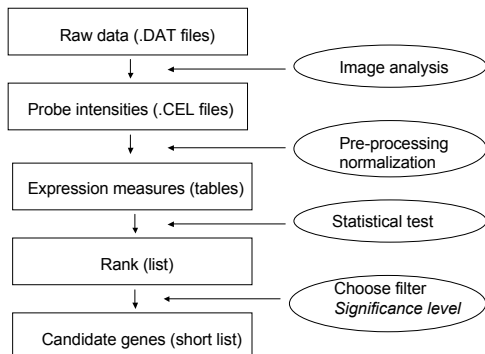


PART B

Differential Expression in Two Populations

Work flow



Introduction

by Terry Speed

- Many microarray experiments are carried out to find genes which are differentially expressed between two (or more) samples of cells. Examples abound:
- Initially, comparative microarray experiments were done with few, if any replicates, and statistical criteria were not used for identifying differentially expressed genes. Instead, simple criteria were used such as fold-change, with 2-fold being a popular cut-off.
- It did not take long for people to want to assign statistical significance to their findings concerning differentially expressed genes. Could p -values be attached, confidence statements be made, and so on? These questions raised a number of issues which were unfamiliar to the molecular biologists doing the experiments: replication, systematic versus random differences, multiplicity of tests, etc.

The simplest cDNA microarray data analysis problem is identifying differentially expressed genes using one slide

- This is a common enough hope
- Efforts are frequently successful
- It is not hard to do by eye
- The problem is probably beyond formal statistical inference (valid p -values, etc) for the foreseeable future....why?

The second simplest cDNA microarray data analysis problem is identifying differentially expressed genes using replicated hybridizations

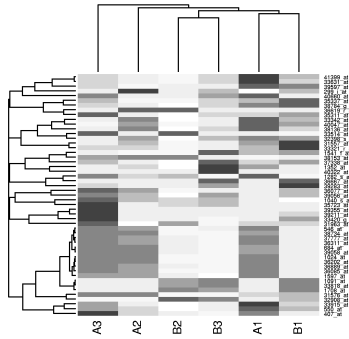
There are a number of different aspects:

- First, between-slide normalization; then
- What should we look at: averages, SDs, t -statistics, other summaries?
- How should we look at them?
- Can we make valid probability statements?

Let us start with EDA

- If we are interested in finding a few genes that are differentially expressed it seems obvious we want to plot differences of average (usually of log intensities)
- However, some use heat maps as default

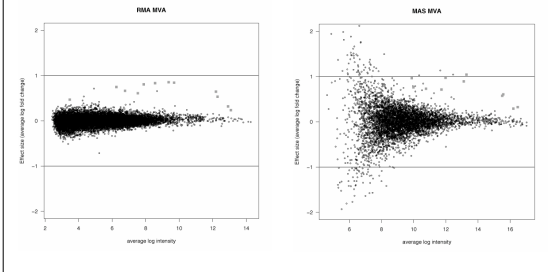
Clustering is not a good tool



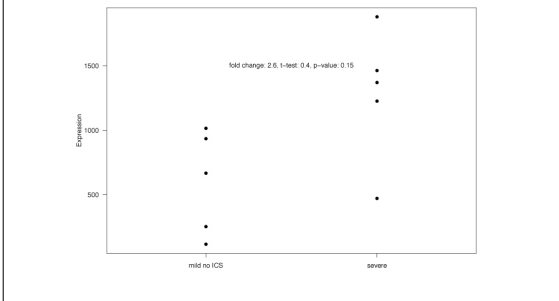
Back to Basics

- If we are interested in genes with over-all large fold changes why not look at average log ratios?
- We can make MA plots:
 - M = difference in average log intensities and
 - A = average of average log intensities

