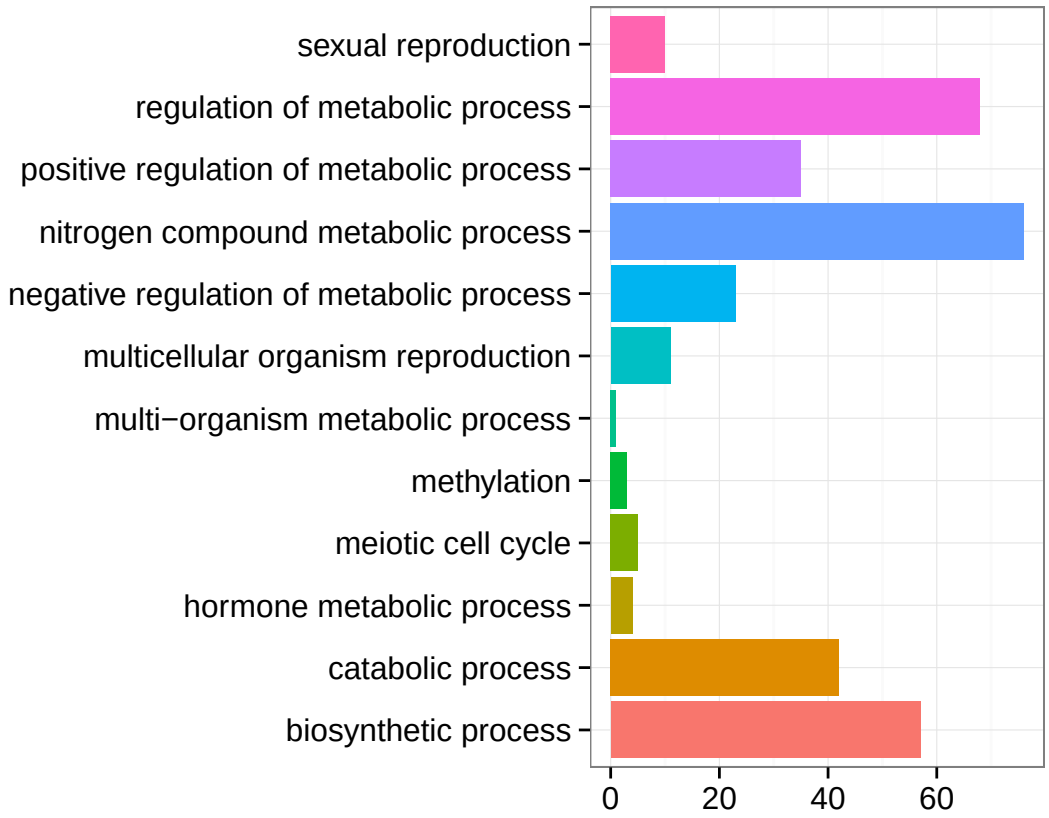
The *readable* is a logical parameter to indicate the input gene IDs will map to gene symbols or not.

