JANUARY 13, 2004

STATISTICAL ANALYSIS WITH R

Terry A. Cox, M.D., Ph.D.
National Eye Institute

Course Outline

1. The R website and documentation
2. Installing and updating R
3. The Windows GUI
4. R language essentials
5. R graphics
6. Basic statistics in R

URLs

- http://www.r-project.org/

  The source for R software and documentation. Links to ESS and R-Winedt, which provide R support for EMACS and Winedt, respectively.

- http://www.bioconductor.org/

  Bioconductor is an open source and open development software project for the analysis and comprehension of genomic data.

- http://gifi.stat.ucla.edu/pub/

  R for Mac OS X. Also includes Python and Emacs.

- http://www.cs.wisc.edu/∼ghost/

  Ghostscript and GSview, software for interpreting and viewing PostScript files.

- http://www.math.auc.dk/∼dethlef/Tips/introduction.html

  Installation instructions for EMACS and LaTeX for MS Windows. Also covers Ghostscript.

- http://www.gnu.org/software/emacs/emacs.html

  Site for current EMACS versions.

- http://www.winedt.com/

  Shareware text editor for Windows designed for use with LaTeX. R-Winedt provides integration with R.

- http://www.textpad.com/

  An easy-to-use inexpensive text editor for Windows. Syntax definition files for R are available (see the add-ons directory at the website).

Documentation

R comes with several manuals in both html and pdf formats. Of particular relevance is *An Introduction to R.* Also the Contributed Documentation section at the R website contains several introductory manuals. In addition, the r-help mailing list is quite active, and search facilities are available for its archives. *R News*, available at the R website, is also very useful.

Books

- *Introductory Statistics with R*
  by Peter Dalgaard
  Publisher: Springer Verlag
  ISBN: 0387954759
  Publication Date: August 2002
  List Price: $44.95
  Paperback: 288 pages

  — Excellent for getting started with R. Covers basic statistical analysis, as well as linear models, logistic regression, and survival analysis.

- *Modern Applied Statistics with S*, 4th edition
  by Brian D. Ripley and William N. Venables
  Publisher: Springer Verlag
  ISBN: 0387954570
  Publication Date: July 2002
  List Price: $69.95
  Hardcover: 512 pages

  — Intermediate-level text that includes many state-of-the-art methods.

- *Regression Modeling Strategies*
  by Frank E. Harrell
  Publisher: Springer Verlag
  ISBN: 0387952322
  Publication Date: June 2001
  List Price: $79.95
  Hardcover: 582 pages

  — Lots of good stuff on linear models, logistic regression, and survival analysis.

- *The Elements of Graphing Data (Revised Edition)*
  by William S. Cleveland
  Publisher: Hobart Press
  ISBN: 0963488414
  Publication Date: September 1994
  List Price: $45.00
  Hardcover: 297 pages

  — Highly recommended. R uses Cleveland’s principles by default.

R Examples

Preliminaries

Before proceeding install the packages **ISwR**, **car**, and **locfit**.

If you want to type in the following listings, replace the left arrow symbols with “<–”. The equals sign, “=”, can also be used, except where noted.

Vectors

```
x ← c(92,63,22,32,56,80,51,14,21,38) # Or x ← scan()
x
x[1]
x[2:4]
x[seq(1,9,2)]
?seq
x[-1]
x[-(2:4)]
x[c(1,3,5)]
(x > 50)
x[x > 50]
which(x > 50)
sort(x)
rev(x)
c.x ← c(rep("Boy",5),rep("Girl",5))
# Or c.x ← rep(c("Boy","Girl"), c(5,5))
is.character(c.x)
mode(c.x)
```

Computations

```
2*x
x^2
xbar ← mean(x)
xv ← (x - xbar)^2
xvar ← sum(xv)/(length(x)-1)
xsd ← sqrt(xvar)
sd(x)
```

Missing Data

```
x.m ← x
x.m[x>50] ← NA
x.m
mean(x.m)
mean(x.m, na.rm=TRUE)
```

Matrices

```
y ← c(79,24,38,45,64,58,20,53,15,83)

z ← cbind(x,y)
z[,"x"]
which(z>50)
which(z>50, arr.ind=T)
matrix(y, nrow=2, byrow=TRUE)
matrix(1,2,3)
zcp ← t(z) %*% z
diag(zcp)
diag(3)
```