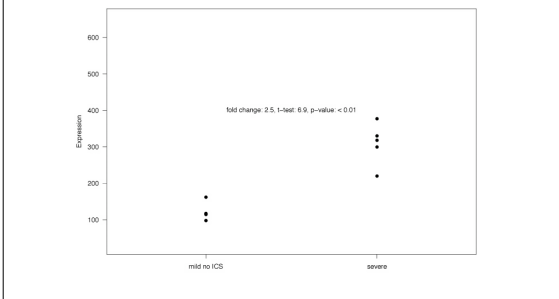
MA plot of average log ratios much better



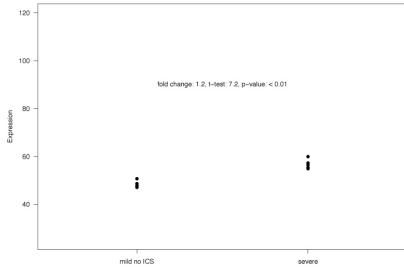
Should we consider variability?



Should we consider variability?



Should we consider variability?



Back to Basics

- If N and M are big then the t-statistic is normally distributed with mean 0 and SD of 1
- If the observed data is normally distributed then the t-statistic follow a t distribution regardless of N,M
- Regardless, the square of the t-test is proportional to the ratio of across group variance to within group variance

Back to Basics

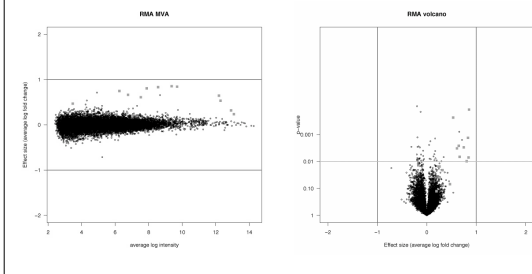
t - statistic squared if N=M:

$$N \times \frac{(\bar{Y} - \bar{X})^2}{s_Y^2 + s_X^2}$$

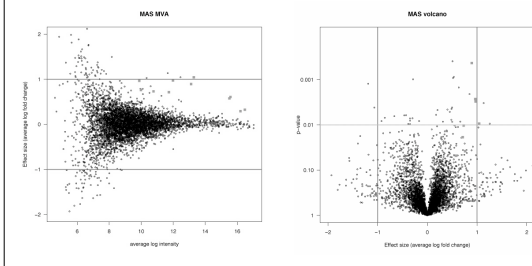
Useful Plots

- The MA plot shows $M = \log$ fold change, plotted against $A = \text{average log intensity}$
- If we have various replicates in each population we can plot $M = \text{difference in log averages in two populations}$
- The volcano plot shows, for a particular test, negative \log p-value against the effect size (M)
- How do we get p-values? Are they really p-values?

With RMA t-test is not powerful



If you insist on using MAS 5.0 it really helps



Estimating the variance

- If different genes (or probes) have different variation then it is not a good idea to use average log ratios even if we do not care about significance
- Under a random model we need to estimate the SD
- The t-test divides by SD
- But with few replicates, estimates of SD are not stable
- This explains why t-test is not powerful
- There are many proposals for estimating variation
- Many *borrow strength* across genes
- Empirical Bayesian Approaches are popular
- SAM, an ad-hoc procedure, is even more popular

Some Examples of Tests

Notation:

- T is average log expression in Tx
- C is average log expression in Control
- S is SD

- Note taking log before average is important

Tests:

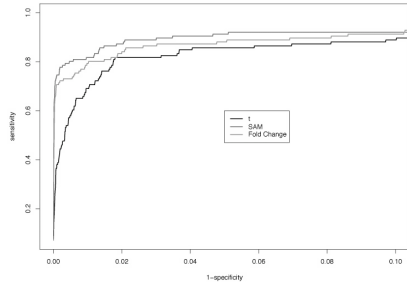
- Average log fold-change: $(T-C)$
- t-statistic: $(T-C) / S$
- SAM shrunken t-statistic: $(T-C) / (S + S_0)$
- Bayesian posteriors: $(T-C) / \sqrt{(S^2+K^2)}$
- Wilcoxon: Rank test
- Ad-hoc pairwise comparison: No formula

