An Introductory Tutorial:
Learning R for Quantitative Thinking in the Life Sciences

Scott C Merrill

September 26th, 2012

Chapter 5

Elements in R from class

Run through the code below again to make sure you understand what is going on. Check with me if you are not sure.

```
A = matrix(data = 0,nrow=5,ncol=2)
A[,1] = c(1995:1999)
A[,2] = c(10.0,12.5,15.6,19.5,24.4)

# without comma
A[ A[,2] < 19]

# with comma
A[ A[,2] < 19,]

# break it down into pieces
A[,2] < 19
A[TRUE]

# Looking at elements in a matrix
A[c(TRUE,TRUE,TRUE,FALSE,FALSE)]

# So confusing!!! R given partial information,
#                    will continue with the query based on the information given
# for example, given a short query:

A[c(TRUE,FALSE,FALSE)]

# R will repeat the three logical steps for the whole matrix.
# This makes the above statement read the same as the below statement:

A[c(TRUE,FALSE,FALSE,TRUE,FALSE,FALSE,TRUE,FALSE,FALSE,TRUE)]
```

```
# Let's revisit: take a step back. Compile:

A[6]

# now compile:
A[c(1:6)]

# The above statement asks R to write the sequence of elements
#                   1 to 6 to the console
# you will note that R reads elements in the matrix vertically
#                   first and then horizontally left to right

# Elements can also be used in arrays with the read being:
#                   vertically, horizontally(L-R) then next "worksheet"
                    (or however you think about it)
# Arrays actually allow a different visualization of the read order:
#         The read for an array of dim = c(3,5,2) is read order first
#                   number (3), second number (5), and then third number (2)

# for example:
array1 <- array(sample(c(1:6),30,replace=TRUE), dim = c(3,5,2))
array1

# values in elements 13-18 cross from the first "worksheet" to the second "worksheet"
array1[c(13:18)]


# Why would you ever want to use elements?
#                   1) An alternate way to search/subset a matrix or array
#                   e.g., finding the sum of values in all elements of an array that are greater than 3
                    sum(array1[array1>3])
#                   2) Elements are also used in objects (and can be very useful there)
#                   e.g., the statistical example below:


# Using our A matrix create a simple model that linearly regresses numbers in column 2
#                   against year. That is, are the numbers good predictors of year
model1 = lm(A[,1] ~ A[,2])


model1$coefficients
model1[1] # coefficients
model1[2] # residuals
model1[3] # effects
```

```
model1[4] # rank
model1[5] # fitted values
model1[6] # assign
model1[7] # some others such as $tol, $pivot

plot(model1$residuals)

model1$residuals[3] # third element of the residuals

# elements can be used in functions / calculations etc.
model1$residuals[3]*3

# this is more a note so that you are aware that you can extract and use

# There are some helpful extraction elements
summary(model1)
summary(model1)$r.squared
summary(model1)$adj.r.squared

# loop that runs through elements 1 to 11 of the summary() function
for (x in 1:11) {
	print(summary(model1)[x])
}

#####
# subsetting!
model1$coefficients[model1$coefficients>10]
```

Exercises

Access the ChickWeight dataset. Compile:

```
> require(datasets)

> data(ChickWeight)
# note ChickWeight is a data frame not a matrix. There are differences between matrices
#	including differences in the Elements of a data frame compared to a matrix.
> ChickWeight
> attach(ChickWeight)
> names(ChickWeight)
```

How many rows of data exist when time is greater than 14 days?

Create a subset of the ChickWeight data where Diet equals 4.

Create a subset of the ChickWeight data where Time is greater than 14. Plot the Diet column by the Weight column. How smart is R (This should give you a boxplot). What does a box plot tell you?

Another looping if then exercise

Return to Chapter 4's if then exercise pine trees and fire. Add in a fire to plots 2, 4, & 6 on year 7.

Add in mortality of beetle kill to mature trees. Mature trees have a 1/10 probability of dying each year from beetles.

Hints:

For every plot that has mature trees, create a for loop based on the number of mature trees and randomly kill off trees if their number comes up. Reduce the mature tree population by 1 if they die.

```
> death = sample(1:10,1)
```

Create a new matrix the holds values of the number of mature trees over the ten year period. Plot this matrix. Hint: matplot() from Chapter 4 exercises.