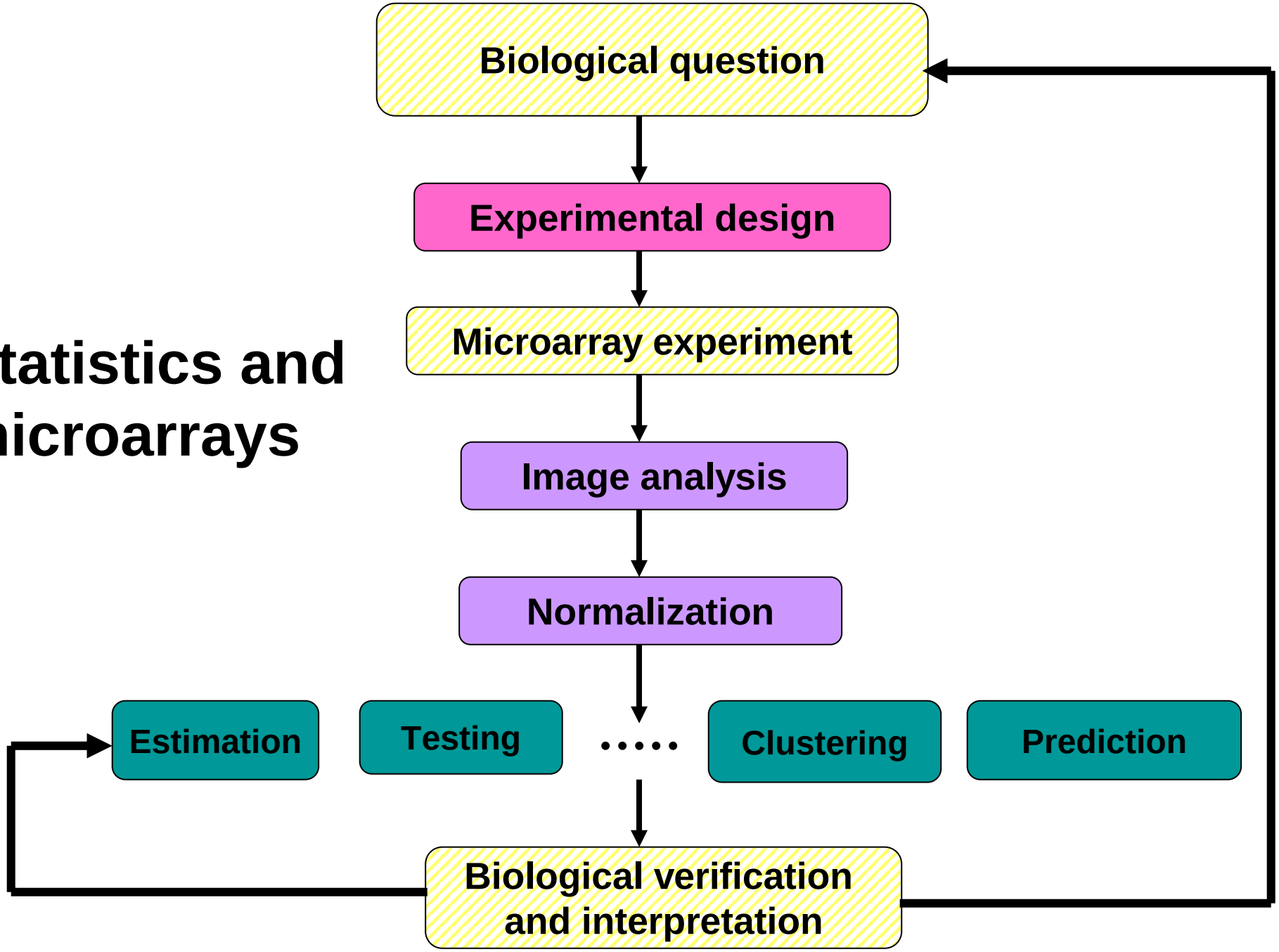
Overview of the Bioconductor project and marray packages

Sandrine Dudoit

PH296, Section 36

May 6, 2002

Statistics and microarrays



Statistical computing

Everywhere …

- for statistical design and analysis:
  - pre-processing, estimation, testing, clustering, prediction, etc.
- for integration with biological information resources (in house and external databases)
  - gene annotation (GenBank, LocusLink);
  - literature (PubMed);
  - graphical (pathways, chromosome maps).