Many of these are in the `limma` package
SAM is in the `siggenes` package
Also look at `Ebayes` package

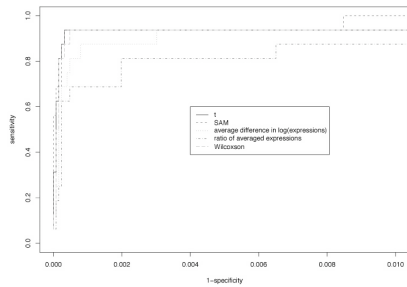
ROC curves

- ROC = Receiver Operator Characteristic
- To compare tests it is important to look at both specificity and sensitivity
- For every cut-off value there will be some true positives and some false positives
- We can make a curve that plots true positives versus false positives as we move the cut-off

Does it make a difference (N=3)?

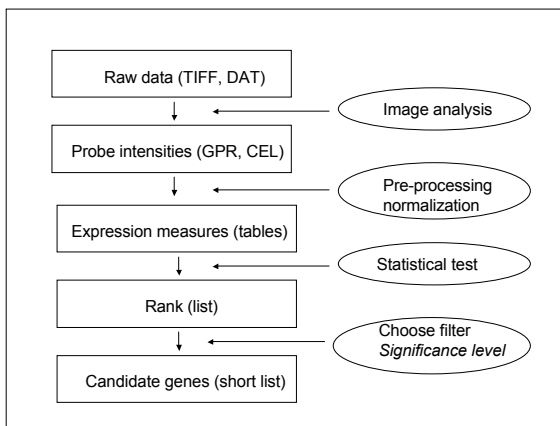


Does it make a difference (N=12)?



Demo

The Multiple Comparison Problem



Hypothesis testing

- Once you have a score for each gene, how do you decide on a cut-off? p-values are popular. Are they appropriate?
- Test for each gene null hypothesis: no differential expression.
- Two types of errors can be committed
 - Type I error or false positive (say that a gene is differentially expressed when it is not, i.e., reject a true null hypothesis).
 - Type II error or false negative (fail to identify a truly differentially expressed gene, i.e., fail to reject a false null hypothesis)

Multiple comparison:

- We want to know if somebody has nickles that are more likely to be heads than their dimes (differentially expressed coins):
- With five coins, most convincing evidence is:
nickles: HHHHH, dimes:TTTT
- How likely is this?
- Under the null, p-value is 1 in 1024 ($p < 0.001$).
- If we test 10000 how many do we expect to see?
- Even when null hypothesis is true, we expect to see about 10 such cases.

What do we do?

- Bonferoni correction (too conservative)
- Give list of genes and report:
 - Family-Wise Error Rate: probability of including at least one non-differentially expressed gene
 - False discovery rate (FDR): expected proportion of Type I errors among the rejected hypotheses
 - pFDR: Expected proportion of false discoveries among the genes in your list conditioning on at least one gene is included in the differential list.
- Bayesian inference
- Forget about inference: use EDA

Many of these are in the mult test package

Demo

Class Prediction

Class Prediction Model

- Given a sample with an expression profile vector x of log-ratios or log signals and unknown class.
- Predict which class the sample belongs to
- The class prediction model is a function f which maps from the set of vectors x to the set of class labels $\{1,2\}$ (if there are two classes).
- f generally utilizes only some of the components of x (i.e. only some of the genes)
- Specifying the model f involves specifying some parameters (e.g. regression coefficients) by fitting the model to the data (*learning* the data).

