Summer 2015: R Workshop 1 Method List

Module 1: Importing and Manipulating Data

getwd()
lists what your current working directory is

setwd("folder name")
sets your working directory to whatever folder path you give, usually something like "C:/My Documents/R data"

install.packages("package name")
installs packages that may be useful to you like xlsx for importing data in Excel, vcd for categorical variable plotting

library(package name) [do not add quotation marks around package name]
loads package into R environment

read.xls("filename.xlsx") [remember to add .xlsx to end of filename]
loads Excel data into R environment, works when either gdata or xlsx package

<- [assignment variable] *z <- 3, z has been assigned to 3*

c( ) [concatenate] *makes a vector, ex.* c(1,2,3) gives you a vector of [1,2,3]

matrix() *makes a matrix* matrixex <- matrix(c(1, 2, 3, 4, 5, 6), nrow=2, ncol=3, byrow=TRUE)

$ *pulls out data from a data frame* moddata$flower.color*, pulls the flower.color column from the moddata data frame*

Module 2: Displaying and Summarizing Data

barplot(data, main ="title", xlab = "x-axis label", ylab = "y-axis label", xlim = c(x-range), ylim = c(y-range))
barplot method produces a bar graph, the other plotting methods follow the same format, this includes hist (*for histogram*), plot (*for scatter plots)*, boxplot (*for boxplots*)

table(vector1, vector2) *makes a table from two data vectors*

mean(vector1) *gives the mean for a numerical data vector*

median(vector1) *gives the median for a numerical data vector*

sd(vector1) *gives the standard deviation for a numerical data vector*

IQR(vector1) *gives the interquartile range for a numerical data vector*

summary(data frame) *gives mean, median, IQR for all data columns in data frame*

Module 3: Hypothesis Testing

Goodness-of-fit test
results <- chisq.test(observed vector1, p = vector of expected probabilities)

Chi-square test for independence
results <- chisq.test(contingency table)

t-test for paired data
results <- t.test(treatment1 vector, treatment2 vector, paired = TRUE)

t-test for unpaired data
results <- t.test(treatment1 vector, treatment2 vector, alternative = "greater", var.equal=T)
removes the Welch's correction from the t-test

one-way ANOVA
results <- aov(dep.var~indep.var, data = data frame)
remember to format your data correctly before applying aov