Lists and data frames

```
z.lst ← list(first=x, second=y, gender=c.x)
z.lst$first

z.df ← data.frame(first=x, second=y, gender=c.x)
z.df$first
z.df[z.df$first>50,]

lapply(z.df[,1:2], mean)
sapply(z.df[,1:2], mean) # See also apply, mapply, tapply
lapply(z.df[,sapply(z.df,is.numeric)], mean)

z.df[grep("b", as.character(z.df$gender), ignore.case=TRUE),]
```

Miscellaneous

```
search()
ls()
rm(z.lst)
?"%*%"
help.search("fisher")
options()

?kfm
library(ISwR)
data(kfm)
```

The functions, source() and sink(), are also frequently useful.

Data entry

```
k.df ← read.delim("R:/R Course/kfm.txt")
summary(k.df)
fix(k.df) # Use only for looking at data

attach(k.df)
search()
mat.height
detach(k.df)
```

See also the R manual, *R Data Import/Export*, and the package, **foreign**.

Summary plots

```
hist(k.df$weight, xlab="Weight (kg)")

boxplot(weight ∼ sex, data=k.df, boxwex=0.3, ylab="Weight (kg)", names=c("Boys","Girls"))
```

See Figure 1 for an example of a dot plot, an alternative to bar graphs.

Scatter plots

```
plot(k.df$mat.weight, k.df$weight)
plot(weight ∼ mat.weight, data=k.df)

attach(k.df)
library(locfit)
fit ← locfit(weight ∼ mat.weight)
plot(fit, band="global")
points(mat.weight, weight, pch=20, col="gray50")
detach(k.df)

library(car)
scatterplot(weight ∼ mat.weight, reg.line=lm, smooth=TRUE, labels=FALSE,
    boxplots='xy', span =0.5, data=k.df)

pairs(k.df[,c("weight","mat.weight","mat.height")])
```

Cumulative histogram

The following code plots the empirical cumulative distribution function:

```
attach(k.df)
boy.wt ← weight[sex=="boy"]
girl.wt ← weight[sex=="girl"]
m ← length(boy.wt)
n ← length(girl.wt)
plot(sort(boy.wt), (1:m)/m, type="s", ylim=c(0,1), xlim=range(weight),
    xlab="Weight", ylab="Cumulative frequency", lty=1)
lines(sort(girl.wt), (1:n)/n, type="s", lty=2)
legend(c(6,6.5), c(0.14,0.3), legend=c("Boys","Girls"), lty=1:2)
detach(k.df)
```

See also the function, ecdf(), in Frank Harrell's **Hmisc** library.

Line plots

```
x ← seq(-4,4,0.01)
y ← dnorm(x)
plot(x, y, type="l", main="Normal Density", xlab="x", ylab=substitute(paste(phi, "(x)")))
text(-3, 0.2, expression(phi), cex=2, col="gold") # For fun
phi ← (sqrt(5) + 1)/2
text(3, 0.2, phi)
```

Plot output

The following code produces Figure 2:

```
attach(k.df)
postscript(file="R:/R Course/Fig2.ps", horizontal=F, width=5, height=5)
plot(weight ∼ mat.weight, type="n", xlab="Maternal weight", ylab="Infant weight")
points(weight[sex=="boy"] ∼ mat.weight[sex=="boy"], pch=19, col="blue")
points(weight[sex=="girl"] ∼ mat.weight[sex=="girl"], pch=19, col="red")
dev.off()
detach(k.df)
```

Postscript files produce publication-quality graphics on laser printers, and they can be used in LaTeX documents. To create an encapsulated postscript file that can be imported into MS Word, substitute the following line:

```
postscript(file="R:/R Course/Fig2.eps", horizontal=FALSE, onefile=FALSE,
    paper="special", width=5, height=5)
```

The following code produces a graphic that can be imported into MS Powerpoint:

```
win.metafile("R:/R Course/boxplot.wmf")
old.par ← par(no.readonly=TRUE)
line.col ← "gray"
boxplot(weight ∼ sex, data=k.df, boxwex=0.3, ylab="", names=c("Boys","Girls"),
    notch=TRUE, col=heat.colors(2), border=line.col,
    pars=par(fg=line.col, col.axis=line.col, col.lab=line.col, cex=1.5, lwd=2))
mtext("Weight (kg)", side=2, line=2.5, cex=1.5)
par(old.par)
dev.off()
```