Do Not Confuse Statistical Methods Appropriate for Class Comparison with Those Appropriate for Class Prediction

- Demonstrating statistical significance of prognostic factors is not the same as demonstrating predictive accuracy.
- Demonstrating goodness of fit of a model to the data used to develop it is not a demonstration of predictive accuracy.
- Statisticians are used to inference, not prediction
- Most statistical methods were not developed for $p \gg n$ prediction problems

Components of Class Prediction

- **Feature (gene) selection**
 - Which genes will be included in the model
- **Select model type**
 - E.g. DLDA, Nearest-Neighbor, ...
- **Fitting parameters (regression coefficients) for model**

Feature Selection

- **Key component of supervised analysis**
- **Genes that are univariately differentially expressed among the classes at a significance level α (e.g. 0.01)**
 - The α level is selected to control the number of genes in the model, not to control the false discovery rate
- **Small subset of genes which together give most accurate predictions**
- **Many published complex methods for selecting combinations of genes do not appear to have been properly evaluated**

Linear Classifiers for Two Classes

- **Fisher linear discriminant analysis (weights based on assumed multi-variate normal distribution of expression vector in each class with common covariance matrix)**
- **Diagonal linear discriminant analysis (DLDA) assumes features are uncorrelated**
- **Compound covariate predictor and Golub's weighted voting method are variants of DLDA**

Linear Classifiers for Two Classes

- Support vector machines with inner product kernel are linear classifiers with weights determined to minimize errors
- Perceptrons with principal components as input are linear classifiers with no well defined criterion for defining weights

Weighted Gene Voting is DLDA

With equal priors, DLDA is:

With two classes we select class 1 if

This can be written as

with

Weighted Gene Voting simply uses

Notice the units and scale for sum are wrong!

Nearest Neighbor Classifier

- To classify a sample in the validation set as being in outcome class 1 or outcome class 2, determine which sample in the training set it's gene expression profile is most similar to.
 - Similarity measure used is based on genes selected as being univariately differentially expressed between the classes
 - Correlation similarity or Euclidean distance generally used
- Classify the sample as being in the same class as it's nearest neighbor in the training set

Advantages of Simple Classifiers

- **Do not over-fit data**
 - Incorporate influence of multiple variables without attempting to select the best small subset of variables
 - Do not attempt to model the multivariate interactions among the predictors and outcome

- **When $p \gg n$, a linear classifier can almost always be found which fits the data perfectly.**
- **Why consider more complex models?**
- **The full set of linear models is too rich and selecting a linear model to minimize training errors does not lead to generalizable results when $p \gg n$.**

Some Points

- **That complex classification algorithms such as neural networks perform better than simpler methods for class prediction.**
- **Artificial intelligence sells to journal reviewers and peers who cannot distinguish hype from substance when it comes to microarray data analysis.**
- **Comparative studies have shown that simpler methods work as well or better for microarray problems because the number of candidate predictors exceeds the number of samples by orders of magnitude. (Dudoit, Fridlyand and Speed JASA 2001)**

Evaluating a Classifier

- “Prediction is difficult, especially the future.”
 - Neils Bohr
- Fit of a model to the same data used to develop it is no evidence of prediction accuracy for independent data.

Split-Sample Evaluation

- **Training-set**
 - Used to select features, select model type, determine parameters and cut-off thresholds
- **Test-set**
 - Withheld until a single model is fully specified using the training-set.
 - Fully specified model is applied to the expression profiles in the test-set to predict class labels.
 - Number of errors is counted
- **Example: Leave one out cross-validation, leave 10%-out cross validation**
- **Publications are using all the data to select genes and then cross-validating only the parameter estimation component of model development, which gives biased estimates of error rates**

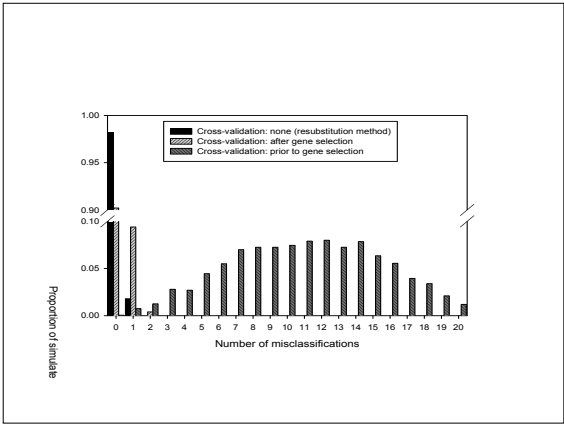
Prediction on Simulated Null Data

Generation of Gene Expression Profiles

- 20 specimens (P_i is the expression profile for specimen i)
- Log-ratio measurements on 6000 genes
- $P_i \sim \text{MVN}(\mathbf{0}, \mathbf{I}_{6000})$
- Can we distinguish between the first 10 specimens (Class 1) and the last 10 (Class 2)?

