Ice Break of R
An Introduction to the R Programming Language

Jinlong Zhang

Kadoorie Farm & Botanic Garden

November 25, 2011
Table of Contents

1 Introduction
   - History
   - Advantages & Disadvantages

2 R Basics
   - CRAN & Packages
   - Objects

3 R graphics
   - Plots Examples
   - Create a scatter plot step by step

4 Programming
   - Scripting

5 Further Reading
What is R?

R is an open source software for statistical computing. The commands that driving R is also called R programming language.
R is a dialect of S language, a statistical programming language developed by Dr. John Chambers at A&T lab in the 1970s.

S+ is an software that implementing S, and many additional features. Before the launch of R, S+ has been widely used in statistics and modeling.

The first version of R was developed by Robert Gentleman and Ross Ihaka, at the University of Auckland around 1997, for teaching S+.
The Characteristics of R

- **Open Source**
  The source code of R program and the extensions could be examined line by line.

- **Integrating with other Programming Language**
  R is an interpreting language, can be rather slow, but could integrate with high efficient languages such as C, C++ or Fortran

- **OS independence**
  UNIX, Linux, Windows, MacOS, FreeBSD...

- **Command line Driven**
  You have to write Commands...
Why R?

- The most extensive modeling resources in scientific research
- The fine publishing quality graphs
- Easy to develop your own model
- R is free, GNU
The Disadvantages

- **R is slow**
  R is an interpreting language and is not very fast. Could be 1/40 of C.

- **Limitation of Memory**
  All the objects are in memory.

- **R is hard to learn**
  One has to memorize the commands/functions, and understand the logics of programming.
  The fluency in R requires great time and energy.
88 Sites in 36 Regions
The software and packages are deposited on CRAN mirrors
First Step

R could be downloaded at
http://www.r-project.org/
at any CRAN mirror around the world.
There are 87 Sites in 36 Regions, which are called "CRAN", and currently hosting R and the 3400 packages.
R core Team provides a simple GUI for R.
Other GUls

Rstudio, available on Linux, Windows, MacOS
A Package is a collection of R functions with comprehensive documents. A Package includes: R functions, Data Example, Help Files, Namespace and Description. The default installation is kept as minimum. The function of R could be extent by loading R packages.

- Phylogenetics
- Multivariate
- Bayesian statistics
The Packages I

**ade4**  Analysis of Ecological Data using Euclidean methods
**ape**  Analysis of Ecology and Evolution
**boot**  Bootstrap
**cluster**  Cluster Analysis
**ecodist**  Ecological Distance
**e1071**  Misc Functions of the Dept. Stat. TU Wien
**MASS**  Venables and Ripley’s MASS
**lattice**  Lattice Graphics
The Packages II

maptools  Tools for reading and handling spatial objects
mvpart   Multivariate partitioning
nlme     Linear and Nonlinear Mixed Effects Models
ouch     Ornstein-Uhlenbeck models
raster   Geographic analysis and modeling with raster data
sp       classes and methods for spatial data
spatstat Spatial Point Pattern analysis, model-fitting
vegan    Community Ecology Package
The packages are reviewed periodically by profounding experts, the results are appeared online as CRAN TaskViews.
vegan: Community Ecology Package

Ordination methods, diversity analysis and other functions for community and vegetation ecologists.

Version: 1.17-2
Suggests: MASS, mgcv, lattice, cluster, scatterplot3d, rgl, tcltk
Published: 2010-03-08
Author: Jari Oksanen, F. Guillaume Blanchet, Roeland Kindt, Pierre Legendre, R. B. O'Hara, Gavin L. Simpson, Peter Solymos, M. Henry H. Stevens, Helene Wagner
Maintainer: Jari Oksanen <jari.oksanen at oulu.fi>
License: GPL-2
URL: http://vegan.r-forge.r-project.org/
In views: Environmetrics, Multivariate, Phylogenetics, Psychometrics, Spatial
CRAN checks: vegan results

Downloads:
Package source: vegan_1.17-2.tar.gz
MacOS X binary: vegan_1.17-2.tgz
Windows binary: vegan_1.17-2.zip
Reference manual: vegan.pdf

Description and the checking results of packages could be found at CRAN Mirror.
Install Packages

To install a package, just type

\begin{verbatim}
install.packages("vegan")
\end{verbatim}

Or,

Packages > install packages from local files
Use a Package

Packages must be loaded into memory before the functions could be called.

