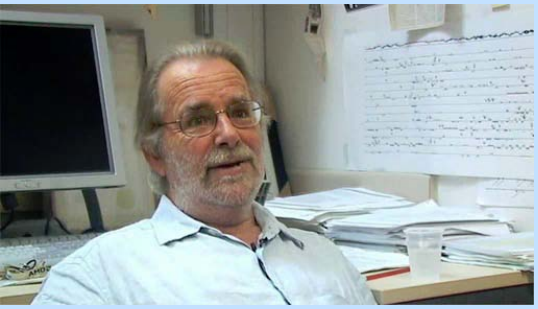
Enrichment Analysis

Using GO for Expression Analysis



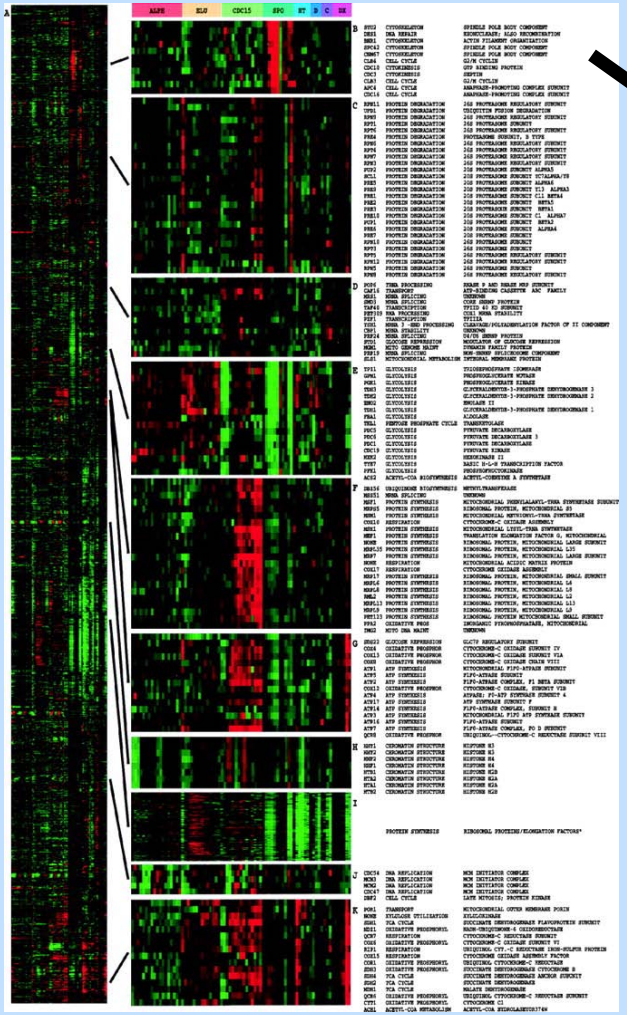
Jane Lomax, EBI
Jennifer Deegan, EBI

Ashburner et al. Gene Ontology: tool for the unification of biology. *Nature Genetics 00*

Jane Lomax (EBI) www.geneontology.org/teaching_resources/presentations/2006-02_MUGEN_expression-analysis_jlomax.ppt

The Gene Ontology (GO)

- A set of biological phrases (terms) which are applied to genes, e.g.:
  - protein kinase
  - apoptosis
  - Membrane
- Genes are linked, or associated, with GO terms by trained curators at genome databases
  - known as gene associations or GO annotations
- Some GO annotations are created automatically
- Allows biologists to make inferences across many genes without researching each one individually
- As usual, we say genes but mean gene products (proteins)

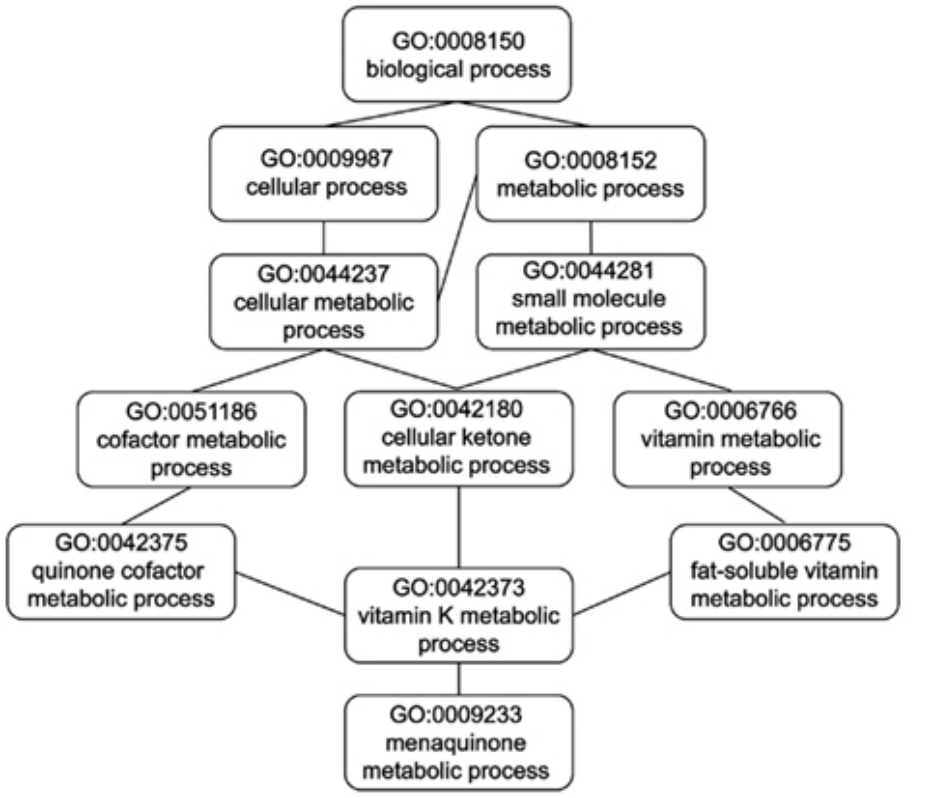


ASPARAGINE UTILIZATION
CRYSTAL VIOLET RESISTANC
LYSINE BIOSYNTHESIS
CELL WALL CATABOLISM
OXIDATIVE STRESS RESPONS
GLUCOSE REPRESSION
AGING
RIBOSE METABOLISM
PROTEIN FOLDING
ANTIPROLIFERATIVE PROTEI
RNA PROCESSING
UBIQUINONE BIOSYNTHESIS
TRANSCRIPTION
1 PROTEIN SYNTHESIS

Eisen, Michael B. et al. (1998) Proc. Natl. Acad. Sci. USA 95, 14863-14868

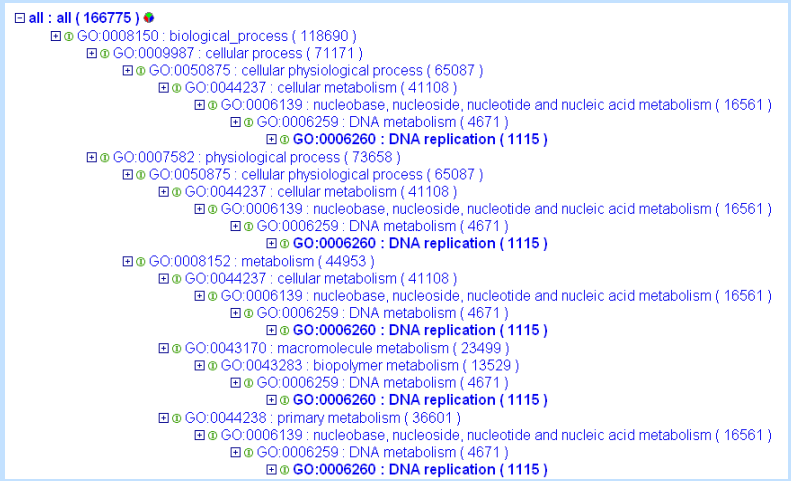
GO structure

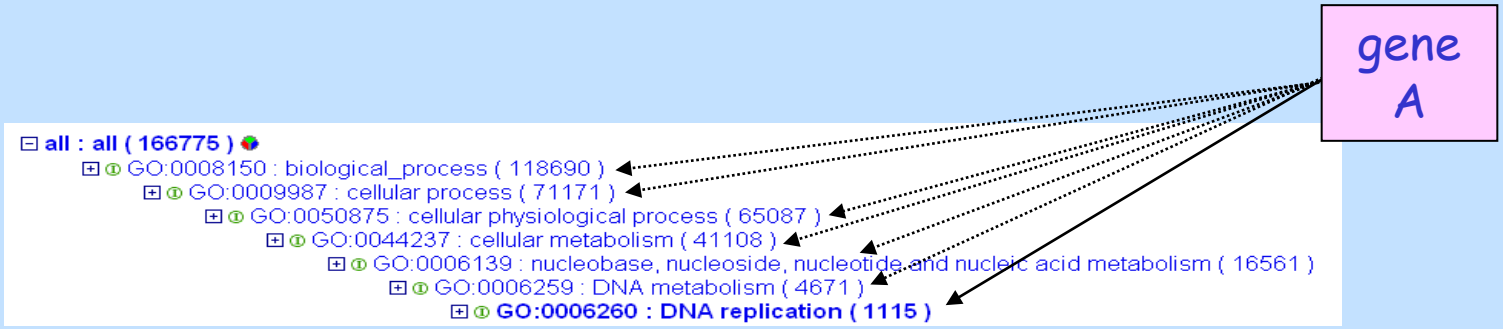
- GO terms are related within a DAG hierarchy
- Edge types: ⓘ *is a* (*is a subtype of*); Ⓟ *part of*; (also: has *part*, *regulates*, *negatively regulates*, *positively regulates*…)