Prediction Method

- Compound covariate prediction
- Compound covariate built from the log-ratios of the 10 most differentially expressed genes.



A subtle problem

- Let $M(b,D)$ denote a classification model developed on a set of data D where the model is of a particular type that is parameterized by a scalar b .
- Use cross-validation to estimate the classification error of $M(b,D)$ for a grid of values of b ; $Err(b)$.
- Select the value of b^* that minimizes $Err(b)$.
- Caution: $Err(b^*)$ is a biased estimate of the prediction error of $M(b^*,D)$.
- This error is made in some commonly used methods

Potential Sources of Bias in Estimation of Error Rates

- **Confounding by sample handling or assay effects**
 - Design evaluation carefully
- **Failure to incorporate important sources of future variability**
 - Assay drift
- **Change in distribution of un-modeled variables**
 - In split sample validation, split samples by institution

R class prediction packages

- `class`
 - `k-nearest neighbor (knn)`
 - `learning vector quantization (lvg)`
- `classFP`: **projection pursuit.**
- `e1071`: **support vector machines (svm).**
- `ipred`: **bagging, resampling based estimation of prediction error.**
- `knnTree`: **k-nn classification with variable selection inside leaves of a tree.**
- `LogitBoost`: **boosting for tree stumps.**
- `MASS`: **linear and quadratic discriminant analysis (lda, qda).**
- `mlbench`: **machine learning benchmark problems.**
- `nnet`: **feed-forward neural networks and multinomial log-linear models.**
- `pamR`: **prediction analysis for microarrays.**
- `randomForest`: **random forests.**
- `rpart`: **classification and regression trees.**
- `sma`: **diagonal linear and quadratic discriminant analysis, naïve Bayes (stat.diag.da).**

Download
from CRAN

Annotation

Annotation

- **One of the largest challenges in analyzing genomic data is associating the experimental data with the available biological metadata, e.g., sequence, gene annotation, chromosomal maps, literature.**
- **AND MAKING THAT DATA AVAILABLE FOR COMPUTATION**
- **Bioconductor provides three main packages for this purpose:**
 - `annotate` (end-user);
 - `AnnBuilder` (developer)
 - `annaffy` (end-user – will see a name change)

WWW resources

- Nucleotide databases: e.g. GenBank.
- Gene databases: e.g. LocusLink, UniGene.
- Protein sequence and structure databases: e.g. SwissProt, Protein DataBank (PDB).
- Literature databases: e.g. PubMed, OMIM.
- Chromosome maps: e.g. NCBI Map Viewer.
- Pathways: e.g. KEGG.
- Entrez is a search and retrieval system that integrates information from databases at NCBI (National Center for Biotechnology Information).
- if you know of some we should be using – please let us know

annotate: matching IDs

Important tasks

- Associate manufacturers or in-house probe identifiers to other available identifiers.

E.g.

Affymetrix IDs → LocusLink LocusID

Affymetrix IDs → GenBank accession number.

- Associate probes with biological data such as chromosomal position, pathways.
- Associate probes with published literature data via PubMed (need PMID).

annotate: matching IDs

Affymetrix identifier	"41046_s_at"
HGU95A chips	
LocusLink, LocusID	"9203"
GenBank accession #	"X95808"
Gene symbol	"ZNF261"
PubMed, PMID	"10486218" "9205841" "8817323"
Chromosomal location	"X", "Xq13.1"