5.8 Visualization

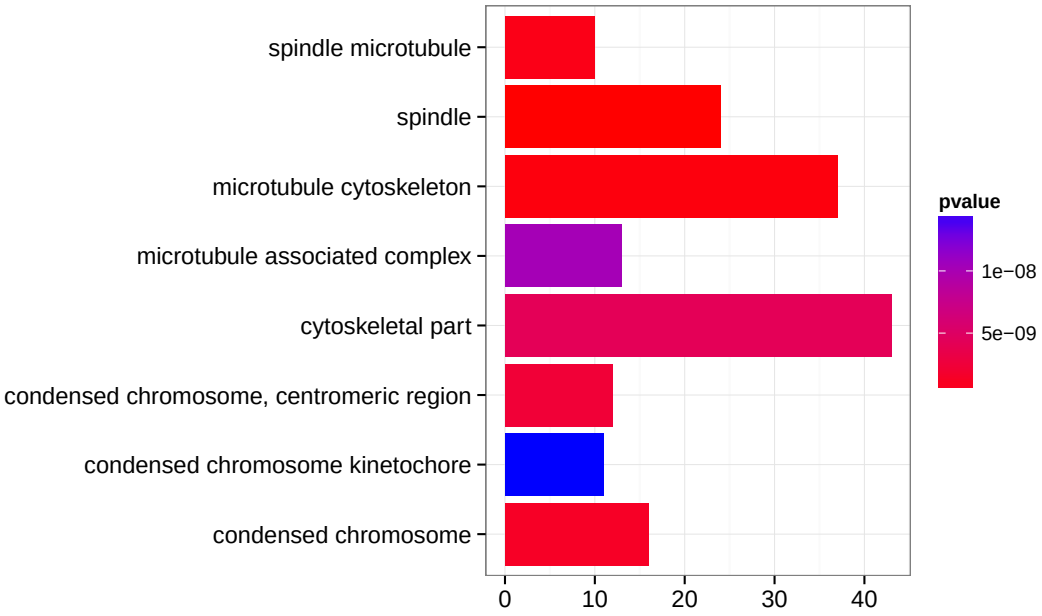
The output of `groupGO`, `enrichGO` and `enrichKEGG` can be visualized by bar plot, enrichment map and category-gene-network plot. It is very common to visualize the enrichment result in bar or pie chart. We believe the pie chart is misleading and only provide bar chart.

5.8.1 barplot

```
barplot(ggo, drop = TRUE, showCategory = 12)
```




```
barplot(ego, showCategory = 8)
```



5.8.2 enrichMap

Enrichment map can be viusalized by `enrichMap`, which support results obtained from hypergeometric test and gene set enrichment analysis.

```
enrichMap(ego)
```

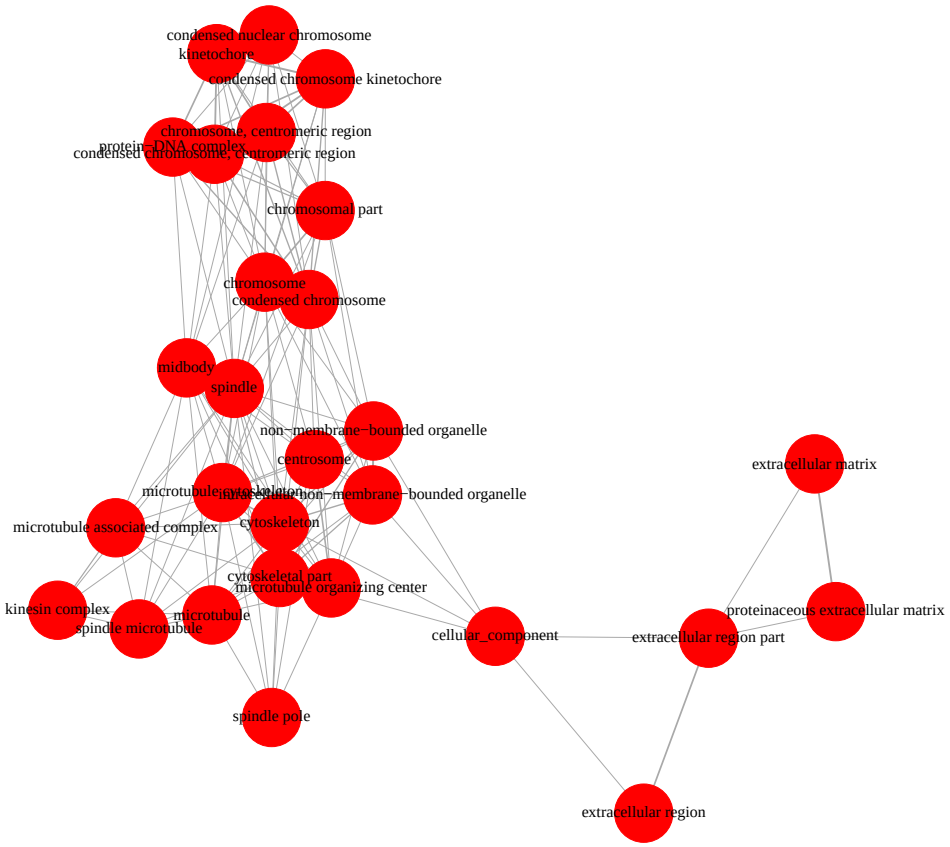


Figure 1: enrichment map of enrichment result

```
enrichMap(ego2)
```

5.8.3 cnetplot

In order to consider the potentially biological complexities in which a gene may belong to multiple annotation categories and provide information of numeric changes if available, we developed `cnetplot` function to extract the complex association.

