Analysis of bulk RNA-seq data - Part II: From counts to DGE

Analysis of Next-Generation Sequencing Data

Friederike Dündar

Applied Bioinformatics Core

Slides at https://bit.ly/2CUdS9z

February 26, 2019

1http://physiology.med.cornell.edu/faculty/skrabanek/lab/angsd/schedule_2018/
1 Normalization of read counts

2 Exploratory analyses

3 Differential gene expression

4 Downstream analyses

5 References
Many slides today were influenced or taken from the excellent book *Data Analysis for the Life Sciences* by Rafael Irizarry and Michael Love, and training material developed by the Harvard Chan Bioinformatics Core.

Go and check them out for even more details! The Harvard Chan Bioinformatics Core’s material can be found at their github page: https://github.com/hbcttraining/DGE_workshop
General bioinformatics workflow for RNA-seq data

1. **Raw reads** (.fastq)
2. **Aligned reads** (.sam/.bam)
3. **Mapping** (STAR)
4. **Counting** (featureCounts)
5. **Normalized read count table** (.txt)
6. **Normalized read count table** (.Robj)
7. **Estimated transcript abundances**
8. **Summarizing to gene levels & Normalizing** (DESeq2, edgeR)
9. **List of fold changes & statistical values** (.Robj, .txt)

**DE test**
- DESeq2, edgeR, limma

**Pseudoalignment**
- kallisto, salmon
Normalization of read counts
Read counts are influenced by numerous factors, not just expression strength

Raw counts\(^2\): number of reads (or fragments) overlapping with the union of exons of a gene.

Raw count numbers are not just a reflection of the actual number of captured transcripts!

They are strongly influenced by:
- sequencing depth
- gene length
- DNA sequence content (\% GC)
- expression of all other genes in the same sample

\(^2\)also true for "estimated" gene counts from pseudoaligners
Normalization of read counts

Influences on read count numbers

1. Sequencing depth

sequencing depth of Sample A $\gg$ Sample B

Sample A Reads

Sample B Reads
Normalisation of read counts

Influences on read count numbers

2. Gene length (and GC bias)
Normalization of read counts

Influences on read count numbers

3. RNA composition - individual gene abundances

very highly expressed transcript soaks up significant portion of the reads reducing the range of read counts available for other transcripts

in the absence of that highly expressed transcript, the remaining transcripts’ expression differences become more clear
Influences on read count numbers - summary

- gene length
- transcript sequence (% GC)

need to be corrected when comparing different genes

- sequencing depth
- expression of all other genes within the same sample

need to be corrected when comparing the same gene between different samples

Which biases are relevant for comparing different samples?
Normalization of read counts

Different units for expression values

- **Raw counts**: number of reads/fragments overlapping with the union of exons of a gene

- **[RF]PKM**: Reads/Fragments per Kilobase of gene per Million reads mapped – AVOID!

- **TPM**: Transcripts Per Million

- **rlog**: log2-transformed count data normalized for small counts and library size (DESeq2)

\[ X_i \]

\[ RPKM_i = \frac{X_i}{l_i} \left( \frac{10^3}{N} \right) \]

\[ TPM_i = \frac{X_i}{l_i} \left( \frac{1}{\sum_j \frac{X_j}{l_j}} \right) \times 10^6 \]
Why not RPKMs?

- [RF]PKM values are not comparable between samples – Do NOT use them!
- If you need normalized expression values for exploratory plots, use TPM or DESeq2’s rlog values
Normalizing of read counts

Working with read counts

- Download the featureCounts results to your laptop.
- Read the featureCounts results into R.
- Let’s normalize!
Exploratory analyses
Exploratory analyses do not test a null hypothesis! They are meant to familiarize yourself with the data to discover biases and unexpected variability!