www.frontiersin.org/articles/10.3389/fgene.2012.00115

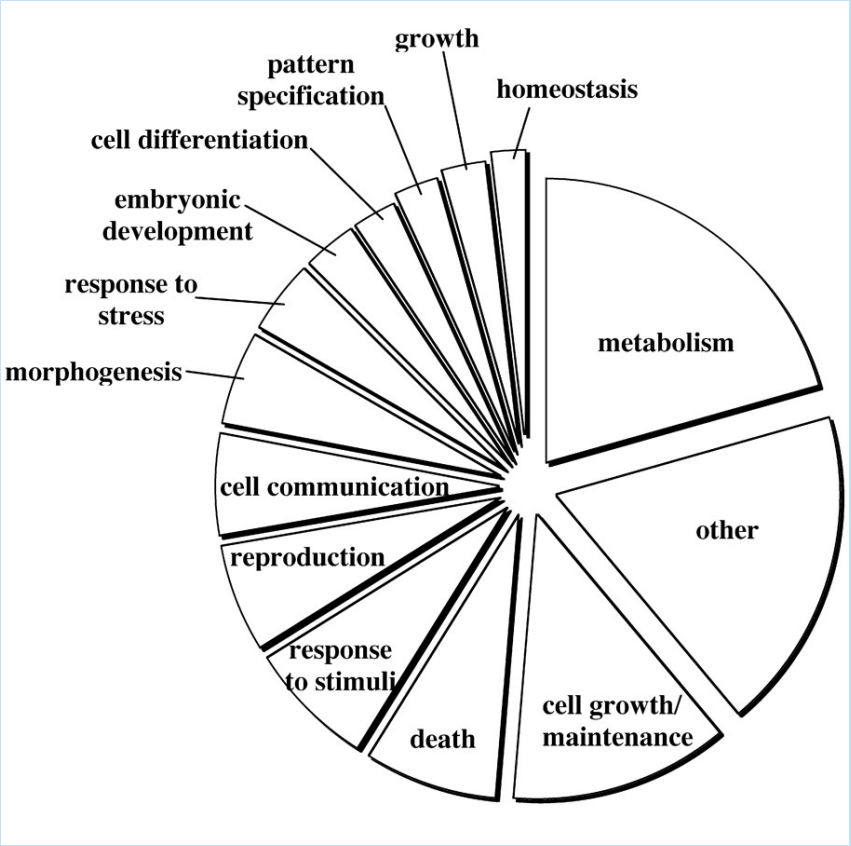
GO structure

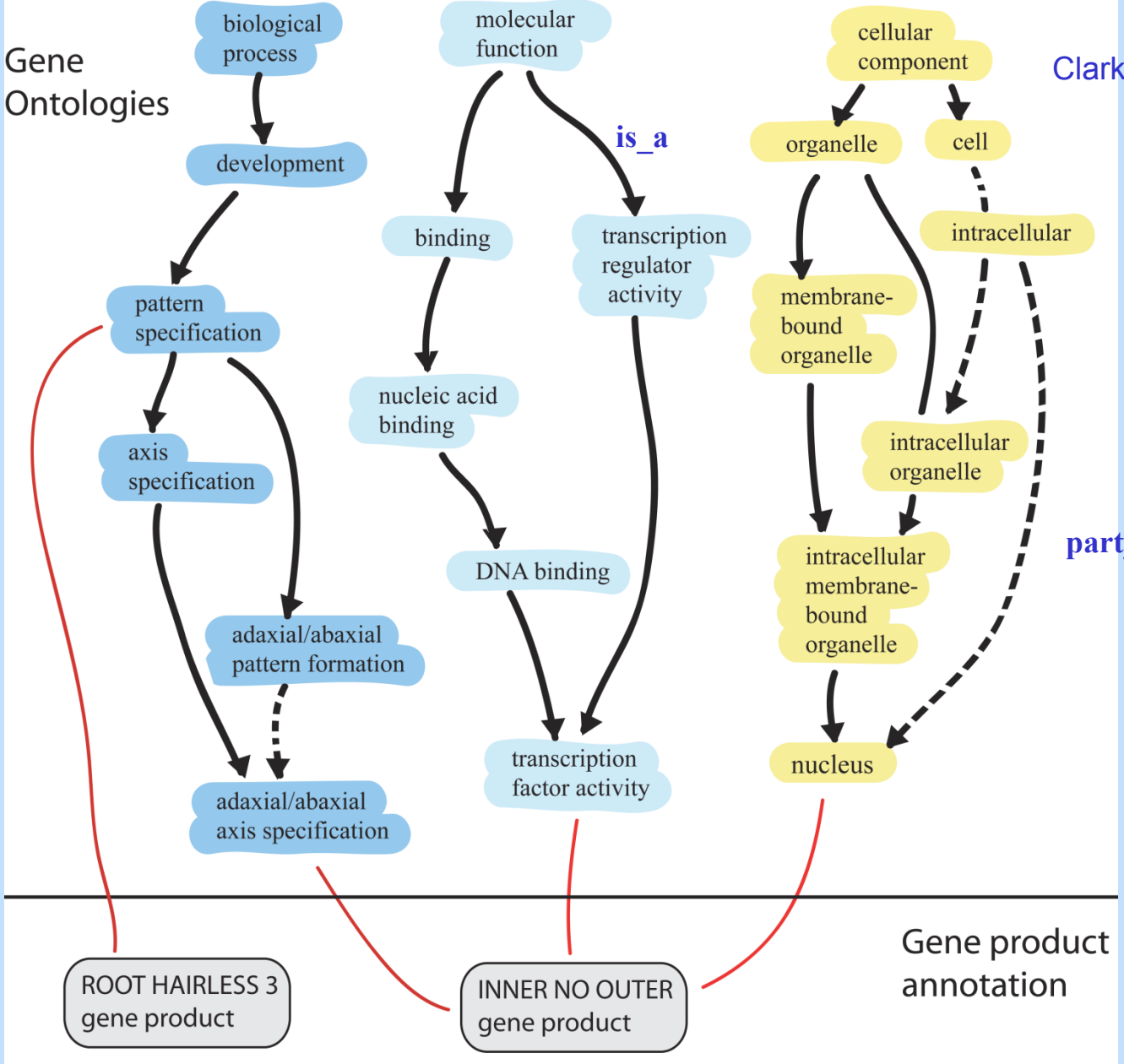




GO structure

- Genes can be grouped according to user-defined levels
- Allows broad overview of gene set or genome




Gene Ontologies
biological process
development
pattern specification
axis specification
adaxial/abaxial pattern formation
adaxial/abaxial axis specification
molecular function
is_a
binding
transcription regulator activity
nucleic acid binding
DNA binding
transcription factor activity
cellular component
organelle
cell
intracellular
membrane-bound organelle
intracellular organelle
intracellular membrane-bound organelle
nucleus
part_of
Clark et al., 2005
ROOT HAIRLESS 3 gene product
INNER NO OUTER gene product
Gene product annotation

How does GO work?

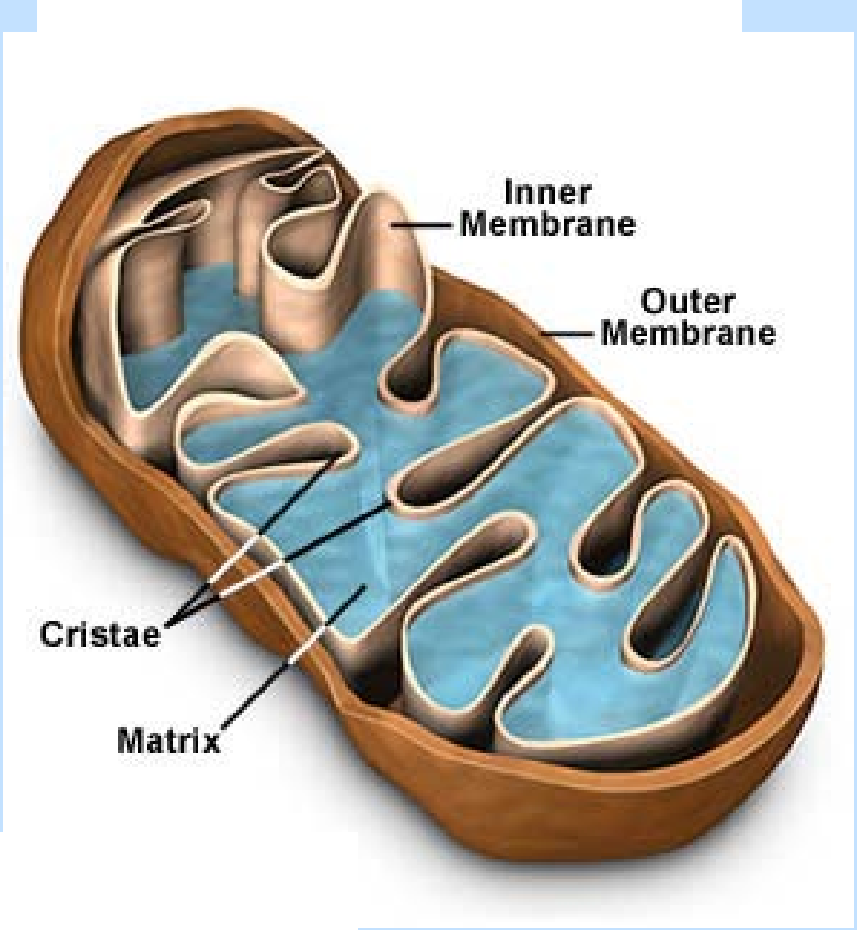
What information might we want to capture about a gene product?