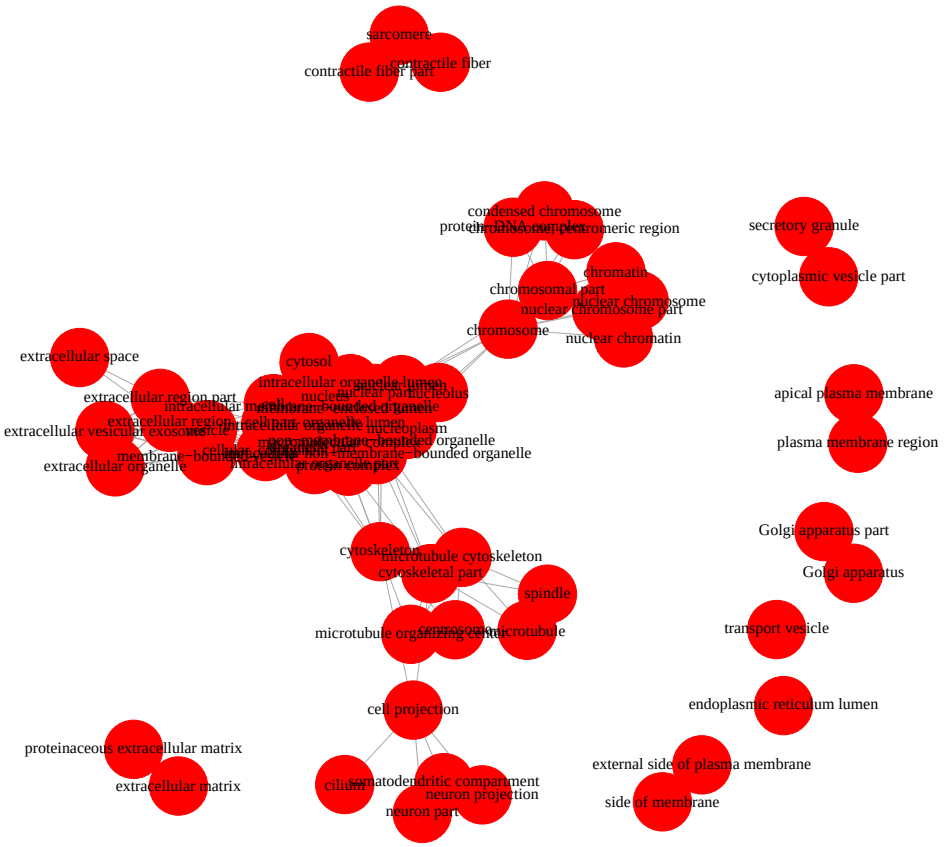
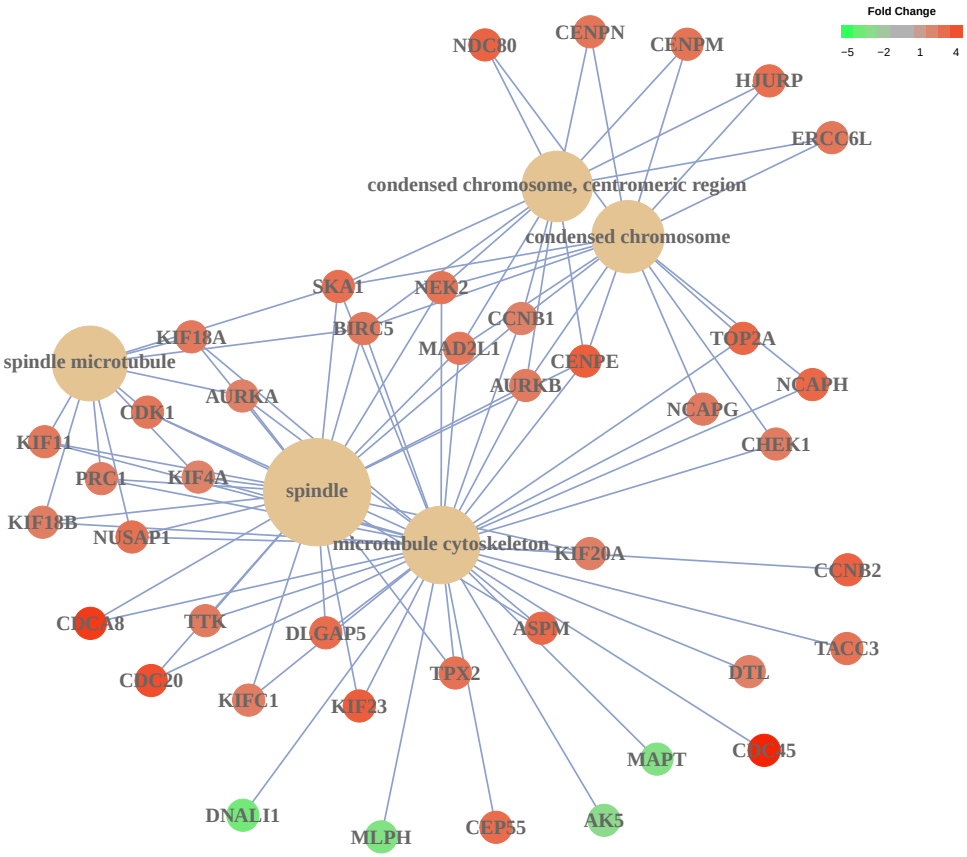