BioConductor: software for bioinformatics

http://www.bioconductor.org

Bioconductor project

- **Goal.** To develop a statistical software infrastructure which promotes the rapid deployment of extensible, scalable, and interoperable software for the analysis and comprehension of biomedical and genomic data.

- **Developers.** About 20 core members, international collaboration.

- **Model.** Open source and open development (GPL, LGPL).

Bioconductor project

- Use of the R language and environment for statistical computing and graphics
  - Open source, GNU's S-Plus.
  - Full-featured programming language
  - Extensive software repository for statistical methodology: linear and non-linear modeling, testing, classification, clustering, resampling, etc.
  - Design-by-contract principle: package system.
  - Extensible, scalable, interoperable.
  - Unix, Linux, Windows, and Mac OS.

Bioconductor project

- Integrated data analysis of large and complex datasets from varied sources:
  - transcript levels from microarray experiments;
  - covariates: treatment, dose, time;
  - clinical outcomes: survival, tumor class;
  - textual data (PubMed abstracts);
  - gene annotation data (GenBank, LocusLink);
  - graphical data (pathways, chromosome maps);
  - sequence data;
  - copy number (CGH);
  - etc.

Bioconductor project

- Object-oriented class/method design: efficient representation and manipulation of large and complex biological datasets of multiple types.

- Widgets: Specific, small scale, interactive components providing graphically driven analyses - point & click interface.

Bioconductor project

- Interactive tools for linking experimental results to annotation/literature WWW resources in real time. E.g. PubMed, GenBank, LocusLink.

- Scenario. For a list of differentially expressed genes obtained from **`multtest`**, use **`annotate`** package to generate an HTML report with links to LocusLink for each gene.

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).`**

Bioconductor training

- R vignettes system:
  - comprehensive repository of step-by-step tutorials covering a wide variety of computational objectives in `/doc` subdirectory;
  - Use `Sweave` function from `tools` package.
  - integrated statistical documents intermixing text, code, and code output (textual and graphical);
  - documents can be automatically updated if either data or analyses are changed.

- Modular training segments:
  - short courses: lectures and computer labs;
  - interactive learning and experimentation with the software platform and statistical methodology.

Diagnostic plots and normalization for cDNA microarrays

- **`marrayClasses`**:
  - class definitions for microarray data objects;
  - basic methods for manipulation of microarray objects.
- **`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.

Classes and methods

- Object-oriented programming in R: John Chamber's **`methods`** package.
- Classes reflect how we think of certain objects and what information these objects should contain.
- Classes are defined in terms of slots which contain the relevant data
- Methods define how a particular function should behave depending on the class of its arguments and allow computations to be adapted to particular classes.

`marrayClasses` package

- See Minimum Information About a Microarray Experiment -- MIAME document.
- Microarray classes should represent
  - gene expression measurements, for example,
    - scanned images, i.e., raw data;
    - image quantitation data, i.e., output from image analysis;
    - normalized expression levels, i.e., log-ratios M.
  - reliability information of these measurements;
  - information on the probe sequences spotted on the arrays;
  - information on the target samples hybridized to the arrays.

