For today...

- We will look at a breast cancer dataset, dowload from here: `http://www.biostat.jhsph.edu/~hcorrada/PASI_2010/chang03.rda`
- We will need a few more packages, you can install with biocLite now if you want to save time: hgu95av2.db, XML, annotate, KEGG.db, GO.db, annaffy
- We will also do a little bit of analysis of second-generation sequencing data, download a dataset from here: `http://www.biostat.jhsph.edu/~hcorrada/PASI_2010/seqdata.zip`
- We will also need a few more packages for sequence analysis: ShortRead, BSgenome.Scerevisiae.UCSC.sacCer1, yeast2probe

From CEL Files to Annotated Lists of Genes (Part 2)

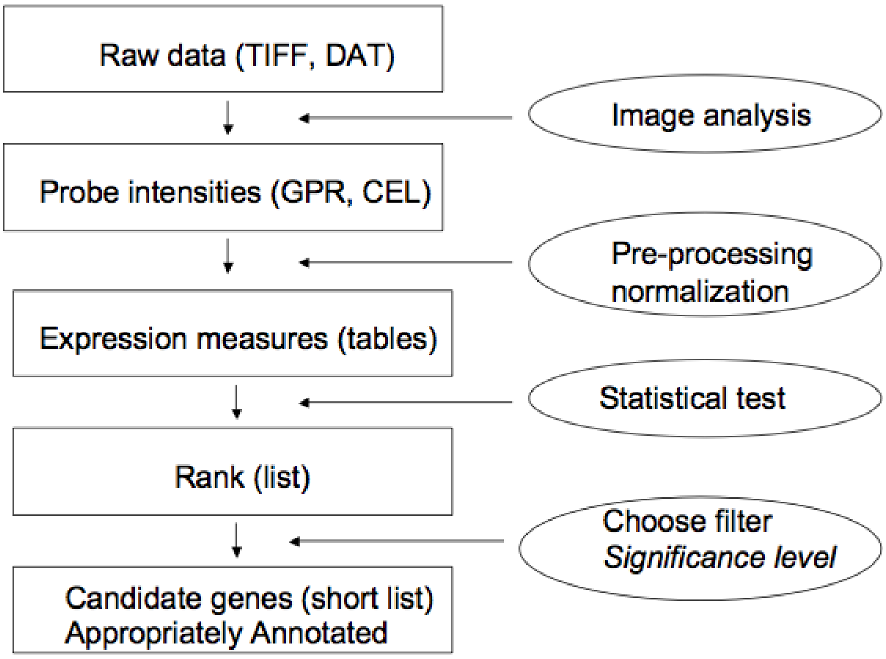
Héctor Corrada Bravo
based on slides developed by
Rafael A. Irizarry and Hao Wu

PASI, Guanajuato, México
May 3-4, 2010

Differential Expression and Annotation

Finding differentially expressed genes.

Workflow



Raw data (TIFF, DAT)

Probe intensities (GPR, CEL)

Expression measures (tables)

Rank (list)

Candidate genes (short list) Appropriately Annotated

Image analysis

Pre-processing normalization

Statistical test

Choose filter *Significance level*

A new dataset

- We will use a new dataset to continue our exercise. You can download the data here: http://www.biostat.jhsph.edu/~hcorrada/PASI_2010/chang03.rda
- The data for this experiment were obtained from: http://pierotti.group.ifom-ieo-campus.it/biocdb/data/experiment/ where you can find a large collection of free microarray data sets for breast and ovarian cancer.
- The data for this experiment have already been normalized for us.