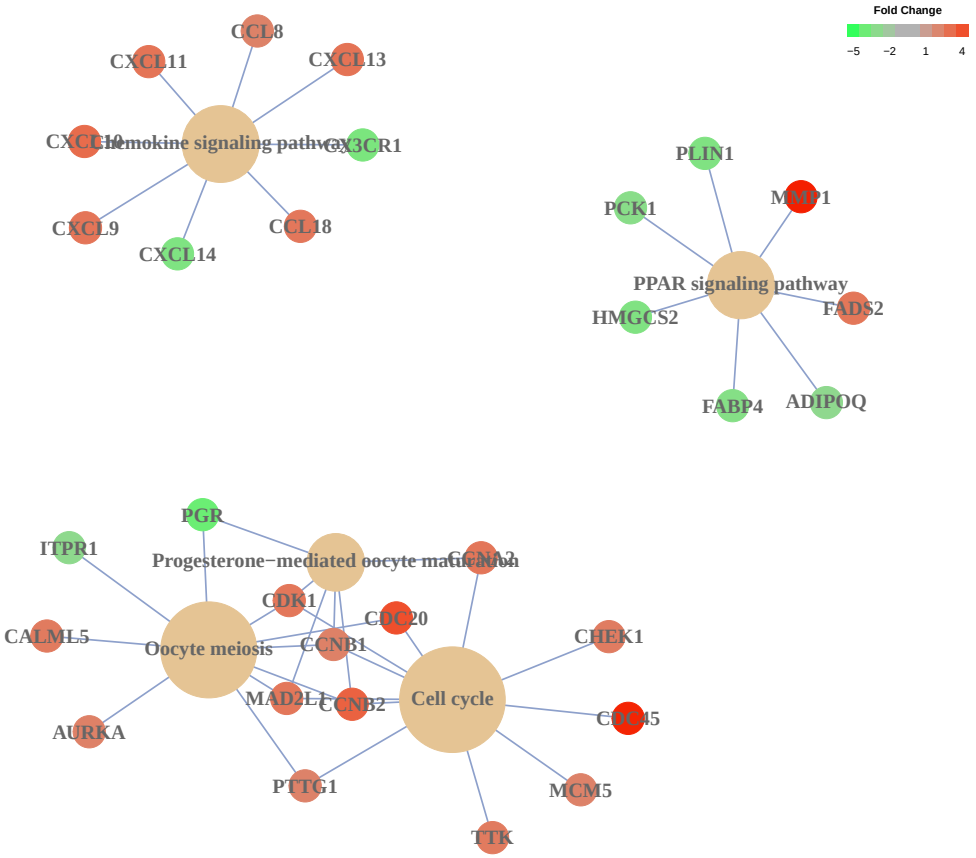