Typical exploratory analyses:
- **correlation** of gene expression between different samples
- (hierarchical) **clustering**
- **dimensionality reduction** methods, e.g. PCA
- dot plots/box plots/violin plots of individual genes

Use **normalized and transformed** read counts for data exploration!
Pairwise correlation of gene expression values

- Replicates of the same condition should show high correlations (>0.9)
  
- **Pearson** method: *metric* differences between samples
  - influenced by outliers
  - covariance of two variables divided by the product of their standard deviation
  - suitable for normally distributed values

- **Spearman** method: based on *rankings*
  - less sensitive
  - less driven by outliers

- R function: `cor()`
Exploratory analyses

Hierarchical clustering – grouping similar samples

**Goal:** partition the objects into homogeneous groups, such that the within-group similarities are large.

- **Result:** *dendrogram*
  - clustering is obtained by **cutting the dendrogram** at the desired level

- **Similarity measure**
  - Euclidean
  - Pearson

- **Distance measure**
  - Complete: largest distance
  - Average: average distance

single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples
Exploratory analyses

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level

- Similarity measure
  - Euclidean
  - Pearson

- Distance measure
  - Complete: largest distance
  - Average: average distance

single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples
Exploratory analyses

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level

- Similarity measure
  - Euclidean
  - Pearson

- Distance measure
  - Complete: largest distance
  - Average: average distance

Single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples.
Hierarchical clustering – grouping similar samples

**Goal:** partition the objects into homogeneous groups, such that the within-group similarities are large.

- **Result:** *dendrogram*
  - clustering is obtained by **cutting the dendrogram** at the desired level

- **Similarity measure**
  - Euclidean
  - Pearson

- **Distance measure**
  - Complete: largest distance
  - Average: average distance

Single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples.
Hierarchical clustering – grouping similar samples

**Goal:** partition the objects into homogeneous groups, such that the within-group similarities are large.

- **Result:** dendrogram
  - clustering is obtained by cutting the dendrogram at the desired level

- **Similarity measure**
  - Euclidean
  - Pearson

- **Distance measure**
  - Complete: largest distance
  - Average: average distance

single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples
Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

- Result: dendrogram
  - clustering is obtained by cutting the dendrogram at the desired level

- Similarity measure
  - Euclidean
  - Pearson

- Distance measure
  - Complete: largest distance
  - Average: average distance

Single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples.
Hierarchical clustering - R code

```r
## calculate the correlation between columns of a matrix
pw_cor <- cor(rlog.norm.counts, method = "pearson"

## use the correlation as a distance measure
distance.m_rlog <- as.dist(1 - pw_cor)

## plot() can directly interpret the output of hclust() to generate
## a dendrogram
plot(hclust(distance.m_rlog),
     labels = colnames(rlog.norm.counts),
     main = "rlog transformed read counts")
```

---

**Exploratory analyses**

F. Dündar (ABC, WCM)

Analysis of bulk RNA-seq data - Part II: From...
**Exploratory analyses**

Principal component analysis – capturing variability

**Goal:** reduce the dataset to have fewer dimensions, yet approx. preserve the distance between samples

starting point: matrix with expression values per gene and sample, e.g. 6,600 genes x 10 samples

<table>
<thead>
<tr>
<th></th>
<th>SNF2_1</th>
<th>SNF2_2</th>
<th>SNF2_3</th>
<th>SNF2_4</th>
<th>SNF2_5</th>
<th>WT_1</th>
<th>WT_2</th>
<th>WT_3</th>
<th>WT_4</th>
<th>WT_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>YDL248W</td>
<td>109</td>
<td>84</td>
<td>100</td>
<td>112</td>
<td>62</td>
<td>47</td>
<td>65</td>
<td>60</td>
<td>95</td>
<td>43</td>
</tr>
<tr>
<td>YDL247W</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>YDL247W.A</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>YDL246C</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>YDL245C</td>
<td>1</td>
<td>6</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>YDL244W</td>
<td>79</td>
<td>59</td>
<td>49</td>
<td>60</td>
<td>37</td>
<td>9</td>
<td>8</td>
<td>12</td>
<td>30</td>
<td>14</td>
</tr>
</tbody>
</table>

assay(DESeq.rlog)[topVarGenes,])

%>% t %>% prcomp

transformed into 6,600 **principal components** x 10 samples

- linear combi of optimally weighted observed variables
- the vectors along which the variation between samples is maximal
- PC1-3 are usually sufficient to capture the major trends!
Exploratory analyses

PCA vs. hierarchical clustering

- Often similar results because both techniques should capture the most dominant patterns.
- PCA will always be run on just a subset of the data!
- Clustering will ALWAYS return clusters, PCA may not if the patterns of variation are too random.