Setup

- Let's start by loading packages

```
> library(Biobase)
> library(genefilter)
> library(affy)
```

- and the data

```
> load("chang03.rda")
```

Our new dataset

Let's get information about the experiment

```
> cat(abstract(experimentData(chang03)))
```

```
BACKGROUND: Systemic chemotherapy for operable breast cancer substantially
decreases the risk of death. Patients often have de novo resistance or incomplete
response to docetaxel, one of the most active agents in this disease. We
postulated that gene expression profiles of the primary breast cancer can predict
the response to docetaxel. METHODS: We took core biopsy samples from primary
breast tumours in 24 patients before treatment and then assessed tumour response
to neoadjuvant docetaxel (four cycles, 100 mg/m2 daily for 3 weeks) by cDNA
analysis of RNA extracted from biopsy samples using HgU95-Av2 GeneChip. FINDINGS:
From the core biopsy samples, we extracted sufficient total RNA (3-6 microg) for
cDNA array analysis using HgU95-Av2 GeneChip. Differential patterns of expression
of 92 genes correlated with docetaxel response (p=0.001). Sensitive tumours had
higher expression of genes involved in cell cycle, cytoskeleton, adhesion,
protein transport, protein modification, transcription, and stress or apoptosis;
whereas resistant tumours showed increased expression of some transcriptional and
signal transduction genes. In leave-one-out cross-validation analysis, ten of 11
sensitive tumours (90% specificity) and 11 of 13 resistant tumours (85%
```

Our new dataset

```
sensitivity) were correctly classified, with an accuracy of 88%. This 92-gene
predictor had positive and negative predictive values of 92% and 83%,
respectively. Correlation between RNA expression measured by the arrays and
semiquantitative RT-PCR was also ascertained, and our results were validated in
an independent set of six patients. INTERPRETATION: If validated, these molecular
profiles could allow development of a clinical test for docetaxel sensitivity,
thus reducing unnecessary treatment for women with breast cancer.
```

Our new dataset

- Find the dimensions of the expression data

```
> dim(exprs(chang03))
[1] 12625    24
```

- Find the dimensions of the measured covariates

```
> dim(pData(chang03))
[1] 24 15
```

- Look at the names of the measured covariates

```
> names(pData(chang03))
 [1] "Patient"
 [2] "disease.state"
 [3] "Tumour.type..IMC.invasive.mammary.carcinoma..IDC.invasive.d
 [4] "Age..years."
 [5] "Menopausal.status"
 [6] "Ethnic.origin"
 [7] "Bidimensional.tumour.size..cm."
 [8] "Clinical.axillary.nodes"
 [9] "Oestrogen..receptor.status"
[10] "Progesterone..receptor.status"
[11] "HER.2..immunhistochemical.analysis."
[12] "species"
[13] "tissue.type"
[14] "sample.type"
```

Our new dataset

- Make a table of the disease.state variable

```
> table(pData(chang03)$disease.state)

docetaxel resistant tumor
                       14
docetaxel sensitive tumor
                       10
```

- Look at disease state by progestorone receptor status

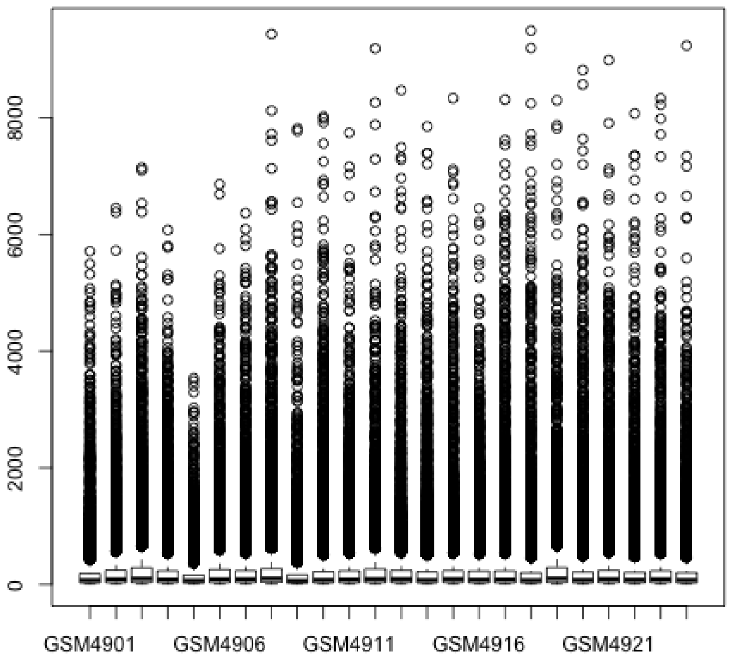
```
> table(pData(chang03)$disease.state,
+      pData(chang03)$Progesterone..receptor.status)

                           + -
  docetaxel resistant tumor 8 6
  docetaxel sensitive tumor 6 4
```