Figure 2: enrichment map of gsea result

```
cnetplot(ego, categorySize = "pvalue", foldChange = geneList)
```




```
cnetplot(kk, categorySize = "geneNum", foldChange = geneList)
```



5.8.4 gseaplot

Running score of gene set enrichment analysis and its association of phenotype can be visualized by `gseaplot`.

```
gseaplot(kk2, geneSetID = "hsa04145")
```

5.8.5 pathview from pathview package

clusterProfiler users can also use `pathview` from the *pathview* [5] to visualize KEGG pathway.

The following example illustrate how to visualize ”hsa04110” pathway, which was enriched in our previous analysis.

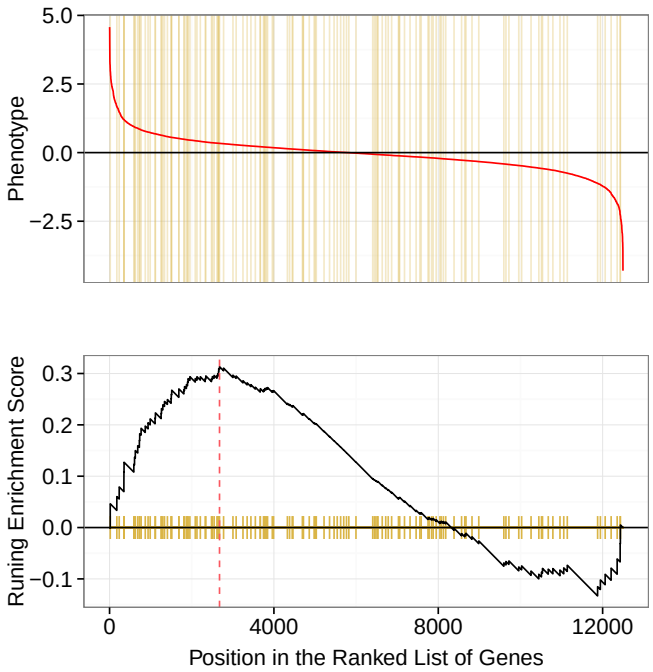


Figure 3: plotting gsea result

```
require(pathview)
hsa04110 <- pathview(gene.data = geneList, pathway.id = "hsa04110",
    species = "hsa", limit = list(gene = max(abs(geneList)),
        cpd = 1))

## Info:  Downloading xml files for hsa04110, 1/1 pathways..
## Info:  Downloading png files for hsa04110, 1/1 pathways..
## Info:  Working in directory /tmp/RtmpTxlu9m/Rbuild5511b0853e2/clusterProfiler/
## Info:  Writing image file hsa04110.pathview.png
```

For further information, please refer to the vignette of *pathview* [5].

6 Biological theme comparison

clusterProfiler was developed for biological theme comparison, and it provides a function, `compareCluster`, to automatically calculate enriched functional categories of each gene clusters.

```
data(gcSample)
ck <- compareCluster(geneCluster = gcSample, fun = "enrichKEGG")
plot(ck)
```