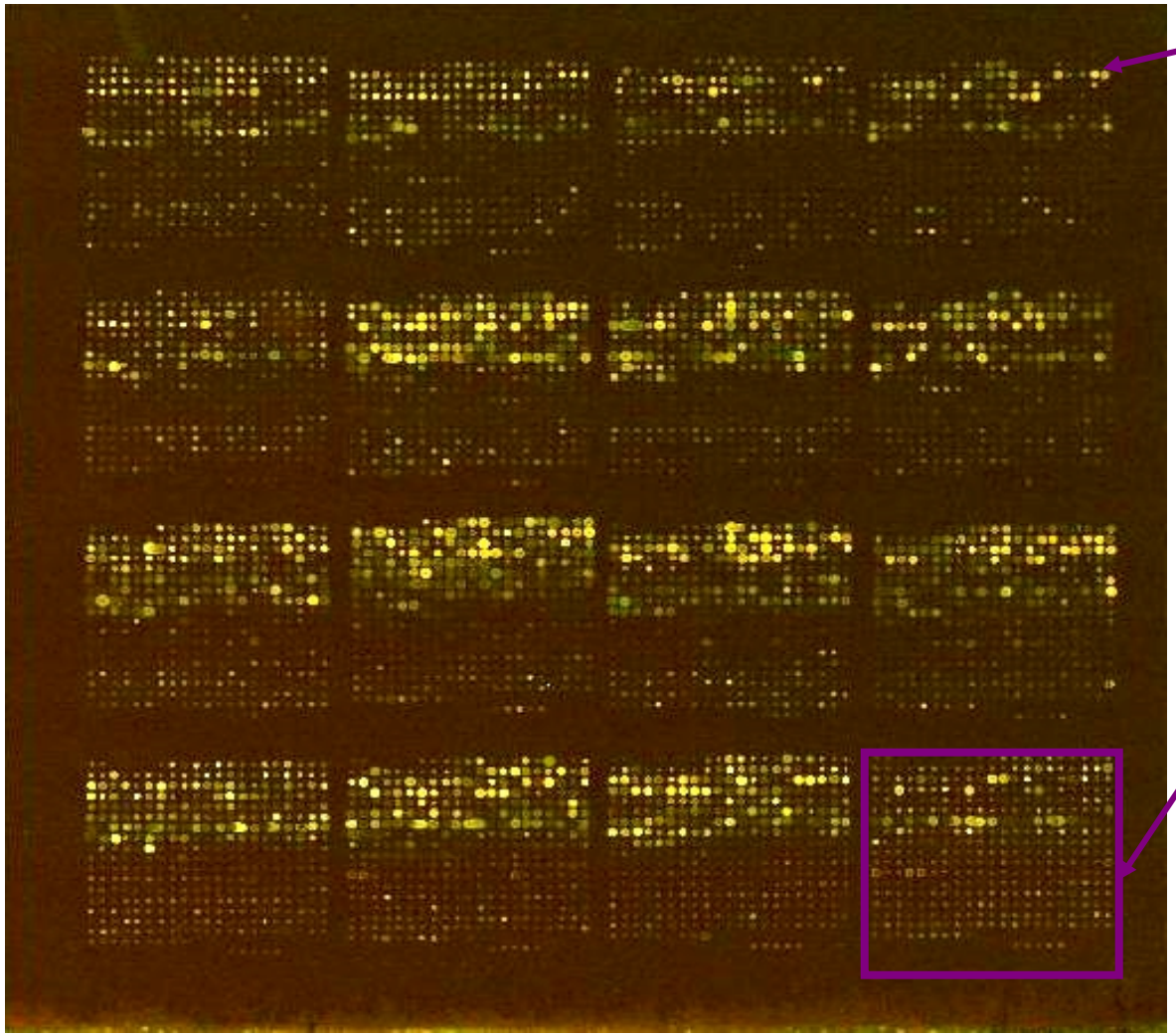
Layout terminology

- Target**:** DNA hybridized to the array, mobile substrate.
- Probe**:** DNA spotted on the array,

  aka. spot, immobile substrate.
- Sector**:** collection of spots printed using the same print-tip (or pin),

  aka. print-tip-group, pin-group, spot matrix, grid.
- The terms slide and array are often used to refer to the printed microarray.
- Batch: collection of microarrays with the same probe layout.
- Cy3 = Cyanine 3 = green dye.
- Cy5 = Cyanine 5 = red dye.

Layout terminology



Probe

4 x 4 sectors
19 x 21 probes/sector
6,384 probes/array

Sector

marrayLayout class

Array layout parameters

maNspots	Total number of spots
maNgr, maNgc	Dimensions of grid matrix
maNsr, maNsc	Dimensions of spot matrices
maSub	Current subset of spots
maPlate	Plate IDs for each spot
maControls	Control status labels for each spot
maNotes	Any notes

`marrayInfo` class

Descriptions of probe sequences or target mRNA samples

maLabels	Vector of probe or array labels
maInfo	Data frame of probe or target sample descriptions
maNotes	Any notes

Not microarray specific

`marrayRaw` class

Pre-normalization intensity data

maRf	maGf	Matrix of red and green foreground intensities
maRb	maGb	Matrix of red and green background intensities
maW		Matrix of spot quality weights
maLayout		Array layout parameters -- `marrayLayout`
maGnames		Description of spotted probe sequences -- `marrayInfo`
maTargets		Description of target samples -- `marrayInfo`
maNotes		Any notes

`marrayNorm` class

Post-normalization intensity data

maA		Matrix of average log-intensities
maM		Matrix of normalized intensity log-ratios
maMloc	maMscale	Matrix of location and scale normalization values
maW		Matrix of spot quality weights
maLayout		Array layout parameters -- `marrayLayout`
maGnames		Description of spotted probe sequences -- `marrayInfo`
maTargets		Description of target samples -- `marrayInfo`
maNormCall		Function call
maNotes		Any notes

`marrayClasses` package

- Useful methods for microarray classes include
- Accessor methods, for accessing slots of microarray objects.
- Assignment methods, for replacing slots of microarray objects.
- Printing methods, for summaries of intensity statistics and probe and target information.
- Subsetting methods, for accessing subsets of spots and/or arrays.
- Coercing methods, for conversion between classes.