More Exploration

Let's do a boxplot of the expression measurements

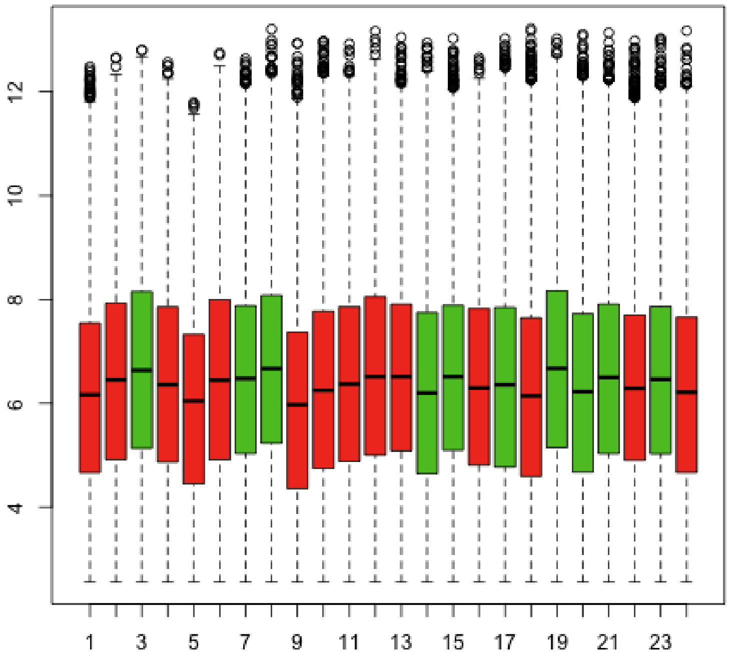
```
> boxplot(exprs(chang03))
```



More Exploration

Oh yeah, work in log space

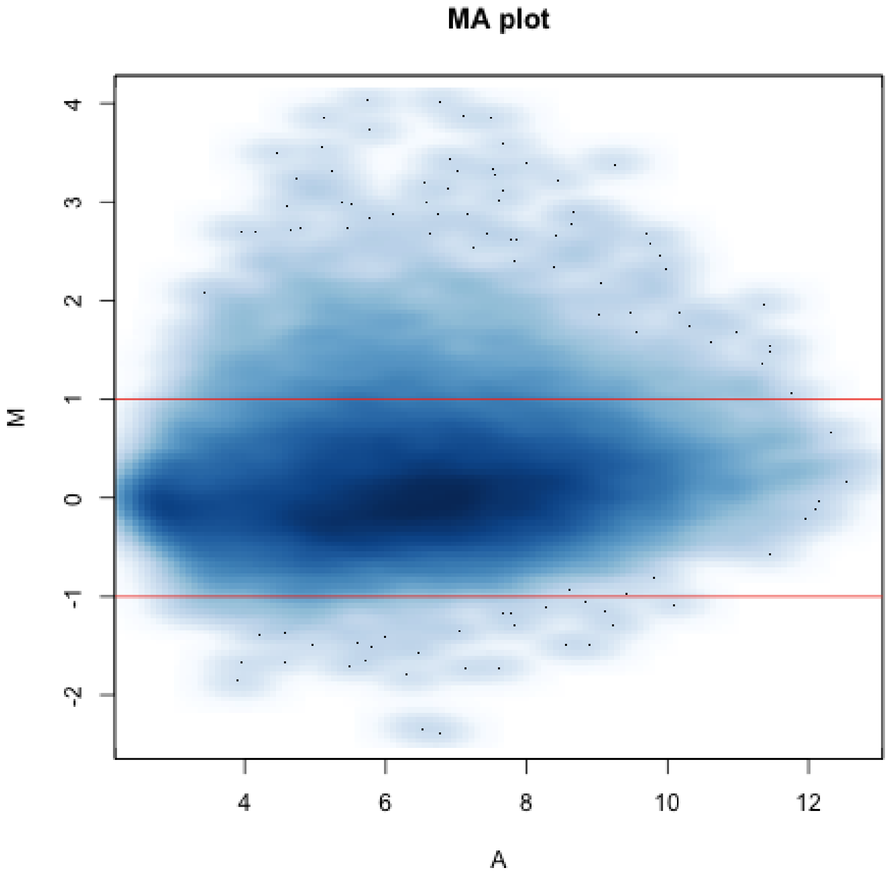
```
> y <- log2(exprs(chang03))
> boxplot(y ~ col(y), col = as.numeric(pData(chang03)$disease.state)
+      1)
```