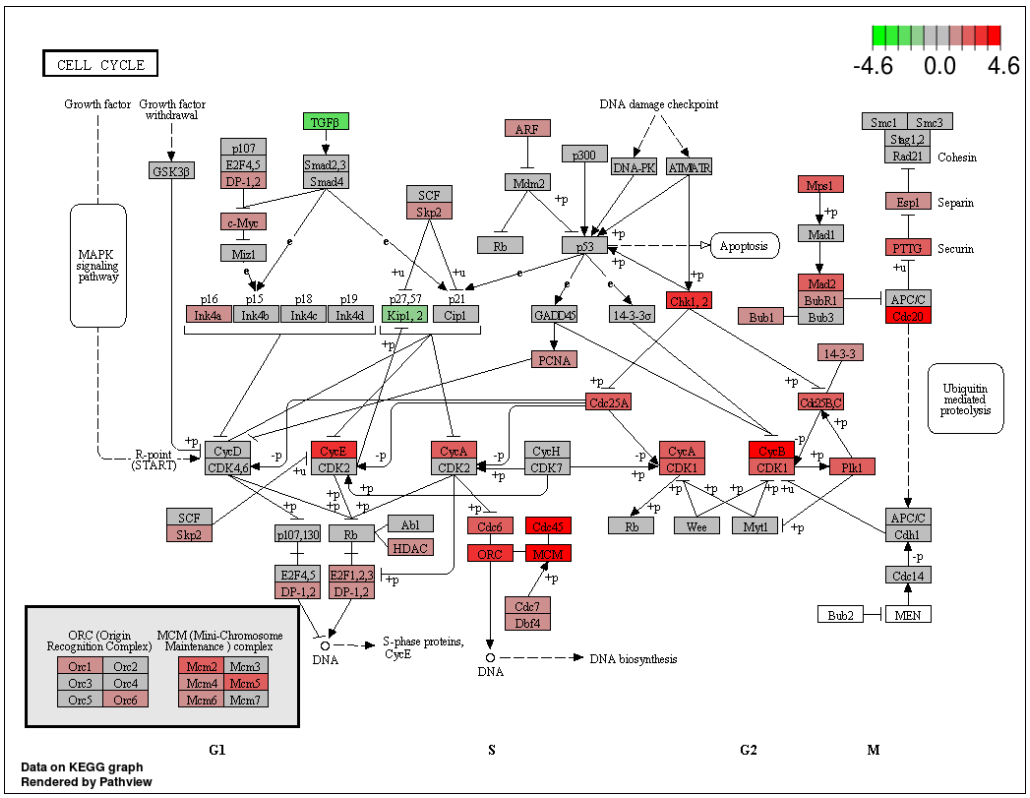
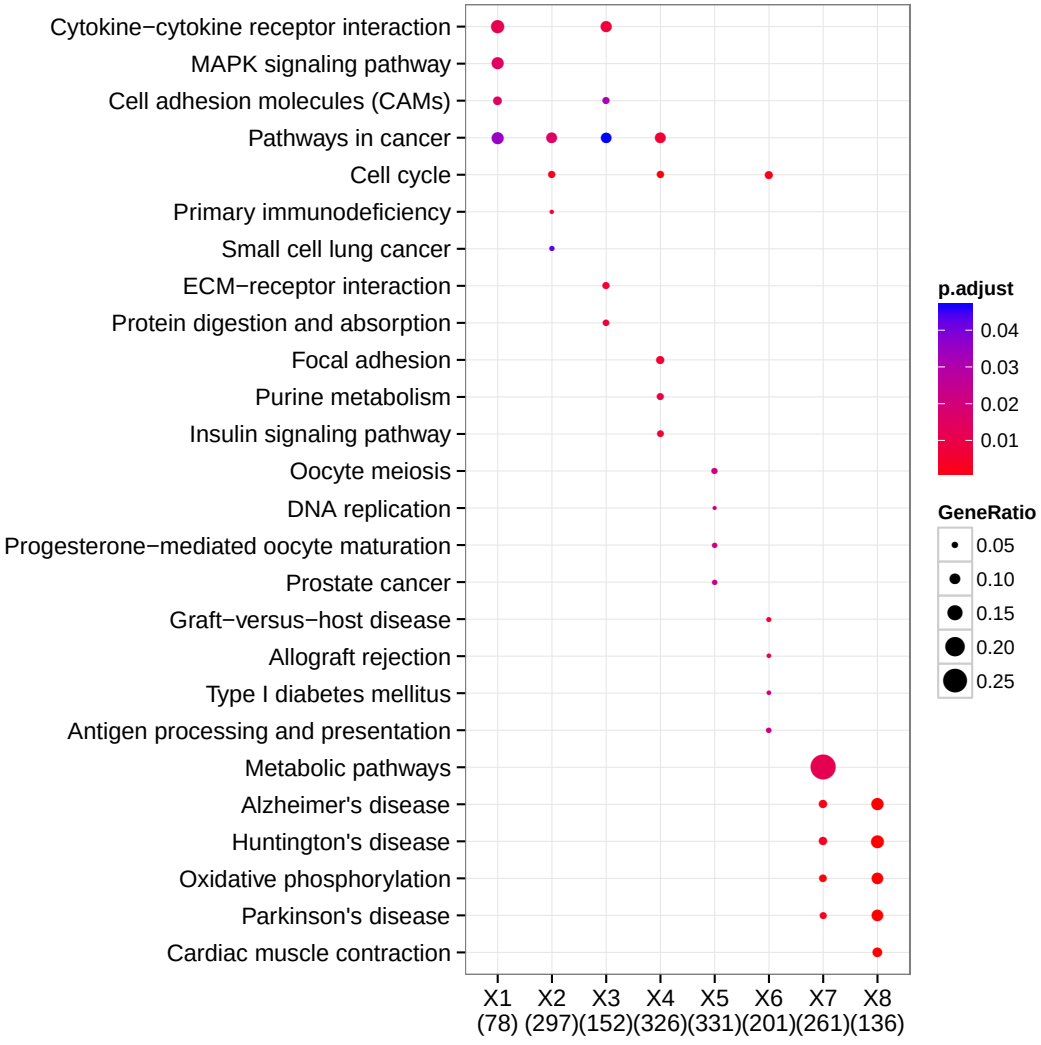