See `practical_exploratory.Rmd` R code to generate exploratory plots. Use the `pcaExplorer` package!

See the chapter “Distance and Dimension Reduction” in Irizarry and Love [2015] for more details and the StatQuest video(s) on youtube.
Differential gene expression
Differential gene expression

Understand your null hypothesis!

- **DGE**: Differential Gene Expression
  - Has the total output of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by DESeq2/edgeR/limma
  - see Soneson et al. [2015] and bioconductor’s tximport package vignette for details

- **DTU**: Differential Transcript Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by DEXSeq (!)
  - see Love et al. [2018] for details
**Understand your null hypothesis!**

- **DGE**: Differential Gene Expression
  - Has the total output of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by DESeq2/edgeR/limma
  - see Soneson et al. [2015] and bioconductor’s tximport package vignette for details

- **DTU**: Differential Transcript Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by DEXSeq (!)
  - see Love et al. [2018] for details
Differential gene expression

DGE basics

H₀: There is no difference in the read distributions of the 2 conditions.

1. Estimate magnitude of DE taking into account differences in sequencing depth, technical, and biological read count variability.

2. Estimate the significance of the difference accounting for performing thousands of tests.

1 test per gene!
Applying linear models for read count modeling

Normalized expression values of snf2 (YOR290C)
Applying linear models for read count modeling

Normalized expression values of snf2 (YOR290C)

Graph showing expression values for snf2 (YOR290C) under different genotypes (SNF2 and WT). The arrows indicate the differences in expression levels, with higher values for SNF2 (9.78) compared to WT (6.67).
To describe all expression values of one (!) example gene (snf2), we can use a linear model like this:

\[
Y = b_0 + b_1 \times x + e
\]

Linear models model a response variable as a linear combination of predictors (betas), plus randomly distributed noise (e).
Applying linear models for read count modeling

To describe all expression values of one (!) example gene (snf2), we can use a linear model like this:

$$ Y = b_0 + b_1 \times x + e $$

- $b_0$: intercept, i.e. average value of the baseline group
- $b_1$: difference between baseline and non-reference group
- $x$: 0 if genotype == “SNF2”, 1 if genotype == “WT”

Linear models model a response variable as a linear combination of predictors (betas), plus randomly distributed noise ($e$).
Model formulae syntax in R

- regression functions in R (e.g., `lm()`), `glm()` use a "model formula" interface
- the basic format is: `response variable ~ explanatory variables`\(^3\), e.g. `lm( y ~ x )`

If you find yourself using linear models and somewhat complicated experimental designs more often than not, we strongly recommend to work through **chapters 4 and 5** of the PH525x series **Biomedical Data Science** [Irizarry and Love, 2016]

---

\(^3\)Tilde means "is modeled by" or "is modeled as a function of". See King [2016] for more details on the specialy meaning of mathematical operators within formula contexts.
Applying linear models for read count modeling

Describe expression values *snf2* using a linear model:

\[
Y = b_0 + b_1 \times x + e
\]

Factor of interest \((b_1)\) can be estimated as follows:

1. FIT the model
   ```r
   lmfit <- lm(rlog.norm ~ genotype)
   ```
2. ESTIMATE the coefficients
   ```r
   coef(lmfit)
   ```

<table>
<thead>
<tr>
<th>(Intercept)</th>
<th>genotypeWT</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.666</td>
<td>3.111</td>
</tr>
</tbody>
</table>

Both values \((b0, b1)\) are estimates!
(They're spot-on because the values are so clear and the model is so simple!)

- **\(b_0\)**: intercept, i.e. average value of the baseline group
- **\(b_1\)**: difference between baseline and non-reference group
- **\(x\)**: 0 if genotype == "SNF2", 1 if genotype == "WT"
H₀: There is no difference in the read distributions of the 2 conditions.

1. Estimate **magnitude** of DE taking into account differences in sequencing depth, technical, and biological read count variability.