Summary statistics

```
mean(k.df$weight)
sd(k.df$weight)
quantile(k.df$weight)
median(k.df$weight)
IQR(k.df$weight)
mad(k.df$weight)

table(k.df$sex)
```

Tabular data

```
wtable ← table(k.df$sex, (k.df$weight < 5.5) )
ft ← fisher.test(wtable)
ct ← chisq.test(wtable, correct=FALSE)
```

For other relevant functions, see the documentation for the package, **ctest**. The package, **vcd**, contains functions for Cohen's kappa and weighted kappa, among others.

t tests

```
t.test(weight ∼ sex, data=k.df)
wilcox.test(weight ∼ sex, data=k.df)
```

Correlation

```
cor(k.df[,c("weight","mat.weight","mat.height")])
cor(k.df[,c("weight","mat.weight","mat.height")], method="spearman")
```

See also the function, **cor.test**(), in the package, **ctest**.

Linear regression

```
k.lm ← lm(weight ∼ mat.weight, data=k.df)
summary(k.lm)
k.lm.summ ← summary(k.lm)
names(k.lm)
names(summary(k.lm))
summary(k.lm)$r.squared
k.lm$coefficients
summary(k.lm)$coefficients
plot(k.lm)

# Regression lines and confidence intervals

pwt ← seq(min(k.df$mat.weight), max(k.df$mat.weight), 0.05)
clim ← predict(k.lm, data.frame(mat.weight=pwt), interval="c")
plim ← predict(k.lm, data.frame(mat.weight=pwt), interval="p")
plot(weight ∼ mat.weight, data=k.df, ylim=range(plim[,2:3]))
lines(pwt, clim[,1], lty=1, col="black")
lines(pwt, clim[,2], lty=2)
lines(pwt, clim[,3], lty=2)
lines(pwt, plim[,2], lty=3)
lines(pwt, plim[,3], lty=3)

# Alternatively, add lines as follows:

matlines(pwt, clim, lty=c(1,2,2), col="black")
matlines(pwt, plim[,2:3], lty=3, col="black")
```

Logistic regression

```
k.glm ← glm(sex ∼ mat.weight*mat.height, data=k.df, family="binomial")
summary(k.glm)
```

Function creation

```
summ.var ← function(y) {
    q.y ← quantile(y, na.rm=T)
    names(q.y) ← c("minimum","1st quartile","median","3rd quartile","maximum")
    c(N = length(y),
      mean = mean(y, na.rm=T),
      "st. dev." = sd(y, na.rm=T),
      q.y[1], q.y[2], q.y[3], q.y[4], q.y[5],
      IQR = IQR(y, na.rm=T),
      "mean abs. dev." = mad(y, na.rm=T),
      missing = sum(is.na(y)),
      "Shapiro-Wilk test" = shapiro.test(y)$p.value )
}
```

```
proc.univariate ← function(y) {
    if (is.numeric(y)) {
        out ← summ.var(y)
    }
    else {
        y.num ← y[,sapply(y,is.numeric)]
        if (is.list(y.num)) out ← sapply(y.num, summ.var) # or lapply
        else {
            out ← list(summ.var(y.num))
            names(out) ← names(y)[sapply(y,is.numeric)]
        }
    }
    return(out)
}

options(digits=4)
proc.univariate(k.df)

proc.univariate
```

See also the function, describe(), in Frank Harrell's **Hmisc** library.

Random numbers

```
sample(1:10)
sample(LETTERS, size=10, replace=T)
set.seed(10); runif(1)

k.df[sample(1:length(k.df$no), size=5),]

x ← rnorm(1000)
hist(x,nclass=20)
```

Functions are automatically available for a number of distributions in R. See also the packages, **bindata**, **mvtnorm**, **SuppDists**, **MCMCpack**, and **MASS**.

```
# Poker

suit ← rep(c("Diamonds","Clubs","Spades","Hearts"), rep(13,4))
card ← rep(c(2:10,"Jack","Queen","King","Ace"), 4)
deck ← paste(card, suit, sep=" of ")

shuffle ← sample(deck)
player.1 ← shuffle[1:5]
player.2 ← shuffle[6:10]
```

Create external dataset

```
k.df$wtbin ← (k.df$weight < 5.5)

zz ← file("R:/R Course/temp.data", "w")
write.table(k.df, file=zz, sep="", quote=FALSE, row.names=FALSE)
close(zz)
```

Exercise

Using the malaria dataset in the **ISwR** library, investigate the relationship between antibody levels and symptoms of malaria with graphics, data summaries, and statistical tests.