- What does the gene product do?
- Where and when does it act?
- Why does it perform these activities?

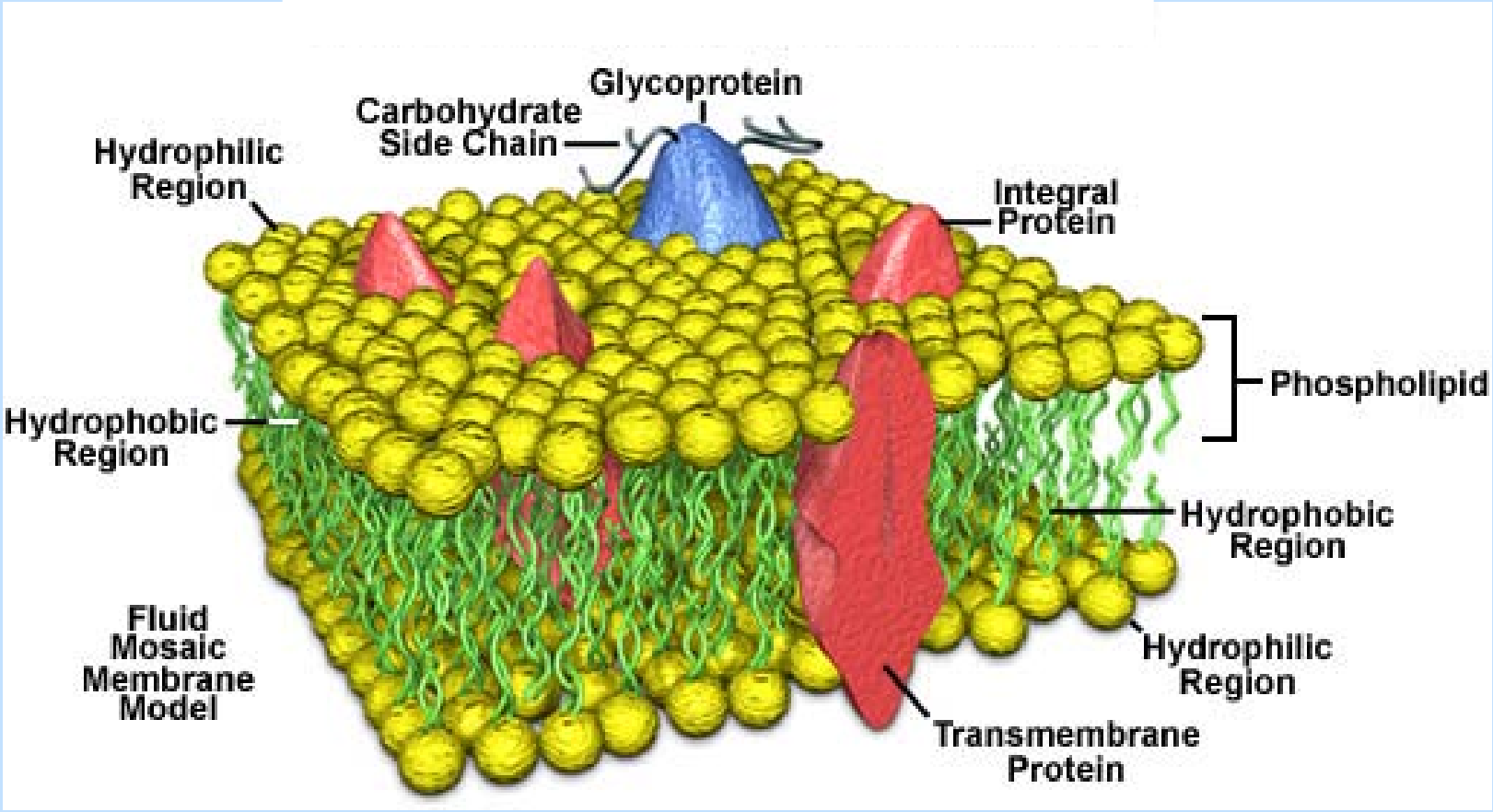
- GO terms are divided into three parts:
  - cellular component
  - molecular function
  - biological process

Cellular Component

- where a gene product acts

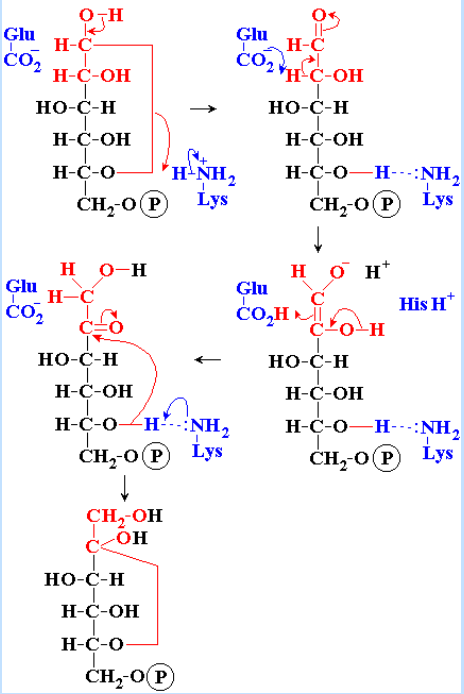


Cellular Component



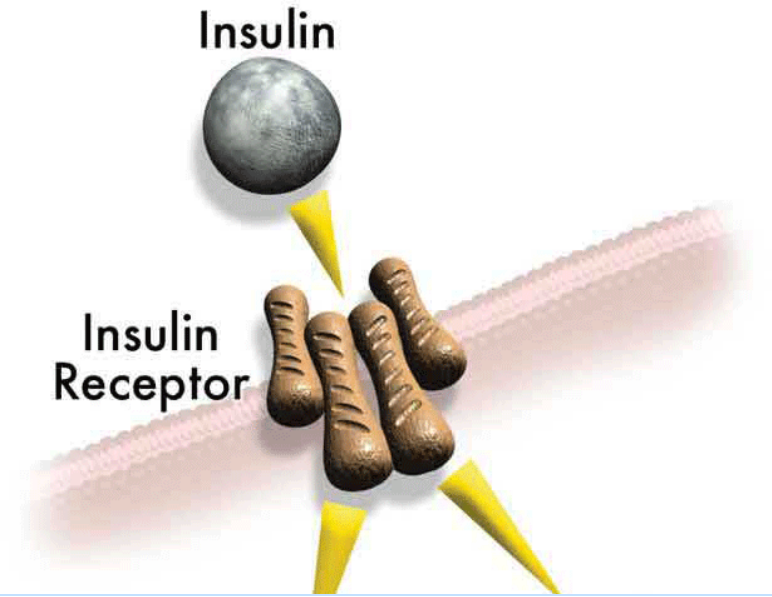
Molecular Function

- activities or "jobs" of a gene product



glucose-6-phosphate

isomerase activity



insulin binding

insulin receptor

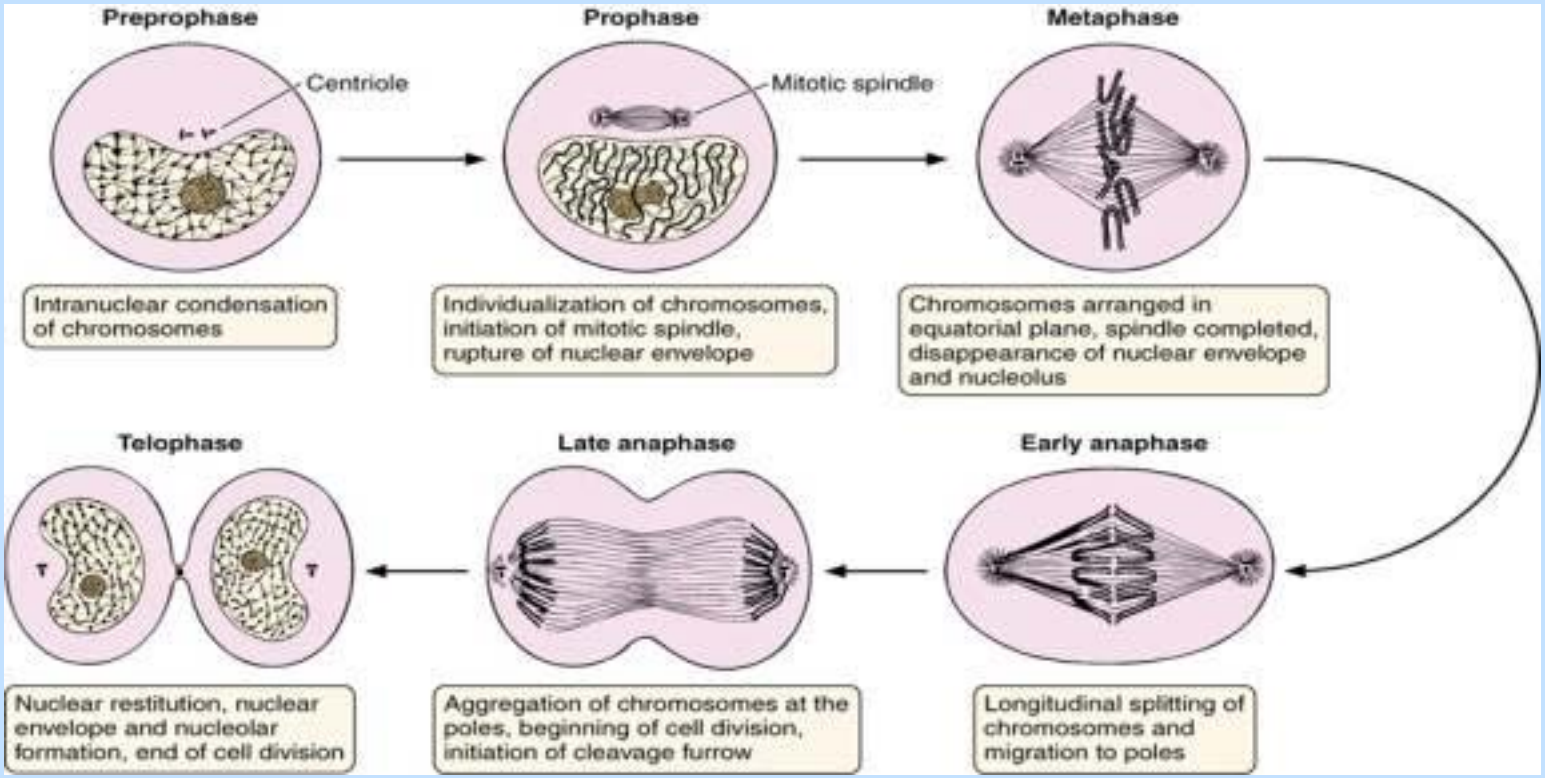
activity

Molecular Function

- A gene product may have several functions; a function term refers to a single reaction or activity, not a gene product.
- Sets of functions make up a biological process.