R code

```r
library("vegan")
```

There are comprehensive documentations for R functions, to look at the help files, just type, for example:

R code

```r
?vegan
help("vegan")
help.search("t.test")
```

or RGui>Help>Html help
Fitting Linear Models

Description

`lm` is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although `aov` may provide a more convenient interface for these).

Usage

```r
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Arguments

- `formula` an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under ‘Details’.
- `data` an optional data frame, list or environment (or object coercible by `as.data.frame`) containing the variables in the model. If not found in `data`, the variables are taken from environment(`formula`), typically the environment from which `lm` is called.
- `subset` an optional vector specifying a subset of observations to be used in the fitting process.
Contents of Help Files

lm{stats}
Fitting Linear Models
Description
Title of the function
The description
Usage
Parameters of the function
Arguments
Parameters in detail
Details
How the algorithms implemented
Author(s)
Author of the function
References
Literature Cited
Examples
Example of the function
R as a calculator

You can directly input R commands:

R code

2 + 2
a <- 2

Assignment
<-, =, or even ->

R code

b <- 2
c <- a+b
c

# Comments

Jinlong Zhang (Kadoorie Farm & Botanic Garden)
Objects in R

- **Type of elements**
  - **Numeric** 100, 0, -4.335
  - **Character** "China"
  - **Logical** TRUE, FALSE
  - **Factor** Different levels
  - **Complex** 2 + 3i

- **Type of Objects**
  - **Vector** `c(1,2,3); c("a","a","b","b","c")`
  - **Matrix** `x <- 1:12; dim(x) <- c(3,4)`
  - **Dataframe** Combined vectors of different types
  - **List** Container of Different objects
  - **Array** Matrix with different dimensions
Operators

Math
+ , −, *, /, ∧

Comparisons
>, <, <=, >=, ==, !=

Logical
!, &, &&, |, ||
Creating Vectors

Character

character <- c("China", "Korea", "Japan", "UK", "USA", "France", "India", "Russia")

Numerical

numeric <- c(1, 3, 6, 7, 3, 8, 6, 4)

Logical

logical <- c(T, F, T, F, T, F, F, T)

Function to Create Vectors

c(2, 5, 6, 9)
rep(2, times=4)
seq(from=3, to=21, by=3)
1:15
Graphics generated by R

The R basic graphs
Ancestral Reconstruction using R
Handling: SHP files, Conversion between Projections, Raster data, extract values, Krigging, Spatial Autoregressive Models
Graphics generated by R

Krigging and spatial analysis
Graphics generated by R

Results of spatial analysis
Higher Plotting Functions

- `plot()`  
  Scatter plot
- `hist()`  
  Histograms
- `boxplot()`  
  Boxplot
- `stripchart()`  
  stripchart
- `barplot()`  
  barplot
- `piechart()`  
  Pies
- `legend()`  
  Add legends
Lower Plotting Functions

- `lines()`: add line to plot
- `curve()`: add curve
- `abline()`: add straight line
- `points()`: add points
- `segments()`: Add segments
- `axis()`: add axis
- `box()`: Add box to the plot
- `title()`: Add title
- `text()`: Add text to plot
Creating a scatter plot

Original Data

```r
x <- runif(50,0,2)
y <- runif(50,0,2)
```

Step 1

```r
plot(x, y, type="n", xlab="", ylab="", axes=F)
```

Step 2

```r
points(x,y)
```
Creating a scatter plot II

Step 3
axis(1);
axis(at=seq(0.2,1.8,0.2), side=2)

Step 4
box()

Step 5
title(main="Main title", sub="subtitle", xlab="x-label", ylab="y-label")
Creating a scatter plot III

A plot step by step
Scripting

Save the R code in a plain text editor, with an extension .r
The code could be implemented later.
Code Highlighting

```r
data2mat <- function(data = data)
{
  if (!any(colnames(data) == "abundance"))
    stop("A column named "abundance" must be specified.")
  if (!any(is.integer(data$abundance)))
    stop("Number of individuals must be integer!")
  col <- which(colnames(data) == "abundance")
  datal <- data[,-col]
  abundance <- as.numeric(datal[,col])
  result1 <- data.frame(rep(NA, sum(abundance)))
  colnames(result1) <- "plots"
  for (i in 1:(ncol(data)-1))
  {
    result1[, i] <- rep(as.character(data[, i]), abundance)
  }
  result <- table(result1)
  return(result)
}
```

Highlighting of key words
Text Editors: TinnR

Code Highlighting by TinnR
Selected Publications
Many Thanks!

Questions?