More Exploration: MA plot

```
> Index <- as.numeric(pData(chang03)$disease.state)
> d <- rowMeans(y[, Index == 2]) -
+     rowMeans(y[, Index == 1])
> a <- rowMeans(y)
> smoothScatter(a, d, main = "MA plot",
+     xlab = "A", ylab = "M")
> abline(h = c(-1, 1), col = "red")
```

More Exploration: MA plot

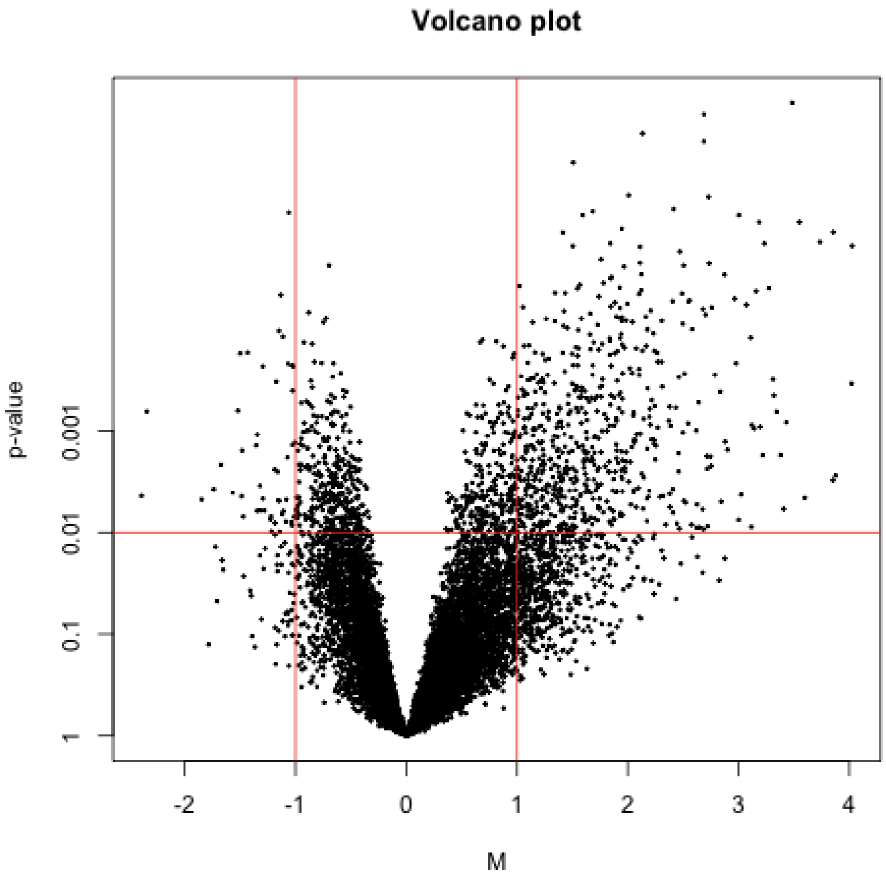


Differential Expression

Let's use limma (Empirical Bayes) again

```
> library(limma)
> design <- model.matrix(~factor(chang03$disease.state))
> fit <- lmFit(y, design)
> ebayes <- eBayes(fit)
```