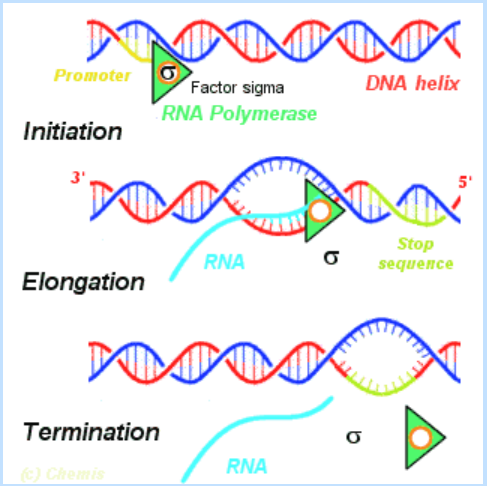
Biological Process

a commonly recognized series of events

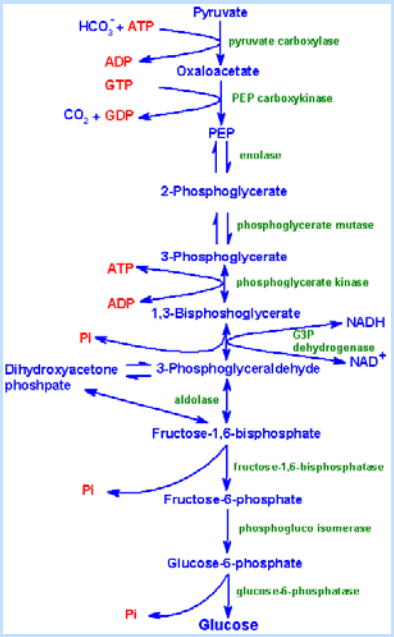


cell division

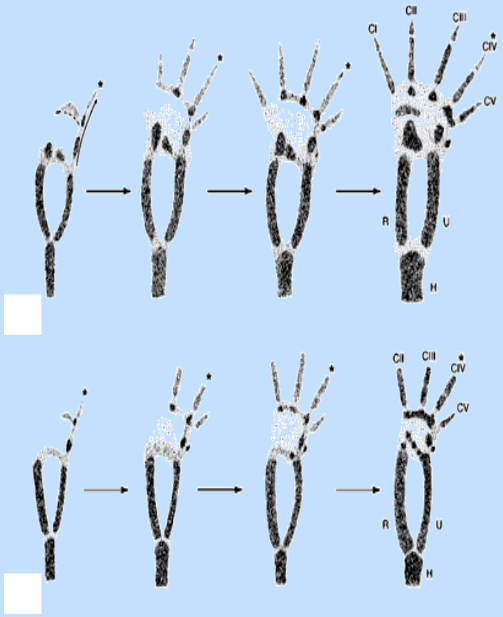
Biological Process



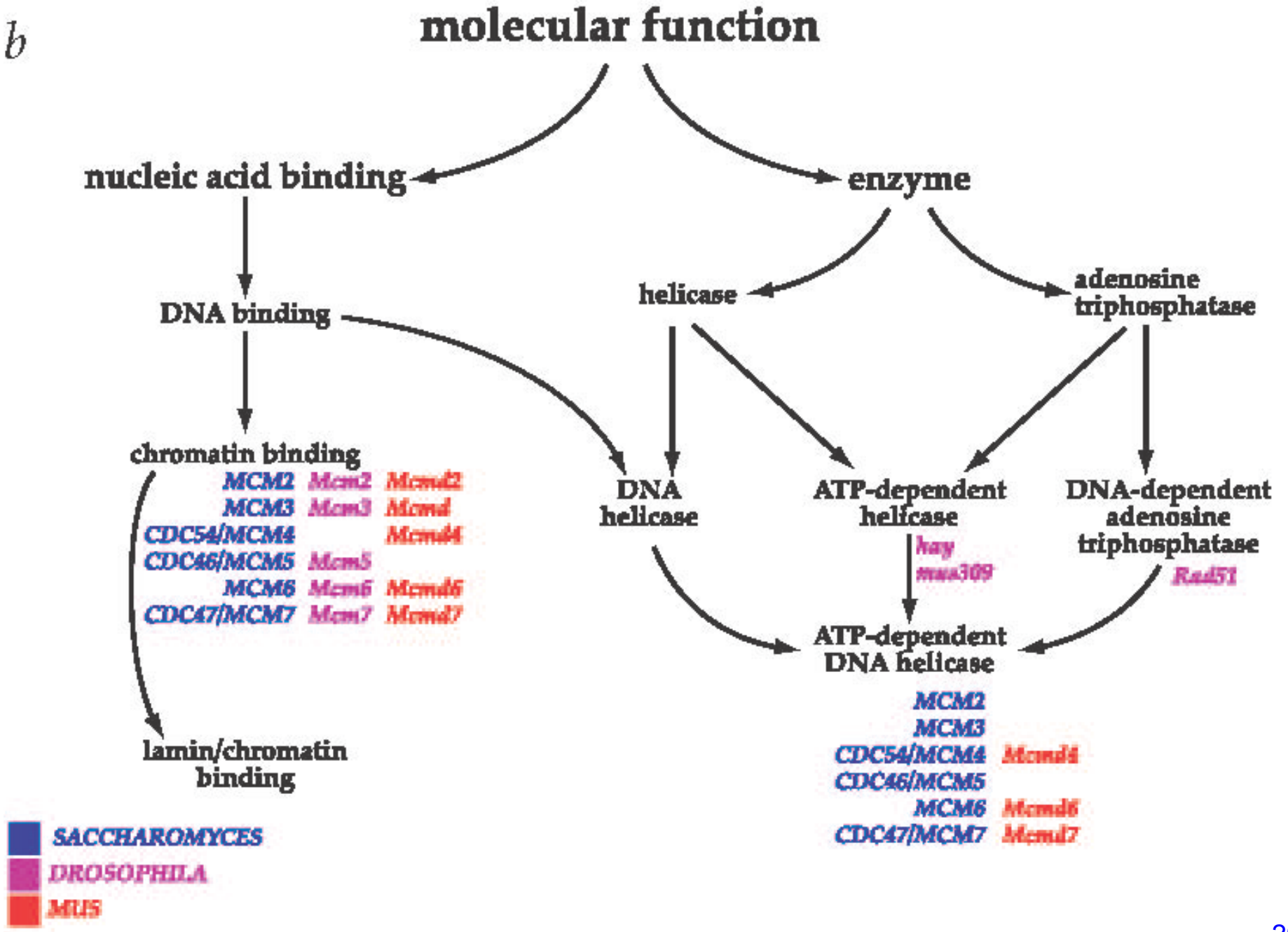
transcription



regulation of gluconeogenesis

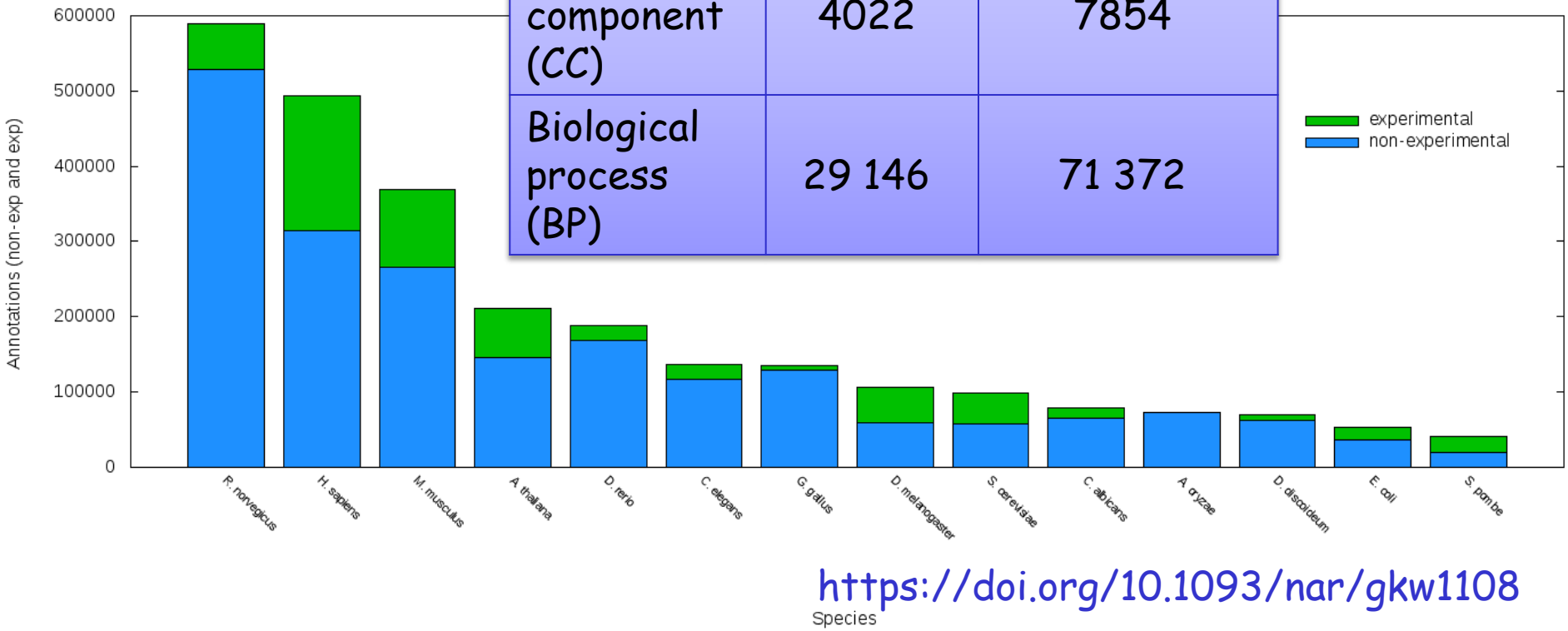


limb development


b
molecular function
nucleic acid binding
enzyme
DNA binding
helicase
adenosine triphosphatase
chromatin binding
MCM2 Mcm2 Mcmd2
MCM3 Mcm3 Mcmd
CDC54/MCM4 Mcmd4
CDC46/MCM5 Mcm5
MCM6 Mcm6 Mcmd6
CDC47/MCM7 Mcm7 Mcmd7
DNA helicase
ATP-dependent helicase
hay
mus309
DNA-dependent adenosine triphosphatase
Rad51
ATP-dependent DNA helicase
MCM2
MCM3
CDC54/MCM4 Mcmd4
CDC46/MCM5
MCM6 Mcmd6
CDC47/MCM7 Mcmd7
lamin/chromatin binding
SACCHAROMYCES
DROSOPHILA
MUS

Current metrics (2016)

Aspect	Terms	Relationships
Molecular function (MF)	10 417	14 039
Cellular component (CC)	4022	7854
Biological process (BP)	29 146	71 372

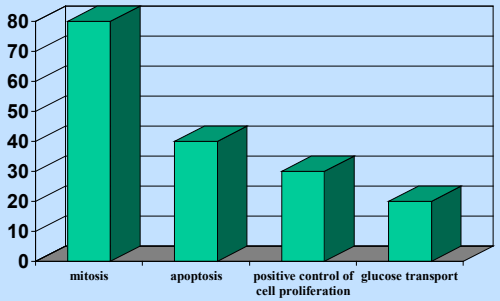


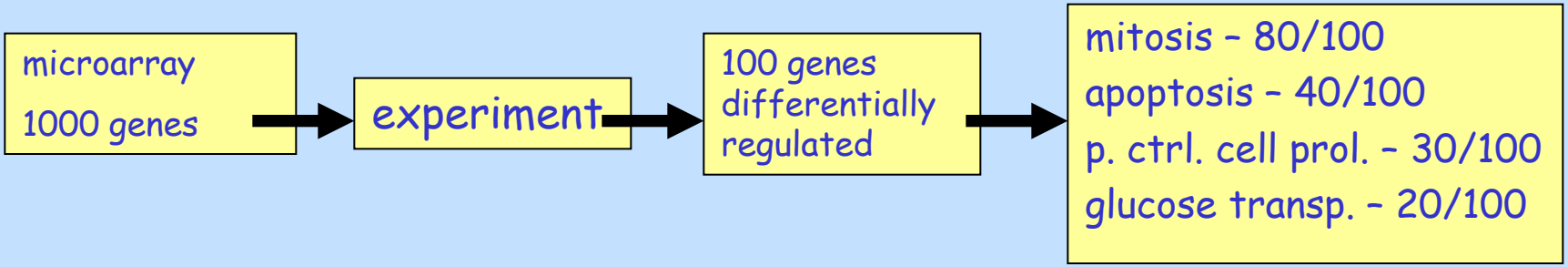
https://doi.org/10.1093/nar/gkw1108

TANGO

Using GO in practice

- I did an experiment and identified a group of 200 genes as having something "special" in common. What can I say about them? Can it help me understand relevant biology?





Using GO in practice (2)

- However, when you look at the distribution of all genes on the microarray:

Process	Genes on array	# genes expected in 100 random genes	I got
mitosis	800/1000	80	80
apoptosis	400/1000	40	40
p. ctrl. cell prol.	100/1000	10	30
glucose transp.	50/1000	5	20

- Need to
  - Normalize for term size,
  - Correct for the fact that we consider multiple terms
  - Account for dependencies
  - Compute statistical significance!

Statistical significance

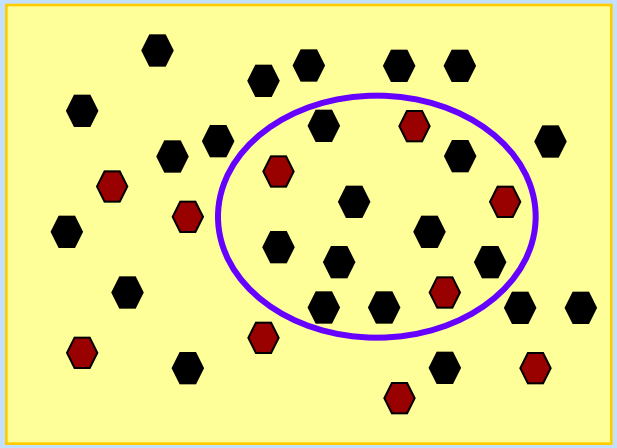
- Input: group(s) of genes, GO hierarchy (+ a term / annotation / function of interest)
- Background (BG) set: N genes, m of them annotated with the term
- Target (T): our subset of n genes, k with the term

- What is the chance of obtaining at least k genes with the term in the Target at random?
