Bioconductor

Course in Practical Microarray Analysis
Heidelberg 23.-27.9.2002

Slides ©2002 Sandrine Dudoit, Robert Gentleman.
Adapted by Wolfgang Huber.
Everywhere ...

- for statistical design and analysis:
  - technology development and validation, data preprocessing, estimation, testing, clustering, prediction, etc.

- for integration with biological information resources (in house and external databases)
  - gene annotation (Unigene, LocusLink)
  - graphical (pathways, chromosome maps)
  - patient data, tissue banks
Outline

- Overview of Bioconductor packages
  - Biobase
  - annotate
  - genefilter
  - marrayClasses, ...Input, ...Norm, ...Plots
  - Affy

- Dynamic statistical reports using Sweave: 'reproducible analyses'
Bioconductor

- Bioconductor is an open source project to design and provide high quality software and documentation for bioinformatics.
- Current focus: microarrays and gene (transcript) annotation
- Most of the early developments are in the form of R packages.
- Open to (your?) contributions
Bioconductor packages

- **General infrastructure**
  - Biobase
  - annotate, AnnBuilder
  - tkWidgets
- **Pre-processing for Affymetrix data**
  - affy
- **Pre-processing for cDNA data**
  - marrayClasses, marrayInput, marrayNorm, marrayPlots.
- **Differential expression**
  - edd, genefilter, multtest, ROC.
- etc.
Bioconductor training

• Extensive documentation and training materials for self-instruction and short courses
  - all available on WWW.

• R help system
  - interactive with browser or printable manuals;
  - detailed description of functions and examples;
  - E.g. help(maNorm), ? marrayLayout.

• R demo system
  - User-friendly interface for running demonstrations of R scripts.
  - E.g. demo(marrayPlots).
Biobase

contains class definitions and infrastructure classes:

- **phenoData**: sample covariate data (e.g. cell treatment, tissue origin, diagnosis)
- **miame** (minimal information about μarray experiments)
- **exprSet**: matrix of expression data, phenoData, miame, and other quantities of interest.
- **aggregate**: an infrastructure to put an aggregation procedure (cross-validation, bootstrap) on top of any analysis
objects of type exprSet allow subsetting w.r.t. genes (probes) and w.r.t. samples.

Expression values, gene and patient annotation are kept consistent under the subsetting

⇒ a frequent source of confusion or even 'bugs' is eliminated!
**genefilter**: separation of tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Programming pendant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define the filter criterion</td>
<td>A function that takes the data for one gene</td>
</tr>
<tr>
<td>Apply it to the data and obtain a selection</td>
<td>A logical vector</td>
</tr>
<tr>
<td>Apply the selection to the data</td>
<td>A new exprSet with the subset of interesting genes</td>
</tr>
</tbody>
</table>
genefilter: supplied filters

- **kOverA** - k samples with expression values larger than A.
- **gapFilter** - samples need to have a large IQR or a gap (jump).
- **ttest** - select genes according to t-test p-values using a covariate.
- **Anova** - select genes according to an Anova p-value.
- **coxfilter** - use Cox model p-values.
**genefilter: example**

Two filters: gene should be above “100” for 5 times and have a Cox-PH-model p-value <0.01

\[ kF \leftarrow kOverA(5, 100) \]
\[ cF \leftarrow \text{coxfilter}(\text{survtime}, \text{cens}, \text{p}=0.01) \]

Assemble them in a filtering function

\[ ff \leftarrow \text{filterfun}(kF, cF) \]

Apply the filter

\[ sel \leftarrow \text{genefilter}(\text{exprs(DATA)}, ff) \]

Select the relevant subset of the data

\[ 
\text{mySub} \leftarrow \text{DATA}[sel,] 
\]
Goal: associate experimental data with available meta data, e.g. gene annotation, literature.

Tasks:
- associate vendor identifiers (Affy, RZPD, ...) to other identifiers
- associate transcripts with biological data such as chromosomal position of the gene
- associate genes with published data (PubMed)
- produce nice-to-read tabular summaries of analyses.
For any gene there is often a large amount of data available from PubMed.

We have provided the following tools for interacting with PubMed.

- `pubMedAbst`: defines a class structure for PubMed abstracts in R.
- `pubmed`: the basic engine for talking to PubMed.

**WARNING**: be careful you can query them too much and be banned!
PubMed: high level tools

- `pm.getabst`: obtain (download) the specified PubMed abstracts (stored in XML).
- `pm.titles`: select the titles from a set of PubMed abstracts.
- `pm.abstGrep`: regular expression matching on the abstracts.
Data rendering

• A simple interface, ll.htmlpage, can be used to generate a webpage for your own use or to send to other scientists involved in the project.
# BioConductor Gene Listing