Annotation data packages

- The Bioconductor project provides annotation data packages, that contain many different mappings to interesting data
 - Mappings between Affy IDs and other probe IDs: hgu95av2 for HGU95Av2 GeneChip series, also, hgu133a, hu6800, mgu74a, rgu34a, YG.
 - Affy CDF data packages.
 - Probe sequence data packages.
- These packages are updated and expanded regularly as new data become available.
- They can be downloaded from the Bioconductor website and also using `installDataPackage`.
- `DPEXplorer`: a widget for interacting with data packages.
- `AnnBuilder`: tools for building annotation data packages.

annotate: matching IDs

- Much of what `annotate` does relies on matching symbols.
- This is basically the role of a hash table in most programming languages.
- In R, we rely on environments.
- The annotation data packages provide R environment objects containing key and value pairs for the mappings between two sets of probe identifiers.
- Keys can be accessed using the `R` `ls` function.
- Matching values in different environments can be accessed using the `get` or `multiget` functions.

annotate: matching IDs

```
> library(hgu95av2)
> get("41046_s_at", env = hgu95av2ACCNUM)
[1] "X95808"
> get("41046_s_at", env = hgu95av2LOCUSID)
[1] "9203"
> get("41046_s_at", env = hgu95av2SYMBOL)
[1] "ZNF261"
> get("41046_s_at", env = hgu95av2GENENAME)
[1] "zinc finger protein 261"
> get("41046_s_at", env = hgu95av2SUMFUNC)
[1] "Contains a putative zinc-binding motif
(MYM)|Proteome"
> get("41046_s_at", env = hgu95av2UNIGENE)
[1] "Hs.9568"
```

annotate: matching IDs

```
> get("41046_s_at", env = hgu95av2CHR)
[1] "X"
> get("41046_s_at", env = hgu95av2CHRLOC)
  X
-68692698
> get("41046_s_at", env = hgu95av2MAP)
[1] "Xq13.1"
> get("41046_s_at", env = hgu95av2PMID)
[1] "10486218" "9205841" "8817323"
> get("41046_s_at", env = hgu95av2GO)
  TAS      TAS      IEA
"GO:0003677" "GO:0007275" "GO:0016021"
```

annotate: matching IDs

- Instead of relying on the general R functions for environments, new user-friendly functions have been written for accessing and working with specific identifiers.
- E.g. `getGO`, `getGODesc`, `getLL`, `getPMID`, `getSYMBOL`.

annotate: matching IDs

```
> getSYMBOL("41046_s_at", data="hgu95av2")
41046_s_at
"ZNF261"
> gg<- getGO("41046_s_at", data="hgu95av2")
> getGODesc(gg[[1]], "MF")
$"GO:0003677"

"DNA binding activity"
> getLL("41046_s_at", data="hgu95av2")
41046_s_at
9203
> getPMID("41046_s_at", data="hgu95av2")
$"41046_s_at"
[1] 10486218 9205841 8817323
```

annotate: querying databases

The `annotate` package provides tools for

- Searching and processing information from various WWW biological databases
 - GenBank,
 - LocusLink,
 - PubMed.
- Regular expression searching of PubMed abstracts.
- Generating nice HTML reports of analyses, with links to biological databases.

annotate: WWW queries

- Functions for querying WWW databases from R rely on the `browseURL` function

```
browseURL("www.r-project.org")
```

Other tools: `HTMLPage` class, `getTDRows`, `getQueryLink`, `getQuery4UG`, `getQuery4LL`, `makeAnchor`.

- The `XML` package is used to parse query results.

annotate: querying GenBank

www.ncbi.nlm.nih.gov/Genbank/index.html

- Given a vector of GenBank accession numbers or NCBI UIDs, the `genbank` function
 - opens a browser at the URLs for the corresponding GenBank queries;
 - returns an `XMLdoc` object with the same data.

```
genbank("X95808", disp="browser")
```

<http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?tool=blcoondoc&cmd=Search&db=Nucleotide&term=X95808>

```
genbank(1430782, disp="data",  
type="uid")
```

annotate: querying LocusLink
www.ncbi.nlm.nih.gov/LocusLink/

- `locuslinkByID`: given one or more LocusIDs, the browser is opened at the URL corresponding to the first gene.

```
locuslinkByID("9203")  
http://www.ncbi.nlm.nih.gov/LocusLink/LocRpt.cgi?l=9203
```

