

R commands for Statistics for Biologists

```
# Get the working directory
getwd()

# Change the working directory
setwd("C:/Documents and
Settings/student/Desktop")
setwd("/Users/student/Desktop")

# Read a tabbed data table
energy = read.delim("energy.txt",
header=T)
intake = read.delim(file("clipboard"),
header=T)
rats =
read.delim("http://iona.wi.mit.edu/bio/ed
ucation/stats2005/exercises/data/rats.txt
", header=T)

# Read and execute an R script
source("script1.R")

# Write data to a file
write.table(exp.p, file =
"pvals_Bonferroni.txt",

# Look at data like in a spreadsheet
edit(energy)

# Get the mean, median, stdev, or summary
mean(energy$lean)
median(energy$lean)
sd(energy$lean)
summary(energy)

# Draw a histogram
hist(energy$lean, breaks=20, col="green")

# Draw a stripchart
stripchart(energy, vertical=T,
method="jitter")

# Draw a boxplot
boxplot(energy, col=c("red", "blue"),
main="This is a boxplot in R", xlab="Exp
groups", ylab="Units go here")

# Get a list of previous commands
history()

# Quit R
q()

# Perform a t test
t.test(energy$lean, energy$obese)

# Perform a F test
var.test(energy$lean, energy$obese)

# Do a power calculation for a t test
power.t.test(n=9, delta=NULL, sd=1.3,
sig.level=0.05, power=0.9)

# Do a Wilcoxon signed rank test
wilcox.test(energy$lean, energy$obese)

# Correct p-values for multiple
# hypothesis testing
# See '?p.adjust' for methods
pvals.fdr = p.adjust(raw.pvals, "fdr")

# Perform a Fisher's exact test
fisher.test(ko)

# Perform a Chi-squared test
chisq.test(ko)

# Test for enrichment with the
# hypergeometric distribution
# h = number of hits in the gene set (the
sample)
# H = number of hits in the genome (the
population)
# n = number of objects (genes) in the
gene set
# N = number of genes in the genome
pvals = 1 - cumsum(dhyper(0:h, H, N-H, n)
)
pvals[length(pvals)]

# Plot survival function data
library(survival)
my.surv.all = survfit(Surv(days, dead),
data=rats)
plot(my.surv.all)

# Test for a difference
# between survival curves
survdif(Surv(days, dead) ~ drug,
data=rats)

# Perform linear regression analysis
# and plot the output
lm.v.g = lm(velocity ~ glucose,
data=heart)
plot(heart$glucose, heart$velocity,
pch=19, cex=2, col="orange",
main="Regression line: velocity ~
glucose")
lines(heart$glucose, fitted(lm.v.g),
lwd=2, col="blue")

# Measure correlation
cor(x, y)
```