Top 100 genes ordered by t-test

Locus Link Genes

<table>
<thead>
<tr>
<th>locus link</th>
<th>Effect Size</th>
<th>p-value</th>
<th>adjusted p-value</th>
<th>v1</th>
<th>v2</th>
<th>v3</th>
<th>v4</th>
<th>nv1</th>
<th>nv2</th>
<th>nv3</th>
<th>nv4</th>
</tr>
</thead>
<tbody>
<tr>
<td>NA</td>
<td>0.032</td>
<td>1.2e-10</td>
<td>1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>NA</td>
<td>0.032</td>
<td>1.2e-10</td>
<td>1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>81822</td>
<td>0.032</td>
<td>1.2e-10</td>
<td>1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>50683</td>
<td>0.032</td>
<td>1.2e-10</td>
<td>1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>NA</td>
<td>0.032</td>
<td>1.2e-10</td>
<td>1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>NA</td>
<td>0.0313</td>
<td>1.3e-10</td>
<td>1.1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>25748</td>
<td>0.0313</td>
<td>1.3e-10</td>
<td>1.1e-06</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>0.032</td>
<td>3.3e-08</td>
<td>0.00029</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>24472</td>
<td>-1.48</td>
<td>1.4e-06</td>
<td>0.012</td>
<td>12</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>14</td>
<td>14</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>NA</td>
<td>0.0308</td>
<td>5.2e-06</td>
<td>0.045</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>12</td>
<td>0.0283</td>
<td>5.6e-06</td>
<td>0.048</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>NA</td>
<td>0.0279</td>
<td>6.9e-06</td>
<td>0.059</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>24472</td>
<td>-1.2</td>
<td>7.9e-05</td>
<td>0.067</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>12</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>19</td>
<td>-0.92</td>
<td>1.1e-05</td>
<td>0.091</td>
<td>11</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>8</td>
<td>1.26</td>
<td>1.1e-05</td>
<td>0.094</td>
<td>9.8</td>
<td>9.9</td>
<td>14</td>
<td>15</td>
<td>9.6</td>
<td>4.9</td>
<td>9.2</td>
<td></td>
</tr>
<tr>
<td>NA</td>
<td>0.895</td>
<td>1.6e-05</td>
<td>0.13</td>
<td>12</td>
<td>12</td>
<td>15</td>
<td>15</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>50260</td>
<td>0.298</td>
<td>1.3e-05</td>
<td>0.14</td>
<td>10</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
</tbody>
</table>
Data packages

The Bioconductor project is starting to develop and deploy packages that contain only data. Available: Affymetrix hu6800, hgu95a, hgu133a, mgu74a, rgu34a, KEGG, GO

These packages contain many different mappings between relevant data, e.g.

KEGG: EnzymeID – GO Category
hgu95a: Affy Probe set ID - EnzymeID

Update: simply by R function update.packages()
dataset: hgu95a

maps to LocusLink, GenBank, gene symbol, gene Name.
chromosomal location, orientation.
maps to KEGG pathways, to enzymes.

data packages will be updated and expanded regularly as new or updated data become available.
Diagnostic plots and normalization for cDNA microarrays
(S Dudoit, Y Yang, T Speed, et al)

- **marrayClasses**: class definitions for microarray data objects and basic methods

- **marrayInput**: reading in intensity data and textual data describing probes and targets;
  automatic generation of microarray data objects;
  widgets for point & click interface.

- **marrayPlots**: diagnostic plots.

- **marrayNorm**: robust adaptive location and scale normalization procedures.
marrayPlots package

- **maImage**: 2D spatial images of microarray spot statistics.
- **maBoxplot**: boxplots of microarray spot statistics, stratified by layout parameters.
- **maPlot**: scatter-plots of microarray spot statistics, with fitted curves and text highlighted, e.g., MA-plots with loess fits by sector.
- See demo(marrayPlots).
demo(marrayPlots)

Swirl array 93: image of Cy3 background intensities
Swirl array 93: pre-normalization MA-plot, lowess fits within print-tip-group.
marrayNorm package

robust adaptive location and scale normalization for a batch of arrays

- intensity or A-dependent location normalization (maNormLoess);
- 2D spatial location normalization (maNorm2D);
- median location normalization (maNormMed);
- scale normalization using MAD (maNormMAD);
- composite normalization.
marrayInput package

• Start from
  - image quantitation data, i.e., output files from image analysis software, e.g., .gpr for GenePix or .spot for Spot.
  - Textual description of probe sequences and target samples, e.g., gal files, god lists.

• read.marrayLayout, read.marrayInfo, and read.marrayRaw: read microarray data into R and create microarray objects of class marrayLayout, marrayInfo, and marrayRaw, resp.
Multiple hypothesis testing

- Bioconductor R `multtest` package
- Multiple testing procedures for controlling
  - FWER: Bonferroni, Holm (1979), Hochberg (1986), Westfall & Young (1993) maxT and minP.
- Tests based on $t$- or $F$-statistics for one- and two-factor designs.
- Permutation procedures for estimating adjusted $p$-values.
- Documentation: tutorial on multiple testing.
Sweave

• The Sweave framework allows dynamic generation of statistical documents intermixing documentation text, code and code output (textual and graphical).

• Fritz Leisch’s Sweave function from R tools package.

• See ? Sweave and manual http://www.ci.tuwien.ac.at/~leisch/Sweave
Sweave input

Source: a text file which consists of a sequence of documentation and code segments ('chunks')

- **Documentation chunks**
  - start with @
  - can be text in a markup language like LaTeX.

- **Code chunks**
  - start with <<name>>=
  - can be R or S-Plus code.

Sweave output

After running Sweave and Latex, obtain a single document, e.g. pdf file containing
- the documentation text
- the R code
- the code output: text and graphs.

The document can be automatically regenerated whenever the data, code or text change.

Ideal medium for the communication of data analyses that want to be reproducible by other researchers: they can read the document and at the same time have the code chunks executed by their computer!