Figure 4: visualize KEGG pathway using pathview



By default, only top 5 (most significant) categories of each cluster was plotted. User can changes the parameter *showCategory* to specify how many categories of each cluster to be plotted, and if *showCategory* was set to *NULL*, the whole result will be plotted.

The dot sizes were based on their corresponding row percentage by default, and user can set the parameter *by* to "count" to make the comparison based on gene counts. The parameter *by* can also set to "rowPercentage" to normalize the dot sizes, since some categories may contain a large number of genes, and make the dot sizes of those small categories too small to compare. The default parameter *by* is setting to "geneRatio", which corresponding to the "GeneRatio" column of the output. To provide the full information, we also provide number of identified genes in each category (numbers in parentheses) when *by* is setting to "rowPercentage" and number of gene clusters in each cluster label (numbers in parentheses) when *by* is setting to "geneRatio", as shown in Figure 3. If the dot sizes were based on "count", the row numbers will not shown.

The p-values indicate that which categories are more likely to have biological meanings. The dots in the plot are color-coded based on their corresponding p-values. Color gradient ranging from red to blue correspond to in order of increasing p-values. That is, red indicate low p-values (high enrichment), and blue indicate high p-values (low enrichment). P-values and adjusted p-values were filtered out by the threshold giving by parameter *pvalueCutoff*, and FDR can be estimated by *qvalue*.

User can refer to the example in [2]; we analyzed the publicly available expression dataset of breast tumour tissues from 200 patients (GSE11121, Gene Expression Omnibus) [6]. We identified 8 gene clusters from differentially expressed genes, and using `compareCluster` to compare these gene clusters by their enriched biological process.

Another example was shown in [7], we calculated functional similarities among viral miRNAs using method described in [8], and compared significant KEGG pathways regulated by different viruses using `compareCluster`.