- Pr(overalp≥k)?

Reminder: Hypergeometric score

- Urn with N balls of which m are red.
- Draw n balls at random w/o replacement
- X = no. of red balls drawn

$$P(X=k)=\frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}}$$



$$HG(N,m,n,k)=\sum_{k'\geq k}P(X=k')$$

P-value for the chance that draw is random → measures enrichment

TANGO: Tool for Analysis of GO classes (Amos Tanay, 03)

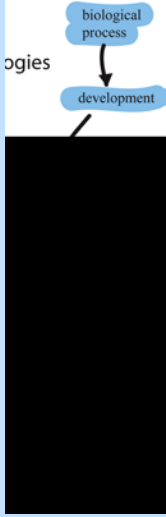
- Input: group(s) of genes, GO hierarchy
- Background (BG) set: N genes, m of them annotated with the function we are considering
- Target (T): our subset of n genes, k with the function
- What is the chance of obtaining at least k genes of the function in the Target at random?

- HG (N,m,n,k)

- But we usually test several target sets, each for many possible terms, and terms are dependent!

TANGO – corrections (1)

- Problem: Many candidate terms tested
- Solution: multiple testing correction
  - Bonferroni – way too stringent; FDR – still stringent
  - Strong dependencies between groups due to DAG structure
- TANGO solution: compute empirical distribution of the enrichment p-value
- For a given target set *Tj*, sample many random gene sets of the same size; compute their *p*-values vs. each of the terms *Ai*.
- Randomization: **permute gene IDs**. This keeps all of the relations among terms *Ai* and among target sets *Tj*, but decouples any dependency between them.
- Correct also for testing multiple target sets



TANGO – filtering redundancies

- Problem: after p-val correction, several related groups may be significant.
- Soln: greedy redundancy filtering
- Given target set T enriched for A', is T enriched for A as well?
- Given $|A \cap T|$, $|A \cap A'|$:

$$CondP(T, A \mid A') = HG(|A'|, |A \cap A'|, |T \cap A'|, |T \cap A \cap A'|) \times HG(n - |A'|, |A - A'|, |T - A'|, |(T - A') \cap A|)$$

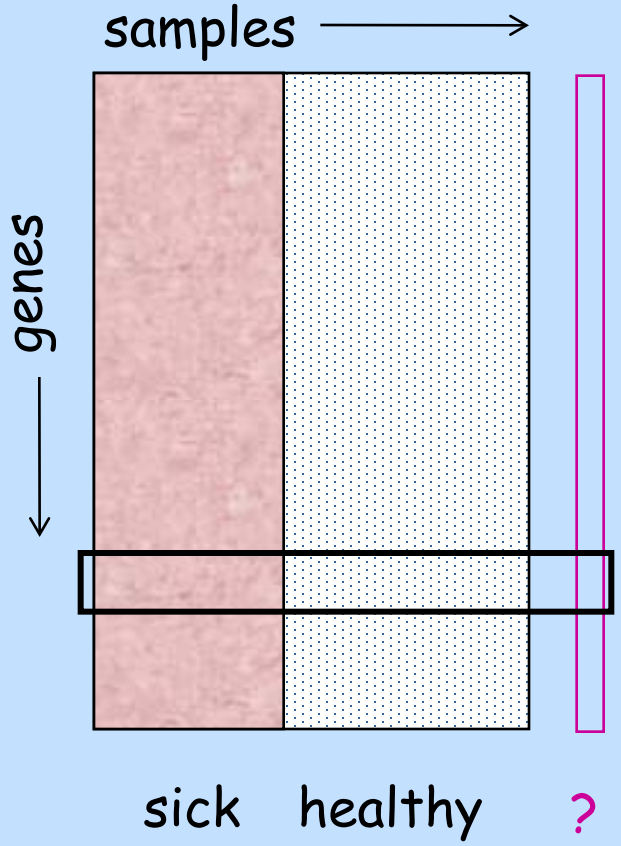
- For target set T, sort Ai by increasing p-vals, accept Aj as enriched only if CondP(T,Aj|Ai)<β for all i<j

- Test available in Expander.