2. Estimate the **significance** of the difference accounting for performing thousands of tests.

1 test per gene!
1 **Fitting** a sophisticated regression model to the read counts (per gene!)
   - library size factor
   - dispersion estimate using information across multiple genes
   - negative binomial distribution of read counts is assumed

\[ K_{ij} \sim NB(\mu_{ij}, \alpha_i) \]

2 Estimating coefficients to obtain the difference (**log2FC**)

3 **Test** whether the log2FC is “far away” from zero (remember H0!)
DGE steps (formulae from DESeq2)

1. **Fitting** a more sophisticated model to the read counts (per gene!)
   - library size factor
   - dispersion estimate using information across multiple genes
   - negative binomial distribution of read counts is assumed

\[
K_{ij} \sim \text{NB}(\mu_{ij}, \alpha_i)
\]

read counts for gene \(i\) and sample \(j\)

2. Estimating coefficients to obtain the difference (\(\log_{2}FC\))

3. **Test** whether the \(\log_{2}FC\) is “far away” from zero (remember H0!)
Differential gene expression

DGE steps (formulae from DESeq2)

1. **Fitting** a more sophisticated model to the read counts (per gene!)
   - library size factor
   - dispersion estimate using information across multiple genes
   - negative binomial distribution of read counts is assumed

\[
K_{ij} \sim \text{NB}(\mu_{ij}, \alpha_i)
\]

- read counts for gene \(i\) and sample \(j\)
- mean expr. (fitted towards the average dispersion)
- library size factor
- gene-specific dispersion parameter

2. Estimating coefficients to obtain the difference (log2FC)

3. **Test** whether the log2FC is “far away” from zero (remember H0!)
DGE steps (formulae from DESeq2)

1. **Fitting** a more sophisticated model to the read counts (per gene!)
   - library size factor
   - dispersion estimate using information across multiple genes
   - negative binomial distribution of read counts is assumed
   \[ K_{ij} \sim \text{NB}(\mu_{ij}, \alpha_i) \]

2. Estimating coefficients to obtain the difference (**log2FC**)
   - define the **contrast of interest**, e.g. ~ condition or ~ batchEffect + condition
   - always put the factor of interest last
   - order of the factor levels determines the direction of fold change that is reported

3. **Test** whether the log2FC is “far away” from zero (remember H0!)
Summary: read counts to DGE and other analyses

matrix of read counts

- size factors
- rlog

QC
- exploratory analyses
- some downstream analyses

DESeq()
- size factors
- gene-wise dispersion estimation
- shrinking gene-wise dispersion estimates
- gene-wise GLM fit
- coefficient estimation
- DE test
- genes with logFC and stat values

results()
Comparison of additional tools for DGE analysis

When in doubt, compare the results of limma, edgeR, and DESeq2 to get a feeling for how robust your favorite DE genes are.

<table>
<thead>
<tr>
<th>Feature</th>
<th>DESeq2</th>
<th>edgeR</th>
<th>limmaVoom</th>
<th>Cuffdiff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq. depth normalization</td>
<td>Sample-wise size factor</td>
<td>Gene-wise trimmed median of means (TMM)</td>
<td>Gene-wise trimmed median of means (TMM)</td>
<td>FPKM-like or DESeq-like</td>
</tr>
<tr>
<td>Assumed distribution</td>
<td>Neg. binomial</td>
<td>Neg. binomial</td>
<td>log-normal</td>
<td>Neg. binomial</td>
</tr>
<tr>
<td>Test for DE</td>
<td>Exact test (Wald)</td>
<td>Exact test for over-dispersed data</td>
<td>Generalized linear model</td>
<td>t-test</td>
</tr>
<tr>
<td>False positives</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Detection of differential isoforms</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Support for multi-factored experiments</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Runtime (3-5 replicates)</td>
<td>Seconds to minutes</td>
<td>Seconds to minutes</td>
<td>Seconds to minutes</td>
<td>Hours</td>
</tr>
</tbody>
</table>

Table 5: Comparison of programs for differential gene expression identification. Based on (Rapaport et al., 2013; Seyednasrollah et al., 2013; Schurch et al., 2015).
Downstream analyses
Understanding the RESULTS of the DGE analysis

- Investigate the results() output:
  - How many DE genes? (FDR/q-value!)
  - How strongly do the DE genes change?
  - Directions of change?
  - Are your favorite genes among the DE genes?