- `locuslinkQuery`: given a search string, the results of the LocusLink query are displayed in the browser.

```
locuslinkQuery("zinc finger")  
http://www.ncbi.nlm.nih.gov/LocusLink/list.cgi?Q=zinc finger&ORG=Hs&V=0
```

- `getQuery4LL`.

annotate: querying PubMed
www.ncbi.nlm.nih.gov

- For any gene there is often a large amount of data available from PubMed.
- The `annotate` package provides the following tools for interacting with PubMed
 - `pubMedAbst`: a class structure for PubMed abstracts in R.
 - `pubmed`: the basic engine for talking to PubMed (`pmidQuery`).

annotate: pubMedAbst class

Class structure for storing and processing PubMed abstracts in R

- `pmid`
- `authors`
- `abstText`
- `articleTitle`
- `journal`
- `pubDate`
- `abstUrl`

annotate: high-level tools for querying PubMed

- `pm.getabst`: download the specified PubMed abstracts (stored in XML) and create a list of `pubMedAbst` objects.
- `pm.titles`: extract the titles from a list of PubMed abstracts.
- `pm.abstGrep`: regular expression matching on the abstracts.

annotate: PubMed example

```
pmid <-get("41046_s_at", env=hgu95aPMID)
pubmed(pmid, disp="browser")

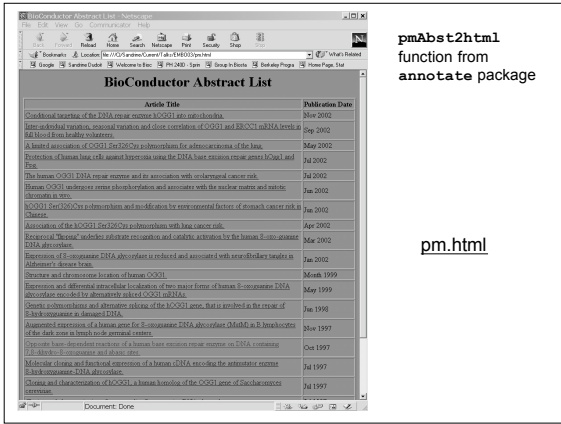
http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?tool=bioconductor&cmd=Retrieve&db=PubMed&list\_uids=10486218%2c9205841%2c8817323

absts <- pm.getabst("41046_s_at", base="hgu95a")
pm.titles(absts)
pm.abstGrep("retardation",absts[[1]])
```

annotate: PubMed HTML report

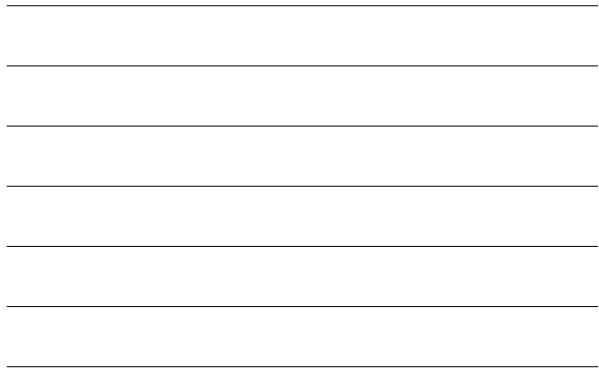
- The new function `pmAbst2HTML` takes a list of `pubMedAbst` objects and generates an HTML report with the titles of the abstracts and links to their full page on PubMed.

```
pmAbst2HTML(absts[[1]],
            filename="pm.html")
```



pmAbst2html
function from
annotate package

pm.html



annotate: analysis reports

- A simple interface, `ll.htmlpage`, can be used to generate an HTML report of analysis results.
- The page consists of a table with one row per gene, with links to LocusLink.
- Entries can include various gene identifiers and statistics.