GSEA

Motivation: Classification

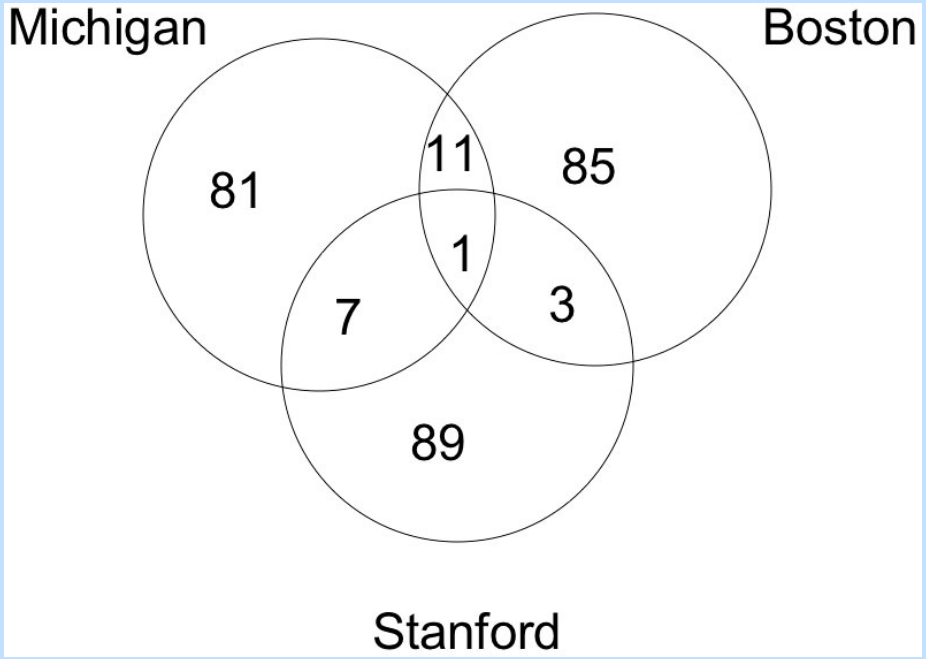
- Given a set of samples partitioned into two types, find a subset of genes that distinguishes between the types best on new samples



Motivation: Selecting genes one at a time gives poor robustness

- Pairwise and three-way overlap between the top 100 genes correlated with poor lung cancer outcome in the Michigan, Boston, and Stanford data sets. (Subramanian et al, PNAS 05)

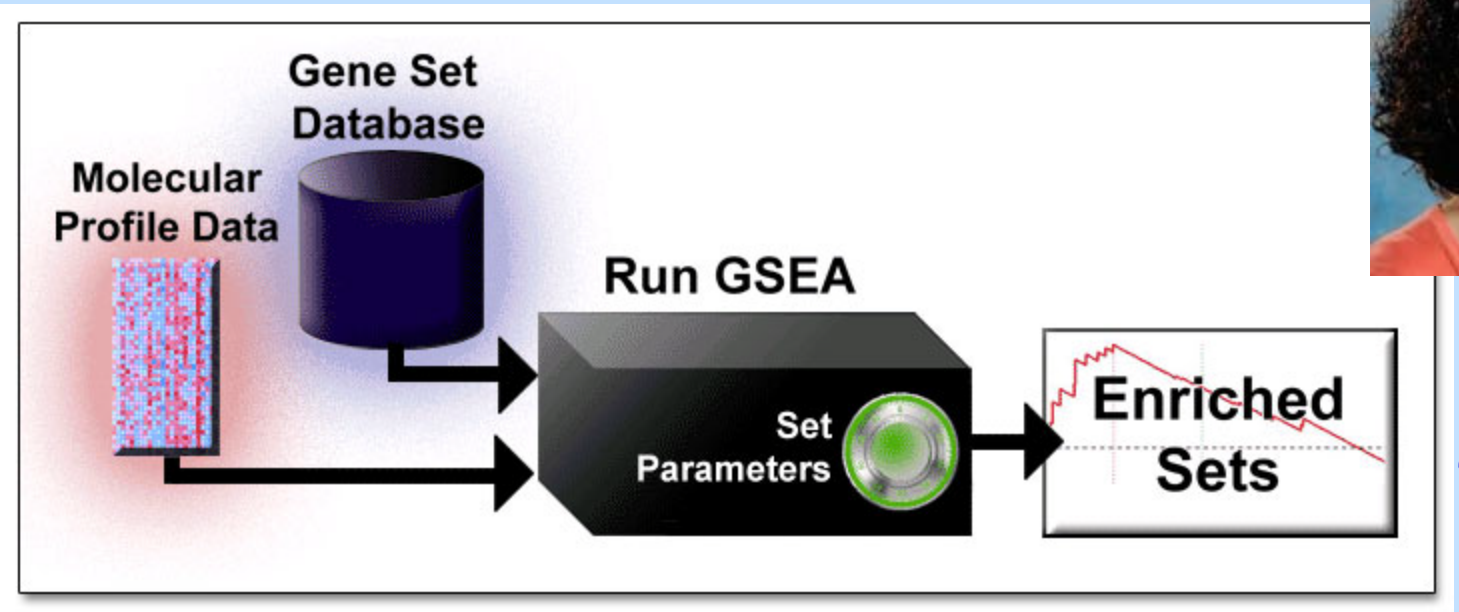
→ **Increase robustness by looking at groups**



Gene Set Enrichment Analysis (Subramanian et al 2005)

- GSEA determines whether an *a priori* defined set of genes shows statistically significant differences between two biological states.

Jill Mesirov



MSigDB – the gene sets

- ~18K sets from databases, literature and computational studies

- MSigDB Home
- About Collections
- Browse Gene Sets
- Search Gene Sets
- Investigate Gene Sets
- View Gene Families
- Help

MSigDB
Molecular Signatures Database

Molecular Signatures Database v6.1

Overview

The Molecular Signatures Database (MSigDB) is a collection of annotated gene sets for use with GSEA software. From this web site, you can

- **Search** for gene sets by keyword.
- **Browse** gene sets by name or collection.
- **Examine** a gene set and its annotations. See, for example, the GO_NOTCH_SIGNALING_PATHWAY gene set page.
- **Download** gene sets.
- **Investigate** gene sets:
  - **Compute overlaps** between your gene set and gene sets in MSigDB.
  - **Categorize** members of a gene set by gene families.
  - **View the expression profile** of a gene set in a provided public expression compendia.

License Terms

GSEA and MSigDB are available for use under these license terms.

Please register to download the GSEA software, access our web tools, and view the MSigDB gene sets. After registering, you can log in at any time using your email address. Registration is free. Its only purpose is to help us track usage for reports to our funding agencies.

Current Version

MSigDB database v6.1 updated October 2017. Release notes.
GSEA/MSigDB web site v6.2 released July 2017

Contributors

Collections

The MSigDB gene sets are divided into 8 major collections:

H **hallmark gene sets** are coherently expressed signatures derived by aggregating many MSigDB gene sets to represent well-defined biological states or processes.

C1 **positional gene sets** for each human chromosome and cytogenetic band.

C2 **curated gene sets** from online pathway databases, publications in PubMed, and knowledge of domain experts.

C3 **motif gene sets** based on conserved cis-regulatory motifs from a comparative analysis of the human, mouse, rat, and dog genomes.

C4 **computational gene sets** defined by mining large collections of cancer-oriented microarray data.

C5 **GO gene sets** consist of genes annotated by the same GO terms.

C6 **oncogenic gene sets** defined directly from microarray gene expression data from cancer gene perturbations.

C7 **immunologic gene sets** defined directly from microarray gene expression data from immunologic studies.

Citing the MSigDB

Inputs to GSEA

1. Expression data set *D* with *N* genes and *k* samples.
2. Ranking procedure to produce Gene List *L*. Includes a correlation (or other ranking metric) and a phenotype or profile of interest *C*.

E.g. given exp profiles of cases and controls, D can be the cases submatrix and C can be an average profile over the controls

3. An exponent *p* to control the weight of the step.
4. Independently derived gene set *S* of N_H genes e.g., *a GO term (or* a pathway, a cytogenetic band...)

Enrichment Score ES(S)

1. Rank order the N genes in D to form L: $g_1 \leq .. \leq g_N$ according to the correlation, $r(g_j)=r_j$, of their expression profiles with C.

Genes in S are called hits, not in S: misses.