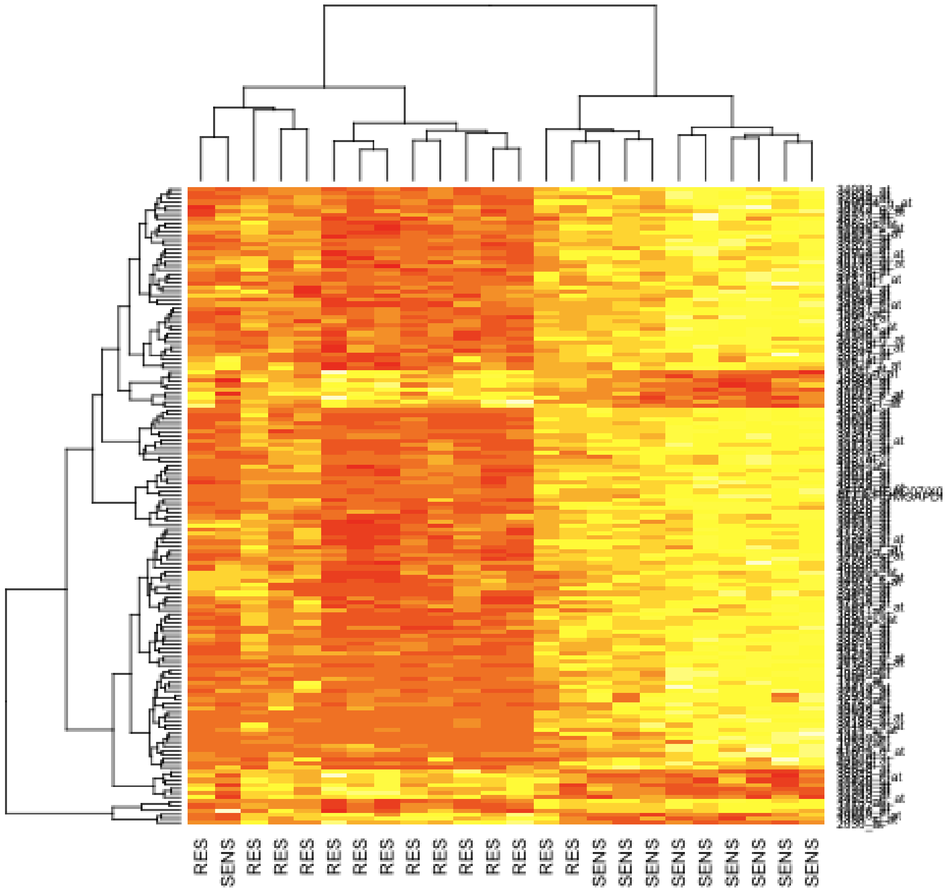
Volcano plot



Heatmap

```
> tab <- topTable(ebayes, coef = 2,
+     adjust = "fdr", n = 150)
> labCol <- c("RES", "SENS")[as.numeric(pData(chang03)$disease.state)
> heatmap(y[tab$ID, ], labCol = labCol)
```

Heatmap



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)

WWW Resources

- Nucleotide databases: e.g. GenBank
- Gene databases: e.g. Entrez Gene, UniGene
- Protein sequence and structure databases: e.g. SwissProt, Protein DataBank (PDB)
- Literature databases: 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 → Entrez Gene IDs
  - Affymetrix IDs → GenBank accession number
- Associate probes with biological data such as chromosomal position, pathways

annotate: matching IDs

Affy ID	41046_s_at
Entrez Gene ID	9203
GenBank accession #	X95808
Gene symbol	ZMYM3
PubMed ID	8817323, 8889548, 9205841
Chromosomal Location	X, Xq13.1