The comparison function was designed as a general-package for comparing gene clusters of any kind of ontology associations, not only `groupGO`, `enrichGO`, and `enrichKEGG` this package provided, but also other biological and biomedical ontologies, for instance, `enrichDO` from *DOSE* and `enrichPathway` from *ReactomePA* work fine with `compareCluster` for comparing biological themes in disease and reactome pathway perspective. More details can be found in the vignettes of *DOSE* and *ReactomePA*.

7 Session Information

The version number of R and packages loaded for generating the vignette were:

- R version 3.1.1 Patched (2014-09-25 r66681), `x86_64-unknown-linux-gnu`

- Locale: LC_CTYPE=en_US.UTF-8, LC_NUMERIC=C, LC_TIME=en_US.UTF-8, LC_COLLATE=C, LC_MONETARY=en_US.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_US.UTF-8, LC_NAME=C, LC_ADDRESS=C, LC_TELEPHONE=C, LC_MEASUREMENT=en_US.UTF-8, LC_IDENTIFICATION=C
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: AnnotationDbi 1.28.0, Biobase 2.26.0, BiocGenerics 0.12.0, DBI 0.3.1, DOSE 2.4.0, GO.db 3.0.0, GenomeInfoDb 1.2.0, IRanges 2.0.0, KEGGgraph 1.24.0, RSQLite 0.11.4, S4Vectors 0.4.0, XML 3.98-1.1, clusterProfiler 2.0.0, graph 1.44.0, knitr 1.7, org.Hs.eg.db 3.0.0, pathview 1.6.0
- Loaded via a namespace (and not attached): Biostrings 2.34.0, DO.db 2.8.0, GOSemSim 1.24.0, KEGG.db 3.0.0, KEGGREST 1.6.0, MASS 7.3-35, Rcpp 0.11.3, Rgraphviz 2.10.0, XVector 0.6.0, codetools 0.2-9, colorspace 1.2-4, digest 0.6.4, evaluate 0.5.5, formatR 1.0, ggplot2 1.0.0, grid 3.1.1, gtable 0.1.2, highr 0.3, httr 0.5, igraph 0.7.1, labeling 0.3, munsell 0.4.2, plyr 1.8.1, png 0.1-7, proto 0.3-10, qvalue 1.40.0, reshape2 1.4, scales 0.2.4, stringr 0.6.2, tools 3.1.1, zlibbioc 1.12.0

References

[1] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. Gosemsim: an r package for measuring semantic similarity among go terms and gene products. *Bioinformatics*, 26(7):976–978, 2010. PMID: 20179076.

[2] Guangchuang Yu, Le-Gen Wang, Yanyan Han, and Qing-Yu He. clusterprofiler: an r package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*, 16(5):284–287, May 2012.

[3] Elizabeth I Boyle, Shuai Weng, Jeremy Gollub, Heng Jin, David Botstein, J Michael Cherry, and Gavin Sherlock. GO::TermFinder–open source software for accessing gene ontology information and finding significantly enriched gene ontology terms associated with a list of genes. *Bioinformatics (Oxford, England)*, 20(18):3710–3715, December 2004. PMID: 15297299.

[4] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, October 2005.

[5] Weijun Luo and Cory Brouwer. Pathview: an R/Bioconductor package for pathway-based data integration and visualization. 29:1830–1831. PMID: 23740750.

[6] Marcus Schmidt, Daniel B?hm, Christian von T?rne, Eric Steiner, Alexander Puhl, Henryk Pilch, Hans-Anton Lehr, Jan G. Hengstler, Heinz K?lbl, and Mathias Gehrmann. The humoral immune system has a key prognostic impact in node-negative breast cancer. *Cancer Research*, 68(13):5405 –5413, July 2008.

[7] Guangchuang Yu and Qing-Yu He. Functional similarity analysis of human virus-encoded miRNAs. *Journal of Clinical Bioinformatics*, 1(1):15, May 2011.

[8] Guangchuang Yu, Chuan-Le Xiao, Xiaochen Bo, Chun-Hua Lu, Yide Qin, Sheng Zhan, and Qing-Yu He. A new method for measuring functional similarity of microRNAs. *Journal of Integrated OMICS*, 1(1):49–54, February 2011.