2. Evaluate the fraction of weighted hits and misses among positions 1,…i in L.

$$P_{\text{hit}}(S, i) = \sum_{\substack{g_j \in S \\ j \leq i}} \frac{|r_j|^p}{N_R}, \quad \text{where } N_R = \sum_{g_j \in S} |r_j|^p$$

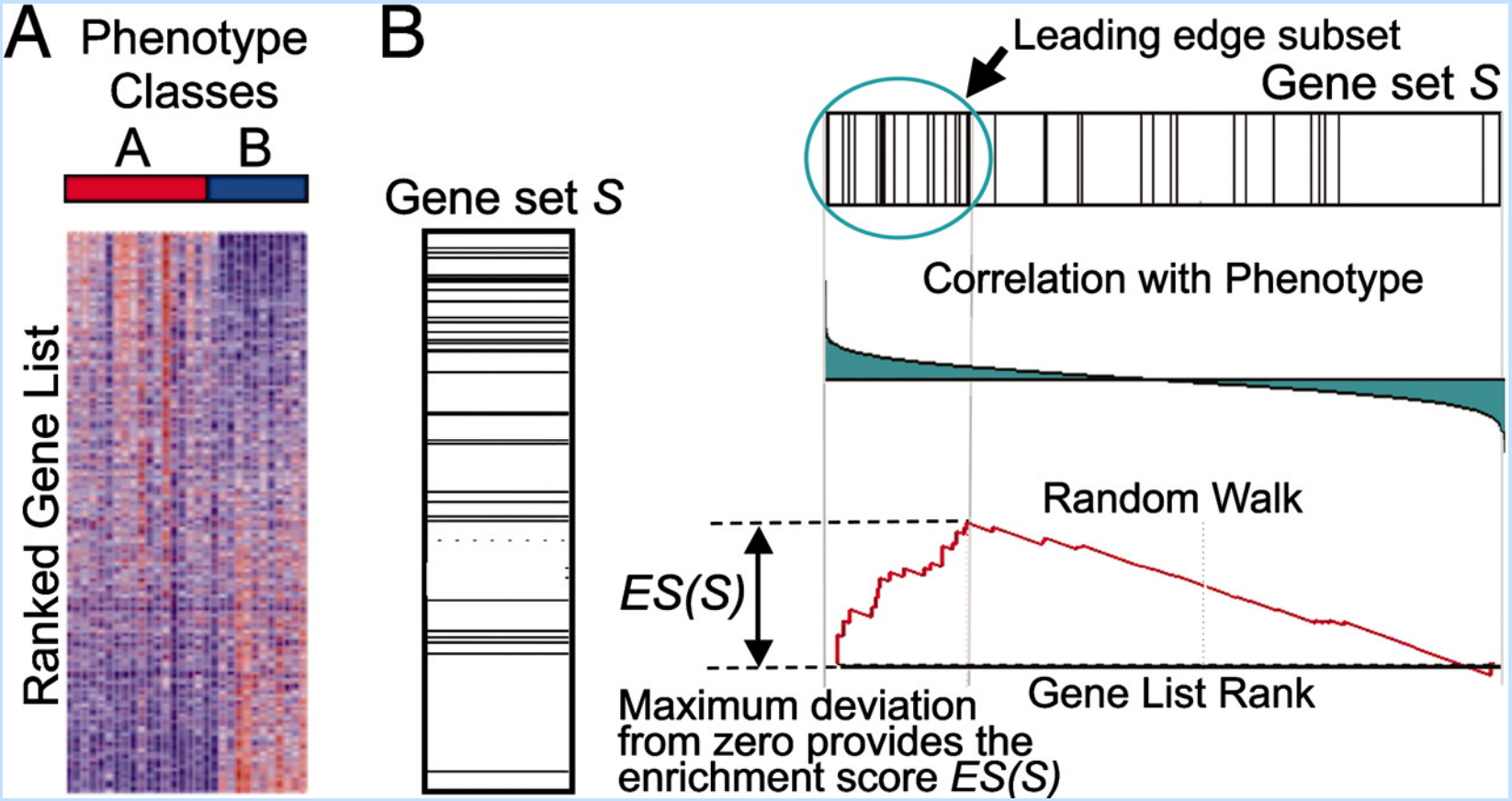
$$P_{\text{miss}}(S, i) = \sum_{\substack{g_j \notin S \\ j \leq i}} \frac{1}{(N - N_H)}.$$

Enrichment Score ES(S)

- $ES(s) = \max_i |P_{hit}(S,i) - P_{miss}(S,i)|$.

- When $p = 0$, $ES(S)$ reduces to the standard Kolmogorov–Smirnov statistic;
- When $p=1$, we are weighting the genes in S by their correlation with C, normalized by the sum of the correlations over all of the genes in S.
- $p=1$ was used in the paper.

A GSEA overview illustrating the method



Subramanian A et al. PNAS 2005;102:15545-15550

Estimating Significance

Compare the observed S with the set of scores ES_{NULL} computed with randomly permuted phenotypes.

1. Randomly assign the original phenotype labels to samples, reorder genes, and re-compute $ES(S)$.
2. Repeat step 1 for 1,000 permutations, and create a histogram of the enrichment scores ES_{NULL}.
3. Estimate nominal P value for S from ES_{NULL} (use the positive or negative portion of the distribution corresponding to the sign of the observed $ES(S)$.)

Multiple Hypothesis Testing

1. Determine ***ES(S)*** for each gene set in the collection.
2. For each *S* and 1000 fixed permutations π of the phenotype labels, reorder the genes in *L* and determine ***ES(S,π)***.
3. **Adjust for variation in gene set size:** Normalize the *ES*(*S*, π) and the observed *ES*(*S*) by dividing by the mean of the *ES*(*S*, π) to yield the normalized scores *NES*(*S*, π) and *NES*(*S*).

$$NES(S,\pi) = \frac{ES(S,\pi)}{\underset{ES(S,\pi))\geq 0}{AVE}[ES(S,\pi)]} \quad \text{if } ES(S,\pi) \geq 0$$

$$NES(S) = \frac{ES(S)}{\underset{ES(S,\pi))\geq 0}{AVE}[ES(S,\pi)]} \quad \text{if } ES(S) \geq 0$$

This is done separately for the positive and negative scores.

Multiple Hypothesis Testing (2)

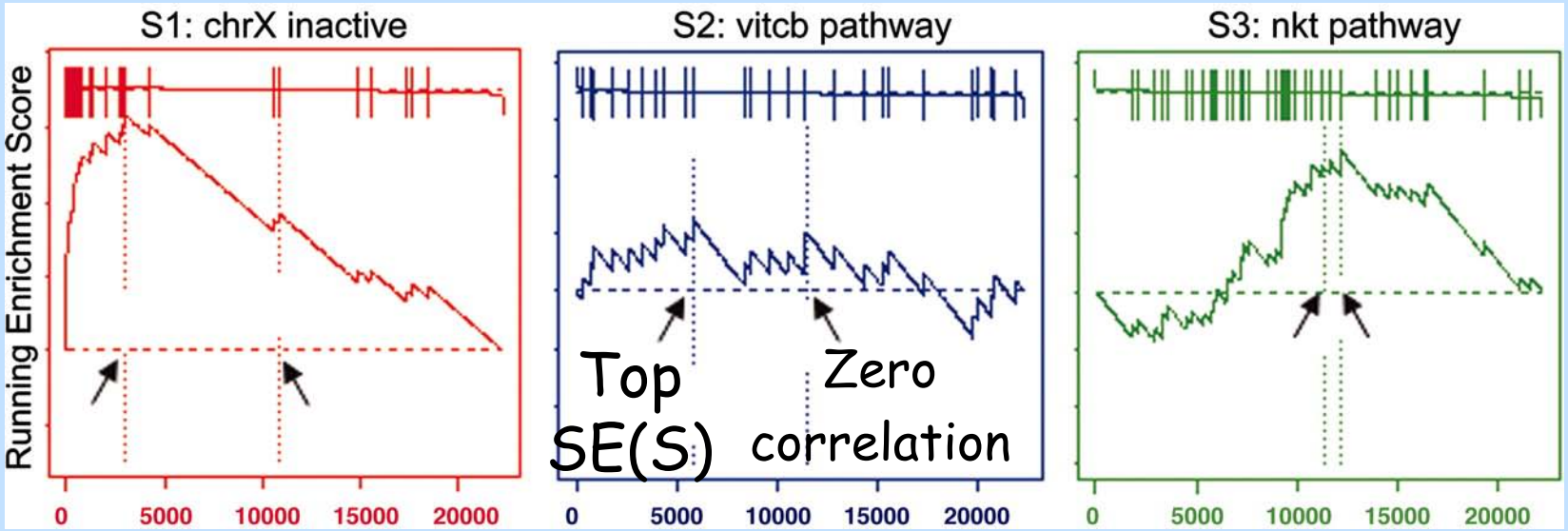
4. **Compute FDR**. Control the ratio of false positives to the total number of gene sets attaining a fixed level of significance:

   Create a histogram of all $NES(S, \pi)$ over all S and π. Use this null distribution to compute an FDR q value, for a given $NES(S) = \alpha \geq 0$.

$$q = \frac{|\{(S,\pi) \mid NES(S,\pi) \geq \alpha\}| / |\{(S,\pi) \mid NES(S,\pi) \geq 0\}|}{|\{S \mid NES(S) \geq \alpha\}| / |\{S \mid NES(S) \geq 0\}|}$$

This is done separately for positive (negative) $NES(S)$ and $NES(S, \pi)$

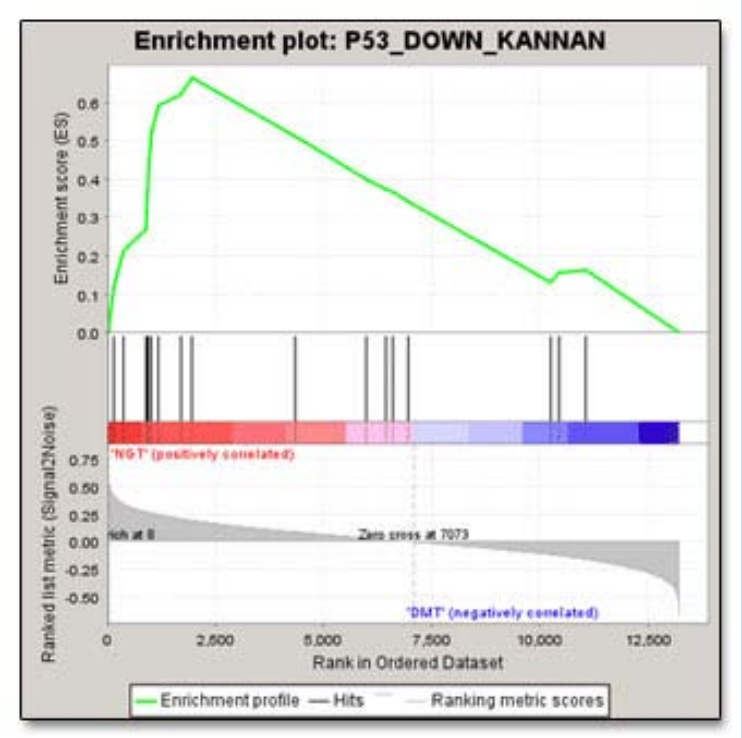
Original (4) enrichment score behaviour – using unweighted ranks