`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)`.

`marrayNorm` package

- **`maNormMain`**: main normalization function, allows 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.
- **`maNorm`**: simple wrapper function. **`maNormScale`**: simple wrapper function for scale normalization.

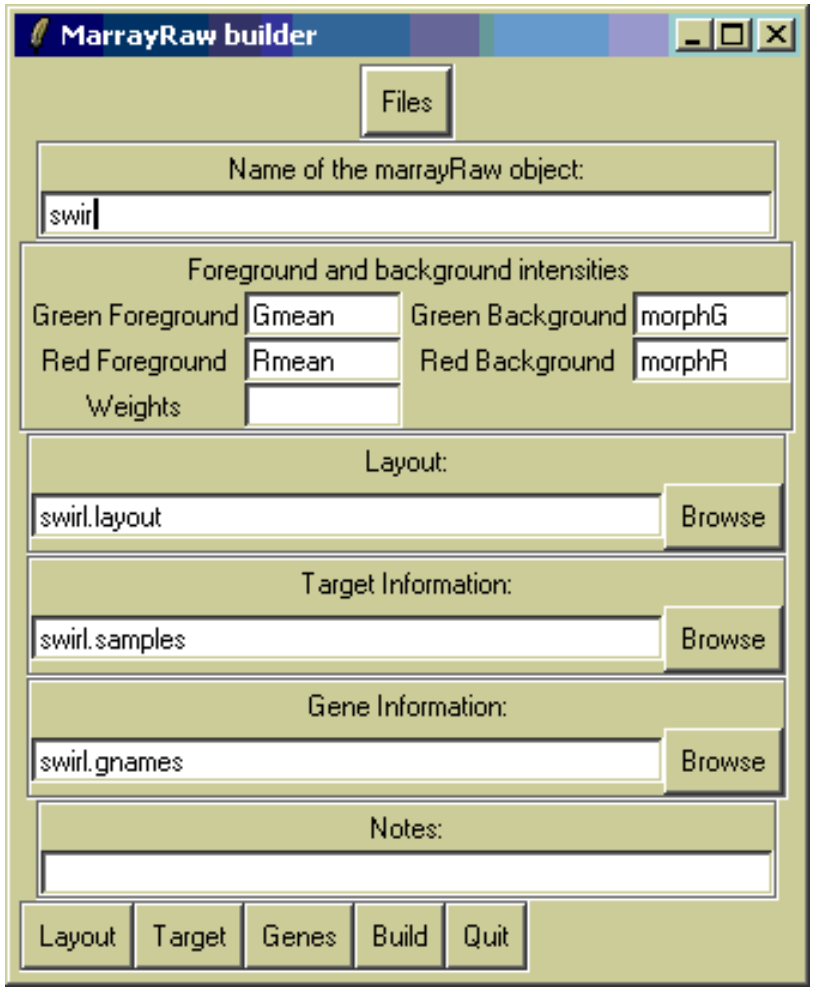
`marrayInput` package

- 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.

`marrayInput` package

- Widgets for graphical interface: **`widget.marrayLayout`**, **`widget.marrayInfo`**, **`widget.marrayRaw`**.



Multiple hypothesis testing

- Bioconductor R **`multtest`** package
- Multiple testing procedures for controlling
  - FWER: Bonferroni, Holm (1979), Hochberg (1986), Westfall & Young (1993) maxT and minP.
  - FDR: Benjamini & Hochberg (1995), Benjamini & Yekutieli (2001).
- 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/