F. Dündar (ABC, WCM)
Understanding the FUNCTIONS of your DE genes

There are myriad tools for this – many are web-based, many are R packages, many will address very specific questions. Typical points of interest are:

- enriched gene ontology (GO) terms
  - ontology = standardized vocabulary
  - 3 classes of gene ontologies are maintained:
    - biological processes (BP), cell components (CC), and molecular functions (MF)
- enriched pathways
  - gene sets: e.g. from MSigDB [Liberzon et al., 2015]
  - physical interaction networks: e.g. from STRING [Szklarczyk et al., 2017]
  - metabolic (and other) pathways: e.g. from KEGG [Kanehisa et al., 2017]
- upstream regulators

None (!) of these methods should lead you to make definitive claims about the role of certain pathways for your phenotype. These are *hypothesis-generating* tools!
Two typical approaches of enrichment analyses

1. Over-representation analysis (ORA)

All known genes in a species (categorized into groups)

<table>
<thead>
<tr>
<th>Category</th>
<th>Background</th>
<th>DE list</th>
<th>Over-represented?</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>35/6600</td>
<td>25/500</td>
<td>likely</td>
</tr>
<tr>
<td>B</td>
<td>56/6600</td>
<td>2/500</td>
<td>unlikely</td>
</tr>
<tr>
<td>C</td>
<td>10/6600</td>
<td>9/500</td>
<td>likely</td>
</tr>
</tbody>
</table>
Two typical approaches of enrichment analyses

1. Over-representation analysis (ORA)
   - “2x2 table method”
   - assessing overlap of DE genes with genes of a given pathway
   - statistical test: e.g. hypergeometric test
   - limitations:
     ▶ direction of change is ignored
     ▶ magnitude of change is ignored
     ▶ interprets genes as well as pathways as independent entities

See Khatri et al. [2012] for details!
# Two typical approaches of enrichment analyses

## 1. Over-representation analysis (ORA)

<table>
<thead>
<tr>
<th>Name</th>
<th>Scope of Analysis</th>
<th>P-value</th>
<th>Correction for Multiple Hypotheses</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Onto-Express</td>
<td>GO</td>
<td>Hypergeometric, binomial, chi-square</td>
<td>FDR, Bonferroni, Sidak, Holm</td>
<td>Web</td>
</tr>
<tr>
<td>GenMAPP/MAPPFinder</td>
<td>GO, KEGG, MAPP</td>
<td>Percentage/z-score</td>
<td>None</td>
<td>Standalone</td>
</tr>
<tr>
<td>(High throughput) GoMiner</td>
<td>GO</td>
<td>Relative enrichment, Hypergeometric</td>
<td>None</td>
<td>Standalone, Web</td>
</tr>
<tr>
<td>FatiGO</td>
<td>GO, KEGG</td>
<td>Hypergeometric</td>
<td>None</td>
<td>Web</td>
</tr>
<tr>
<td>GOstat</td>
<td>GO</td>
<td>Chi-square</td>
<td>FDR</td>
<td>Web</td>
</tr>
<tr>
<td>GOTree Machine</td>
<td>GO</td>
<td>Hypergeometric</td>
<td>Bootstrap</td>
<td>Web</td>
</tr>
<tr>
<td>FuncAssociate</td>
<td>GO</td>
<td>Hypergeometric</td>
<td>Bonferroni, Holm, FDR, Hommel, Hochberg</td>
<td>Web</td>
</tr>
<tr>
<td>GOToolBox</td>
<td>GO</td>
<td>Hypergeometric</td>
<td>Bonferroni</td>
<td>Web</td>
</tr>
<tr>
<td>GeneMerge</td>
<td>GO</td>
<td>Hypergeometric</td>
<td>Benjamini-Yekutieli</td>
<td>Web</td>
</tr>
<tr>
<td>GOEAST</td>
<td>GO, KEGG, BioCarta, User defined</td>
<td>Hypergeometric, Chi-square</td>
<td>Bonferroni step-down, Benjamini-Hochberg</td>
<td>Standalone</td>
</tr>
</tbody>
</table>
Two typical approaches of enrichment analyses