BioConductor Gene Listing
Golub et al. data, genes with permutation maxT adjusted p-value < 0.01

LocusID	Gene name	Chromosome	ALL genes	ANL genes	Statistics	raw p-value	adj p-value
1311	M62996_at	7	0.391	1.58	0.024	0.245	0.245
1411	M62997_at	10	0.481	2.08	0.778	0.245	0.245
1511	M62998_at	13	0.488	1.34	4.03	0.245	0.00214
1602	M62999_at	8	0.384	1.1	0.96	0.245	0.00164
1611	M63000_at	11	0.162	1.36	17.97	0.245	0.245
1620	M63001_at	19	0.655	0.391	7.33	0.245	0.245
1629	M63002_at	1	0.667	0.655	7.42	0.245	0.00178
1638	M63003_at	3	1.84	0.343	7.33	0.245	0.001
1647	M63004_at	5	0.728	0.779	7.31	0.245	0.00114
1656	M63005_at	6	1.36	0.394	7.28	0.245	0.00116
1665	M63006_at	1	1.91	0.888	7.11	0.245	0.0017
1674	M63007_at	1	0.431	0.779	7.08	0.245	0.0018
1683	M63008_at	13	0.438	1.3	7.08	0.245	0.0018
1692	M63009_at	N/A	0.097	11.07	7.07	0.245	0.0019
1701	M63010_at	8	1.62	1.07	7.06	0.245	0.00186
1710	M63011_at	5	0.71	1.51	4.97	0.245	0.00132
1719	M63012_at	1	0.167	0.662	4.96	0.245	0.00138
1728	M63013_at	1	0.411	0.163	4.93	0.245	0.00138
1737	M63014_at	4	0.391	1.32	4.87	0.245	0.00138
1746	M63015_at	8	0.413	0.662	4.86	0.245	0.00138
1755	M63016_at	19	0.289	0.18	4.81	0.245	0.00134
1764	M63017_at	7	0.24	0.304	4.82	0.245	0.00136
1773	M63018_at	19	0.361	0.354	4.79	0.245	0.00131
1782	M63019_at	11	1.21	0.132	4.77	0.245	0.00141
1791	M63020_at	14	1.13	0.132	4.76	0.245	0.00132
1800	M63021_at	12	0.633	1.18	4.76	0.245	0.00132
1809	M63022_at	2	0.599	0.555	4.74	0.245	0.00137
1818	M63023_at	12	0.178	0.393	4.71	0.245	0.00144
1827	M63024_at	8	0.185	0.392	4.68	0.245	0.00142
1836	M63025_at	4	0.541	0.33	4.61	0.245	0.00142
1845	M63026_at	12	0.166	0.392	4.61	0.245	0.00142

ll.htmlpage
function from
annotate
package

genelist.html

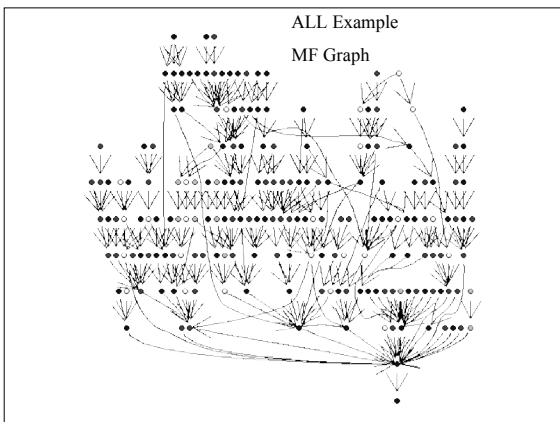
24

What is GO?

- The Gene Ontology Consortium coordinates the development and refinement of GO
- GO is a set of three ontologies for gene products
 - molecular function
 - cellular component
 - biological process

Data

- as part of Bioconductor we provide a GO package which has all the GO specific data
 - terms and relationships
 - some whole species data
- for each instrument (chip) we provide chip specific data
 - maps from the probes to GO terms
 - counts of probes per GO term + children
- constantly evolving and being updated



Demo