Annotation data packages

- Bioconductor provides annotation data packages that contain many different mappings to interesting data
  - Mappings between Affy IDs and other probe IDs: hgu95av2.db for HGU95Av2 GeneChip series, also, hgu133a.db, hu6800.db, etc.
  - Affy CDF data packages, e.g. hgu95av2cdf
  - Probe sequence data packages, e.g. hgu95av2probe
- These packages are updated and expanded regularly as new data becomes available
- They can be installed through biocLite()
- AnnBuilder provides tools for building annotation data packages

annotate: matching IDs

- Find out and load annotation package we need

```
> annotation(chang03)
```

```
[1] "hgu95av2"
```

```
> library(annotate)
> library(hgu95av2.db)
```

- Let's get matching IDs for the first 3 probesets on our list using the lookUp function

```
> probeset <- as.character(tab$ID[1:3])
```

```
> lookUp(probeset, "hgu95av2.db",
+      "ACCNUM")
```

annotate: matching IDs

```
$`36125_s_at`
[1] "L38696"

$`33781_s_at`
[1] "AF075599"

$`40549_at`
[1] "L04658"
```

```
> lookUp(probeset, "hgu95av2.db",
+      "SYMBOL")
```

```
$`36125_s_at`
[1] "RALY"

$`33781_s_at`
[1] "UBE2M"

$`40549_at`
[1] "CDK5"
```

annotate: matching IDs

```
> lookUp(probeset, "hgu95av2.db",
+      "GENENAME")
```

```
$`36125_s_at`
[1] "RNA binding protein, autoantigenic (hnRNP-associated with le

$`33781_s_at`
[1] "ubiquitin-conjugating enzyme E2M (UBC12 homolog, yeast)"

$`40549_at`
[1] "cyclin-dependent kinase 5"
```

```
> lookUp(probeset, "hgu95av2.db",
+      "UNIGENE")
```

annotate: matching IDs

```
$`36125_s_at`
[1] "Hs.136947"

$`33781_s_at`
[1] "Hs.406068"

$`40549_at`
[1] "Hs.647078"
```

```
> lookUp(probeset, "hgu95av2.db",
+      "CHR")
```

```
$`36125_s_at`
[1] "20"

$`33781_s_at`
[1] "19"

$`40549_at`
[1] "7"
```

annotate: matching IDs

```
> lookUp(probeset, "hgu95av2.db",
+      "CHRLOC")
```

```
$`36125_s_at`
      20
32581731

$`33781_s_at`
       19
-59067079

$`40549_at`
         7          7
-150750902 -150750898
```

```
> lookUp(probeset, "hgu95av2.db",
+      "MAP")
```

annotate: matching IDs

```
$`36125_s_at`
[1] "20q11.21-q11.23"

$`33781_s_at`
[1] "19q13.43"

$`40549_at`
[1] "7q36"
```

```
> sapply(lookUp(probeset, "hgu95av2.db",
+     "PMID"), head)
```

```
     36125_s_at 33781_s_at 40549_at
[1,] "7533788"  "9694792"  "1181841"
[2,] "8125298"  "10207026" "1330687"
[3,] "9373149"  "10722740" "1639063"
[4,] "9376072"  "10828074" "7566346"
[5,] "10500250" "12477932" "7834371"
[6,] "11780052" "12522145" "7949095"
```

annotate: matching IDs

For some common IDs, you can use more user-friendly functions provided by annotate

```
> getSYMBOL(probeset, "hgu95av2.db")
```

```
36125_s_at 33781_s_at   40549_at
    "RALY"     "UBE2M"     "CDK5"
```

```
> gg <- getGO(probeset, "hgu95av2.db")
> getGOdesc(gg[[1]][[1]]$GOID, "ANY")
```

```
$`GO:0006397`
GOID: GO:0006397
Term: mRNA processing
Ontology: BP
Definition: Any process involved in
    the conversion of a primary
    mRNA transcript into one or
    more mature mRNA(s) prior to
    translation into polypeptide.
Synonym: mRNA maturation
```

annotate

The annotate package provides tools for

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

annotate: querying PubMed

`http://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 articles 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 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