2. Functional Class Scoring ("Gene set enrichment")
   - gene-level statistics for all genes in a pathway are aggregated into a single pathway-level statistic
   - score will depend on size of the pathway, and the amount of correlation between genes in the pathway
   - all genes are used
   - direction and magnitude of change matter
   - coordinated changes of genes within the same pathway matter, too
Two typical approaches of enrichment analyses

2. Functional Class Scoring ("Gene set enrichment")

Table S2. FCS pathway analysis tools.
Khatri et al. (2012). doi: 0.1371/journal.pcbi.1002375

<table>
<thead>
<tr>
<th>Name</th>
<th>Scope of Analysis</th>
<th>Gene-level Statistics</th>
<th>Gene Statistic</th>
<th>Set P-value</th>
<th>Correction for Multiple Hypotheses</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSEA</td>
<td>GO, KEGG, BioCarta, MAPP, transcription factors, microRNA, cancer molecules</td>
<td>Signal-to-noise ratio, t-test, cosine, euclidian and manhattan distance, Pearson correlation, (log2) fold-change, log difference</td>
<td>Kolmogorov-Smirnov</td>
<td>Phenotype permutation, Gene set permutation</td>
<td>FDR</td>
<td>Standalone, R package</td>
</tr>
<tr>
<td>sigPathway</td>
<td>GO, KEGG, BioCarta, humans</td>
<td>t-statistic</td>
<td>Wilcoxon rank sum</td>
<td>Phenotype permutation, Gene set permutation</td>
<td>FDR (NPMLE)</td>
<td>R package</td>
</tr>
<tr>
<td>Category</td>
<td>GO, KEGG</td>
<td>t-statistic</td>
<td>Wilcoxon rank sum</td>
<td>Phenotype permutation</td>
<td>NA</td>
<td>R package</td>
</tr>
<tr>
<td>SAFE</td>
<td>GO, KEGG, PFAM</td>
<td>Student’s t-test, Welch’s t-test, SAM t-test, f-statistic, Cox proportional hazards model, linear regression</td>
<td>Wilcoxon rank sum, Fisher’s exact test statistic, Pearson’s test, t-test of average difference simple and multinomial logistic regression, Q-statistics mean</td>
<td>Phenotype permutation</td>
<td>FWER (Bonferroni, Holm’s step-up), FDR (Benjamini-Hochberg, Yekutieli-Benjamini)</td>
<td>R package</td>
</tr>
<tr>
<td>GlobalTest</td>
<td>GO, KEGG</td>
<td>NA</td>
<td>Phenotype permutation, asymptotic distribution, Gamma distribution</td>
<td>NA</td>
<td>R package</td>
<td></td>
</tr>
<tr>
<td>PCOT2</td>
<td>User specified</td>
<td>Hotelling’s $T^2$</td>
<td>sum of squared $d$-statistic</td>
<td>Phenotype permutation</td>
<td>FDR (Benjamini-Hochberg, Yekutieli-Benjamini), FWER (Bonferroni, Holm, Hochberg, Hommel)</td>
<td>R package</td>
</tr>
<tr>
<td>SAM-GS</td>
<td>User specified</td>
<td>$d$-statistic</td>
<td>Phenotype permutation</td>
<td>Excel plug-in</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Two typical approaches of enrichment analyses

2. Functional Class Scoring: Example GSEA

http://slideplayer.biz.tr/slide/2738467/10/images/20/Gene+Set+Enrichment+Analysis+(GSEA).jpg
Two typical approaches of enrichment analyses

2. Functional Class Scoring ("Gene set enrichment")

Example GSEA results for positive and negative correlation

Summary – downstream analyses

Know your biological question(s) of interest!

- All enrichment methods potentially suffer from gene length bias
  - Long genes will get more reads [Young et al., 2010]

- For GO terms:
  - Use goseq to identify enriched GO terms
  - Use additional tools, such as GOrilla, REVIGO [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists

- For KEGG pathways:
  - E.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017]

- Miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietary software!)

---

See the additional links and material on our course website!

References
[Anders and Huber, 2010, D’haeseleer, 2005, Dillies et al., 2013, Doroszuk et al., 2012, Subramanian et al., 2005, Dündar et al., 2018]