Extra credit: Use graphics and statistical models to investigate the influence of age on the relationship between antibody levels and symptoms of malaria.

Figures

Figure 1: Example of a dot plot.

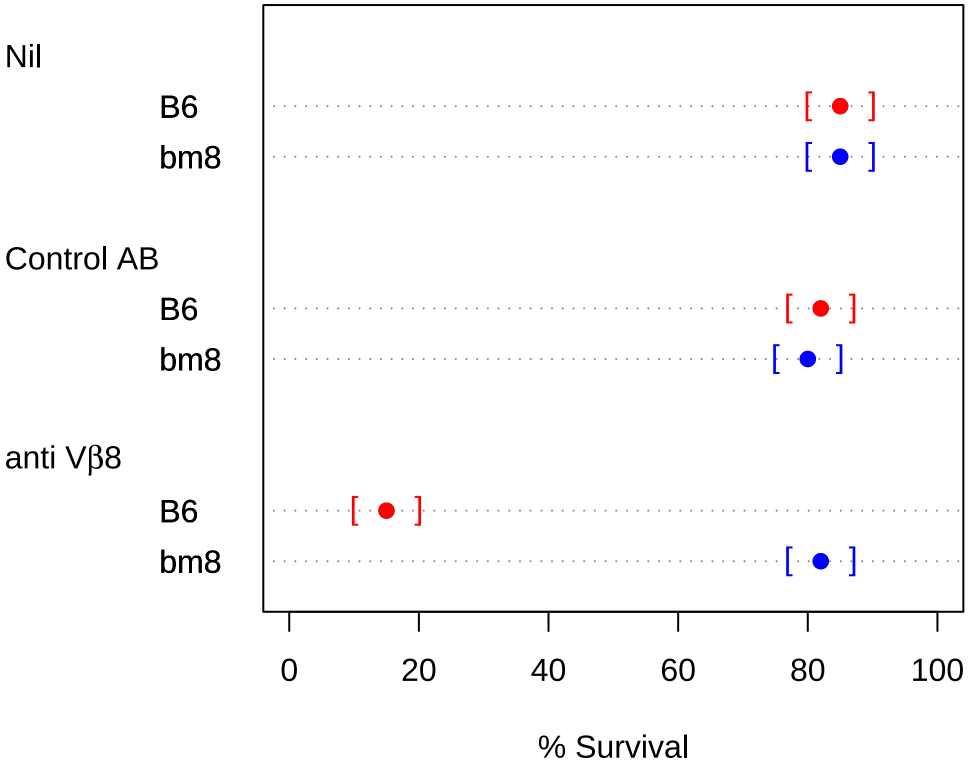

Figure 2: Example of a plot good enough to publish.