Subramanian A et al. PNAS 2005;102:15545-15550

GSEA Output

- Enrichment Plot
- Gene List
- Gene Set Information



	PROBE	GENE SYMBOL	GENE_TITLE	RANK IN GENE LIST	RANK METRIC SCORE	RUNNING ES	CORE ENRICHMENT
1	ARL5	ARL5 Entrez, Source, GeneCards	ADP ribosylation factor-like 5	161	0.404	0.1203	Yes
2	INA	INA Entrez, Source, GeneCards	internexin neuronal intermediate filament protein, alpha	379	0.339	0.2163	Yes

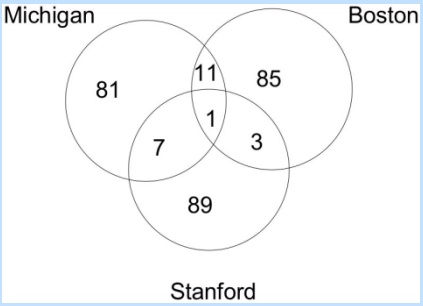
Results – male vs. female

- 15 males and 17 females lymphoblastoid cell lines
- Sought gene sets correlated with male>female, female>male

Table 2. Summary of GSEA results with FDR ≤ 0.25

Gene set	FDR
Data set: Lymphoblast cell lines	
Enriched in males	
chrY	<0.001
chrYp11	<0.001
chrYq11	<0.001
Testis expressed genes	0.012
Enriched in females	
X inactivation genes	<0.001
Female reproductive tissue expressed genes	0.045

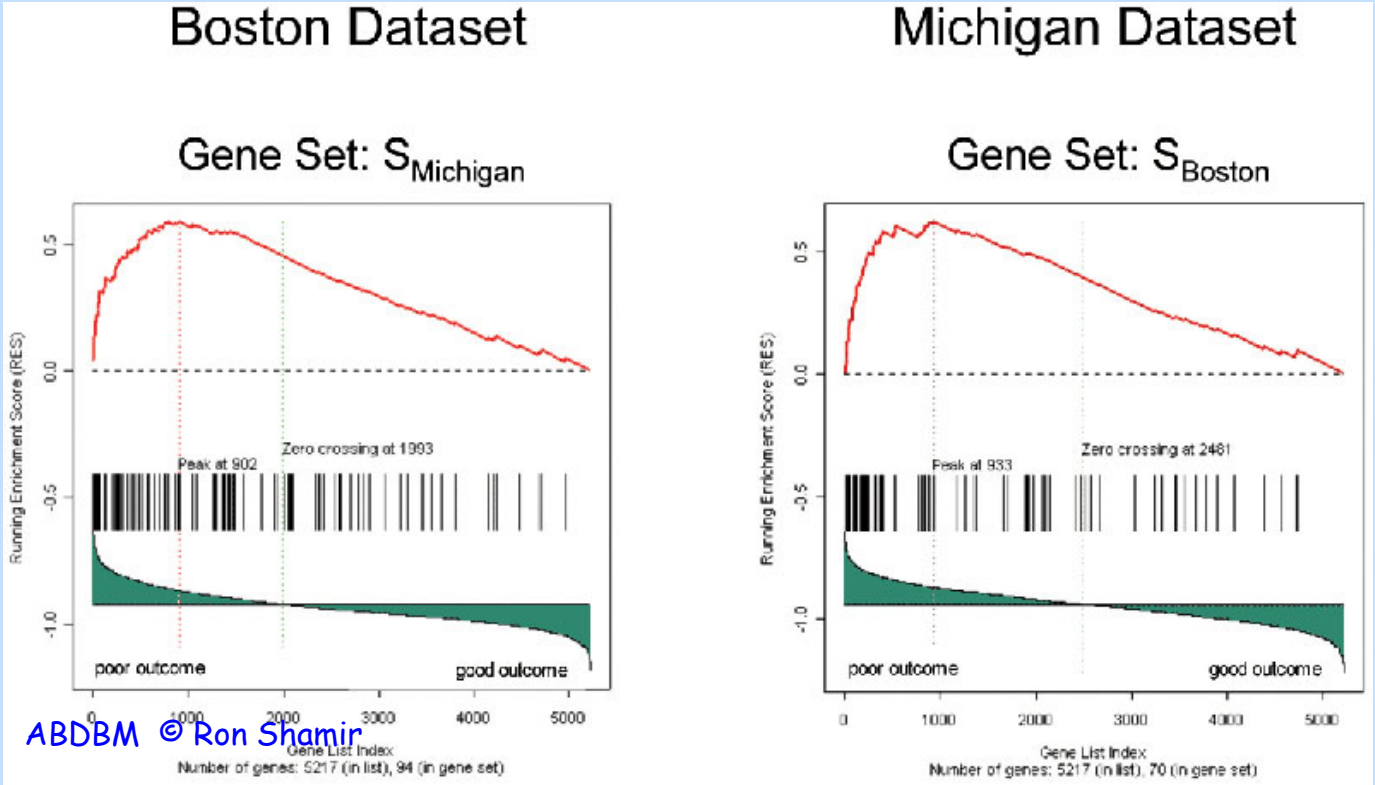
Results – Lung cancer



- Reanalyzed two lung cancer studies: Each obtained ~70 exp profiles for patients that were classified as good/poor outcome
- Little overlap in the genes most correlated with the outcome
- No single gene (!) in either study significantly associated with the outcome ($q<0.05$)
- Using the MsigDB sets on the two datasets – four significant gene sets in common

Results – Lung cancer (2)

- Defined a set of the 100 genes most associated with the outcome that occur also in the other study – ran GSEA on the ranking from the other study with that set → significant results both ways



Hypergeometric in Practice

phyper(q, m, n, k, lower.tail = TRUE, log.p = FALSE)

q -> vector of quantiles representing the number of white balls drawn without replacement from an urn which contains both black and white balls.

m -> the number of white balls in the urn.

n -> the number of black balls in the urn.

k -> the number of balls drawn from the urn.

lower.tail -> logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Hypergeometric in Practice

```
#Functions
p.overlap.gene_set <- function(total_genes, geneset_genes, significant_genes, lower.tail=FALSE) {
        total_genes_in_set <- length(which(total_genes %in% geneset_genes))
        total_genes_not_in_set <- length(total_genes) - total_genes_in_set
        significant_genes_in_set <- length(which(significant_genes %in% geneset_genes))
        c(phyper(significant_genes_in_set, total_genes_in_set, total_genes_not_in_set, length(significant_genes),
lower.tail=lower.tail), significant_genes_in_set)
}

gene_set_stats <- function(total_genes, gene_set_list, significant_genes){
        gs_stats <- c()
        for(i in 1:length(gene_set_list)){
                aux_gs <- gene_set_list[[i]]
                aux_gs <- aux_gs[which(aux_gs %in% total_genes)]
                gs_stats <- rbind(gs_stats, p.overlap.gene_set(total_genes, aux_gs, significant_genes))
        }
        rownames(gs_stats) <- names(gene_set_list)
        gs_stats <- cbind(gs_stats, p.adjust(gs_stats[,1]))
        colnames(gs_stats) <- c("p.value", "Num", "FDR")
        gs_stats <- gs_stats[-which(gs_stats[,"Num"] < 2),]
        gs_stats <- gs_stats[,c(1,3,2)]
        gs_stats <- gs_stats[order(gs_stats[,1], -gs_stats[,3]),]
        gs_stats
}
```

GSEA in Practice

```
#Cor = Multiply sign(fold-change) * -log10(p-value)
exp_cor <- cbind(exp_cor, apply(exp_cor, 1, function(x) sign(x[1]) * -log10(x[2])))
#List Gene Name and Cor, order by Cor in decreasing order
cor_list <- cbind(rownames(exp_cor)[order(exp_cor[,3], decreasing=TRUE)],
exp_cor[order(exp_cor[,3], decreasing=TRUE),3])
rnk_file <- "IACS-10579_hypoxia_auc_rank.rnk"
write.table(cor_list, sep="\t", quote=F, row.names=F, col.names=F, file=rnk_file)

java_command <- "java -cp gsea2-2.2.0.jar -Xmx1024m xtools.gsea.GseaPreranked"
java_command <- paste(java_command, "-gmx c2.all.v5.1.symbols.gmt -collapse false -mode
Max_probe -norm meandiv -nperm 1000 -rnk")
java_command <- paste(java_command, "IACS_10579_auc_rank.rnk -scoring_scheme
weighted -rpt_label IACS_10579_auc -include_only_symbols true")
java_command <- paste(java_command, "-make_sets true -plot_top_x 30 -rnd_seed
timestamp -set_max 500 -set_min 5 -zip_report false -out")
java_command <- paste(java_command, "GSEA_results -gui false")
system(java_command)
```