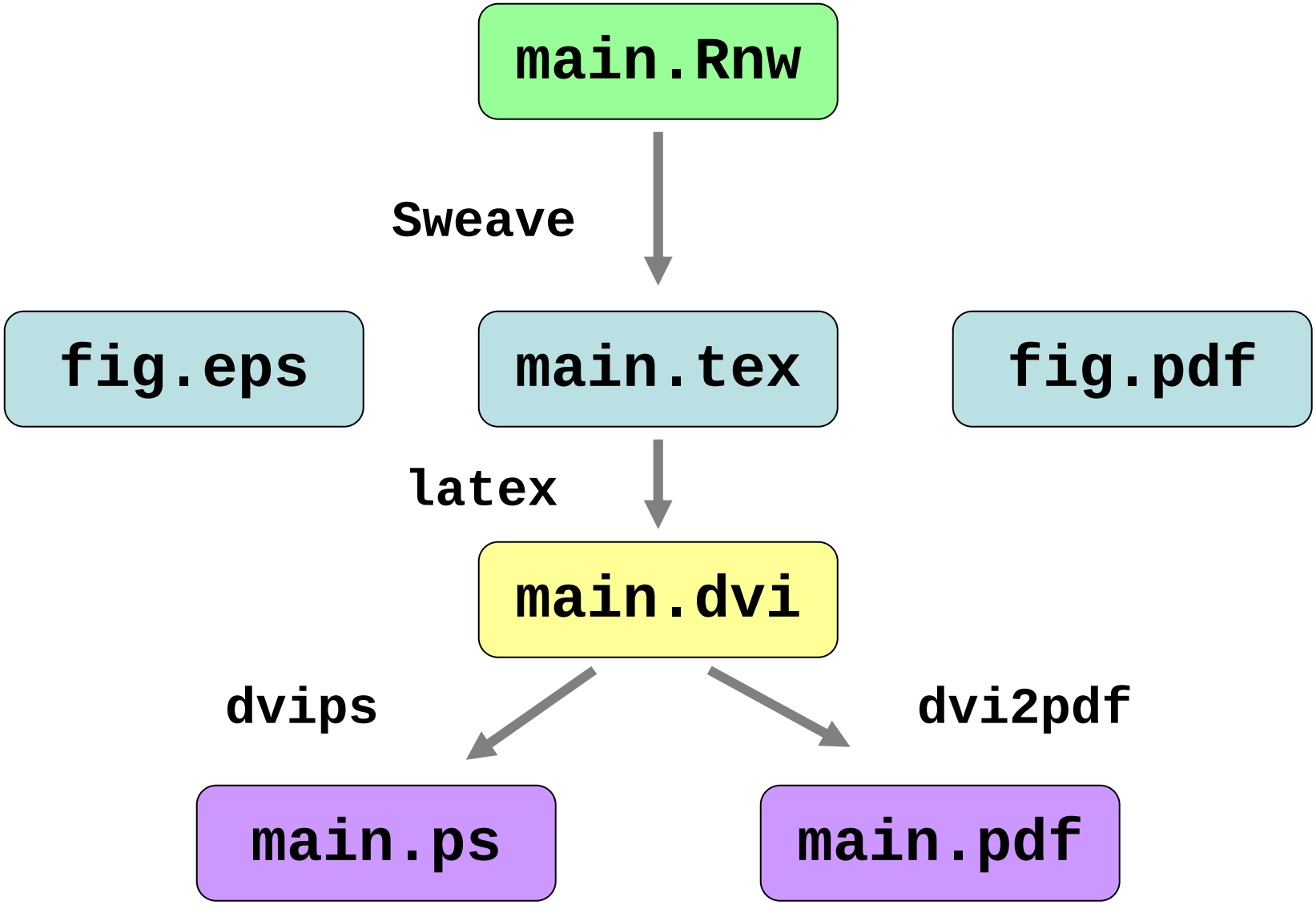
Sweave input

- Source: a noweb file, i.e., a text file which consists of a sequence of code and documentation segments or 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.
  - File extension: **`.rnw`**, **`.Rnw`**, **`.snw`**, **`.Snw`**.

Sweave output

- Output: **`Sweave`** produces a single document, e.g., **`.tex`** file, or **`.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.
- **`Stangle`**: extract only the code.

Sweave


main.Rnw
Sweave
fig.eps
main.tex
fig.pdf
latex
main.dvi
dvips
dvi2pdf
main.ps
main.pdf

Acknowledgements

- **Bioconductor core team**
- **Robert Gentleman**, Biostatistics, Harvard
- **Yongchao Ge**, Statistics, UC Berkeley
- **Yee Hwa (Jean) Yang**, Statistics, UC Berkeley

References

- **R** http://www.r-project.org
  - Software; Documentation; R Newsletter.

- **Bioconductor** http://www.bioconductor.org
  - Software; Documentation; Training materials from workshops; Mailing list.

- **Personal** http://www.stat.berkeley.edu/~sandrine
  - Articles and tech. reports on: image analysis; normalization; identification of differentially expressed genes; cluster analysis; classification.