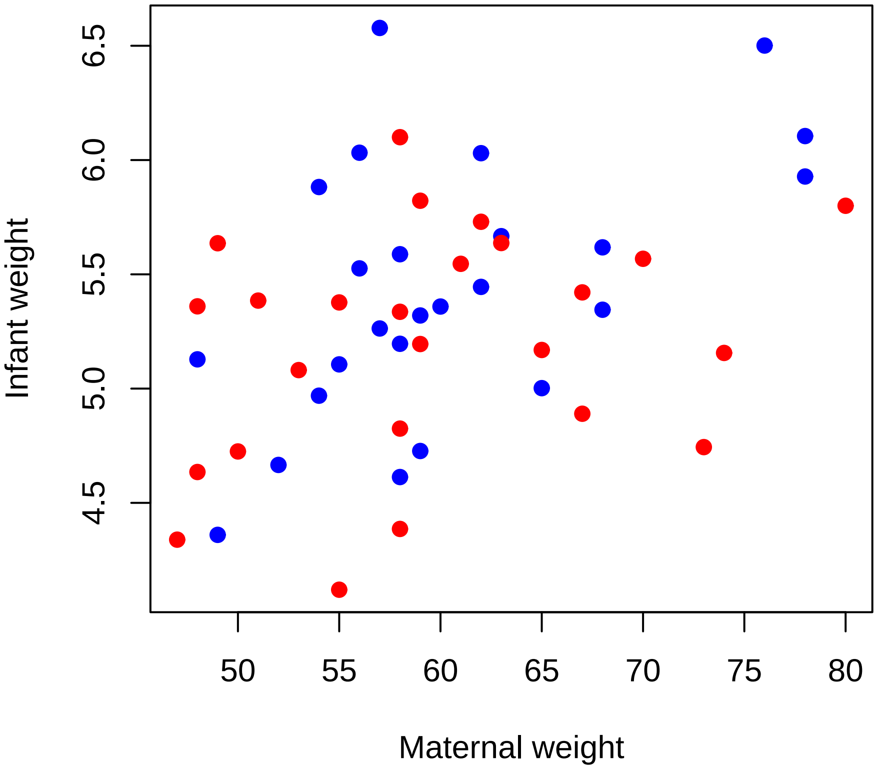

Figure 3: R plotting symbols.

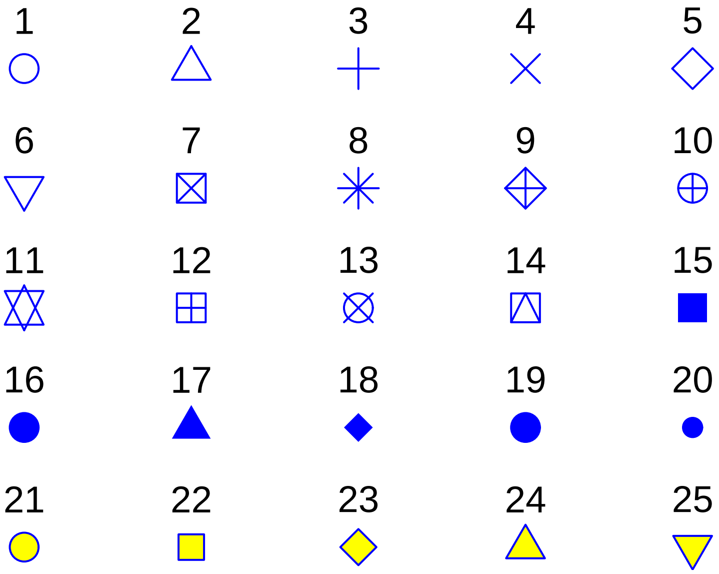

Figure 3 was produced using the following code:

```
postscript(file="R:/R Course/chars.ps", horizontal=F, width=5, height=5)
x ← rep(1:5,5)
y ← rep(5:1,rep(5,5))
z ← rep(seq(5.4,1.4,-1),rep(5,5))
plot(x,y,pch=1:25,type="n",axes=FALSE,xlab="",ylab="",ylim=c(1,5.7))
points(x,y,pch=1:25,cex=2,col="blue",bg="yellow")
text(x,z,pch=as.character(1:25),cex=1.2)
dev.off()
```

î€Institute of Mathematical Statistics, 2008. The Statistical Analysis of fMRI Data. Martin A. Lindquist. Abstract. In recent years there has been explosive growth in the num-. ber of neuroimaging studies performed using functional Magnetic Res-. onance Imaging (fMRI). The ï¬eld that has grown around the acquisiÂ ing techniques, the role of statisticians promises to. only increase in the future. The statistical analysis of fMRI data is challeng-. ing. The data comprise a sequence of magnetic reso-. nance images (MRI), each consisting of a number of. uniformly spaced volume elements, or voxels, that. partition the brain into equally sized boxes. The. 15 Analysis of variance and covariance. 15.1 ANOVA 15.1.1 Single factor or one-way ANOVA 15.1.2 Two factor or two-way and higher-way ANOVA.Â For students studying for academic or professional qualifications in statistics, the level and content adopted is that of the Ordinary and Higher Level Certificates of the Royal Statistical Society (RSS), offered until 2017. Much of the material included in this Handbook is also appropriate for the Graduate Diploma level also, although we have not sought to be rigorous or excessively formal in our treatment of individual statistical topics, preferring to provide less formal explanations and examples that are more approachable by the non-mathematician with links and references to detailed sourc