Introduction to R

For MSA100, autumn 2008

Petter Mostad

mostad@chalmers.se

What is R?

- A programming language for statistical data analysis called S developed at Bell Labs. Later extended to S+.
- R is a free, open source version of S/S+.
- Available from www.r-project.org
- Under active cooperative development; versions change frequently.
- Very popular tool in statistics community: New methods often now implemented there.
- Widely used at this department

Extended with packages

- Examples: LearnBayes, multtest,...
- Reasonably easy to implement your own method into a package
- Sharing and exchange of packages facilitated by websites.
- Example: Bioconductor, with packages covering many areas of bioinformatics, e.g. microarrays.

Comparisons with other languages

- Quite similar to matlab
- More ”higher level” commands than C/C++, Java, etc.
- The open-source cooperative development means it is a bit unwieldy conceptually
- Very slow when using loops; use matrices
- Frequent new releases

Basics of R

- Command-driven language
- Data and functions stored as named objects
- Objects can be fairly simple (vectors, matrices) or more comples (assembled from other objects)
- Runs under for example Linux or Windows (Windows interface is more graphical)

Vectors and matrices

- Calculations often done on vectors or matrices
- Elementwise operations
- Subsetting, indexing
- Logical indexing

Help and documentation

- Use ”An introduction to R” at R website.
- help(<name of object>)
- help(package=<packagename>)
- help.search(”<subject>”)
- help.start()

Organizing your computations

- R has a ”current directory”
- Your objects are contained in your ”current workspace”, which can be saved any time
- Keep separate projects in separate workspaces/directories
- Keep it tidy!

Graphical visualization

- A ”generic” function: plot()
- High level commands, like pairs(), image(), contour()...
- Lower level commands, adding stuff: points(), lines(), text(), title(), legend()...
- plotting characters pch, colors col...

Functions

- Most of R consists of functions
- The arguments can either be input in the right order, or using argument names
- Use help(...) !

Writing your own functions

- Collecting commands into a function
- Arguments to function
- Returning result as a list
- Assignments within functions
- programming: conditional statements, loops (avoid them!) etc...
- fix(<myfunction>)

Import and export of data

- Useful functions: read.table, write.table
- Works with mixed-type data (numbers and text columns)
- Works well with tab-separated data
- scan()

R packages

- A package: collection of functions (and data) concerning special application
- Contributed from different sources/persons
- Can often be downloaded from CRAN
- must be ”loaded” with library()
- search()
- help(package = graphics)
- Also: Documentation from help.start()