- Let's use all the genes on our list

```
> probenames <- as.character(tab$ID)
```

- Load the XML package, and get pubmed abstracts for the first 5 genes

```
> library(XML)
> absts <- pm.getabst(probenames[1:5],
+     "hgu95av2.db")
> absts[[1]][[1]]
```

```
An object of class 'pubMedAbst':
Title: Epstein-Barr virus-induced autoimmune responses. I.
     Immunoglobulin M autoantibodies to proteins mimicking and no
     mimicking Epstein-Barr virus nuclear antigen-1.
PMID: 7533788
Authors: JH Vaughan, JR Valbracht, MD Nguyen, HH Handley, RS Smit
     Patrick, GH Rhodes
Journal: J Clin Invest
Date: Mar 1995
```

annotate: PubMed example

- Let's look at the titles

```
> titl <- pm.titles(absts[1])
> strwrap(titl, simplify = FALSE)
```

```
[[1]]
[1] "c(\"Epstein-Barr virus-induced autoimmune responses. I. Immu
[2] "autoantibodies to proteins mimicking and not mimicking Epste
[3] "virus nuclear antigen-1.\", \"Oligo-capping: a simple method
[4] "the cap structure of eukaryotic mRNAs with oligoribonucleoti
[5] "\"Construction and characterization of a full length-enriche
[6] "5'-end-enriched cDNA library.\", \"The p542 gene encodes an a
[7] "that cross-reacts with EBNA-1 of the Epstein Barr virus and 
[8] "be a heterogeneous nuclear ribonucleoprotein.\","
```

annotate: PubMed HTML report

The 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")
> browseURL("pm.html")
```

annotate: analysis reports

- A simple interface, 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 Entrez Gene, Affymetrix, SwissProt, UniGene or OMIM
- Entries can include various gene identifiers and statistics

annotate: analysis reports

```
> ll <- getEG(probenames, "hgu95av2.db")
> sym <- getSYMBOL(probenames, "hgu95av2.db")
> tab <- data.frame(sym, tab[, -1])
> htmlpage(list(ll), filename = "report.html",
+     title = "HTML report", othernames = tab,
+     table.head = c("Entrez ID",
+         colnames(tab)), table.center = TRUE)
> browseURL("report.html")
```

annaffy

- Provides simplified mappings between Affymetrix IDs and annotation data
- Relies on chip-level annotation packages created by AnnBuilder
- Supplies functions to produce mappings for almost all environments in a given annotation package
- Primary function of annaffy is to produce very nice HTML or text tables containing
  - Links to databases
  - Statistics
  - Expression measures (color-coded to intensity for easy viewing)

annaffy

- Load some more annotation databases we will use

```
> library("KEGG.db")
> library("GO.db")
> library("annaffy")
```

- Make a table

```
> atab <- aafTableAnn(probenames,
+     "hgu95av2.db", aaf.handler())
```

- Save it as HTML

```
> saveHTML(atab, file = "report2.html")
> browseURL("report2.html")
```

Examining the R session

```
> sessionInfo()
```

```
R version 2.11.0 (2010-04-22)
x86_64-apple-darwin9.8.0

locale:
[1] en_US.utf-8/en_US.utf-8/C/C/en_US.utf-8/en_US.utf-8

attached base packages:
[1] stats     graphics  grDevices
[4] utils     datasets  methods
[7] base

other attached packages:
 [1] GO.db_2.4.1
 [2] hgu95av2.db_2.4.1
 [3] org.Hs.eg.db_2.4.1
 [4] RSQLite_0.8-4
 [5] DBI_0.2-5
```

Examining the R session

```
 [6] annotate_1.26.0
 [7] AnnotationDbi_1.10.0
 [8] affy_1.26.0
 [9] genefilter_1.30.0
[10] Biobase_2.8.0
[11] RColorBrewer_1.0-2

loaded via a namespace (and not attached):
[1] affyio_1.16.0
[2] preprocessCore_1.10.0
[3] splines_2.11.0
[4] survival_2.35-8
[5] xtable_1.5-6
```