R Webseries-Week 1

Jacob Burbank, Justin Hubbard & Erik Dean

09/09/2020

How to Use: Basic Commands

```
a <- 10
b <- 2.5

a
```

```
## [1] 10
```

```
b
```

```
## [1] 2.5
```

```
a+b
```

```
## [1] 12.5
```

```
d <- a - b

d
```

```
## [1] 7.5
```

```
c
```

```
## function (...)  .Primitive("c")
```

How to Use: Functions

```
g <- c(5,3.14)

g
```

```
## [1] 5.00 3.14
```

```r
class(g)
```

```
## [1] "numeric"
```

```r
is.numeric(g)
```

```
## [1] TRUE
```

Data Structures

Vector

```r
vecA<-c(a,b,d)
vecA
```

```
## [1] 10.0  2.5  7.5
```

Data Frame

```r
dat <- data.frame(x = 1, y = 1:10, z = "r")
head(dat) #head() prints first 6 rows of data
```

```
##   x y z
## 1 1 1 r
## 2 1 2 r
## 3 1 3 r
## 4 1 4 r
## 5 1 5 r
## 6 1 6 r
```

Matrix

```r
mat1<-matrix(nrow=3,ncol=3)
mat1
```

```
##      [,1] [,2] [,3]
## [1,]   NA   NA   NA
## [2,]   NA   NA   NA
## [3,]   NA   NA   NA
```

```r
mat2<-as.matrix(dat)
mat2
```

```
##       x   y    z
##  [1,] "1" " 1" "r"
##  [2,] "1" " 2" "r"
##  [3,] "1" " 3" "r"
##  [4,] "1" " 4" "r"
##  [5,] "1" " 5" "r"
##  [6,] "1" " 6" "r"
##  [7,] "1" " 7" "r"
##  [8,] "1" " 8" "r"
##  [9,] "1" " 9" "r"
## [10,] "1" "10" "r"
```

Data Continued

```
#mean#
mean(dat$y)
```

```
## [1] 5.5
```

```
#sd
sd(dat$y)
```

```
## [1] 3.02765
```

```
#Round off
round(sd(dat$y), 2)
```

```
## [1] 3.03
```

```
#sample data
s.dat<-sample(dat$y, size=3)
s.dat
```

```
## [1] 1 9 8
```

Exploring Data

```
#Data Structure#
str(dat)
```

```
## 'data.frame':    10 obs. of  3 variables:
##  $ x: num  1 1 1 1 1 1 1 1 1 1
##  $ y: int  1 2 3 4 5 6 7 8 9 10
##  $ z: chr  "r" "r" "r" "r" ...
```

```r
#Length of Element#
length(dat)
```

```
## [1] 3
```

```r
length(dat$y)
```

```
## [1] 10
```

```r
#tail showing last 5 rows#
tail(dat)
```

```
##    x  y z
## 5  1  5 r
## 6  1  6 r
## 7  1  7 r
## 8  1  8 r
## 9  1  9 r
## 10 1 10 r
```

```r
#minimum of Column#
min(dat$y)
```

```
## [1] 1
```

```r
#Maximum of Column#
max(dat$y)
```

```
## [1] 10
```

Manipulating Data

```r
subdat <- head(dat,5)

#add new row#
rbind(subdat,c(2,1,"r"))
```

```
##   x y z
## 1 1 1 r
## 2 1 2 r
## 3 1 3 r
## 4 1 4 r
## 5 1 5 r
## 6 2 1 r
```

```r
#Add new coloumn#
cbind(subdat,ID = c(letters[1:5]))
```

```
##   x y z ID
## 1 1 1 r  a
## 2 1 2 r  b
## 3 1 3 r  c
## 4 1 4 r  d
## 5 1 5 r  e
```

```
#Change column values#
subdat$z <- "q"
subdat
```

```
##   x y z
## 1 1 1 q
## 2 1 2 q
## 3 1 3 q
## 4 1 4 q
## 5 1 5 q
```

```
names(subdat) <- names(trees)
rbind(trees,subdat)
```

```
##    Girth Height Volume
## 1    8.3     70   10.3
## 2    8.6     65   10.3
## 3    8.8     63   10.2
## 4   10.5     72   16.4
## 5   10.7     81   18.8
## 6   10.8     83   19.7
## 7   11.0     66   15.6
## 8   11.0     75   18.2
## 9   11.1     80   22.6
## 10  11.2     75   19.9
## 11  11.3     79   24.2
## 12  11.4     76     21
## 13  11.4     76   21.4
## 14  11.7     69   21.3
## 15  12.0     75   19.1
## 16  12.9     74   22.2
## 17  12.9     85   33.8
## 18  13.3     86   27.4
## 19  13.7     71   25.7
## 20  13.8     64   24.9
## 21  14.0     78   34.5
## 22  14.2     80   31.7
## 23  14.5     74   36.3
## 24  16.0     72   38.3
## 25  16.3     77   42.6
## 26  17.3     81   55.4
## 27  17.5     82   55.7
## 28  17.9     80   58.3
## 29  18.0     80   51.5
## 30  18.0     80     51
## 31  20.6     87     77
```

```
## 32   1.0      1      q
## 33   1.0      2      q
## 34   1.0      3      q
## 35   1.0      4      q
## 36   1.0      5      q
```

Vector Operations

```
vecB <- 1:3

vecB
```

```
## [1] 1 2 3
```

```
vecB/2
```

```
## [1] 0.5 1.0 1.5
```

```
vecA * vecB
```

```
## [1] 10.0  5.0 22.5
```

```
vecB * head(cars)
```

```
##   speed dist
## 1     4    2
## 2     8   20
## 3    21   12
## 4     7   22
## 5    16   32
## 6    27   30
```

Data Frame Operations

```
dat2<-dat[1:5,1:2]
dat2
```

```
##   x y
## 1 1 1
## 2 1 2
## 3 1 3
## 4 1 4
## 5 1 5
```

```
head(cars, 5) * dat2
```

```
##   speed dist
## 1     4    2
## 2     4   20
## 3     7   12
## 4     7   88
## 5     8   80
```

Writing Your Own Functions

```
my.dat <- 1:10

sample(1:10,5)
```

```
## [1]  2 10  8  6  1
```

```
my.fun <- function(x){
            temp<-sample(x, 5)
              mean(temp)
         }

my.fun(my.dat)
```

```
## [1] 5.4
```

```
my.fun2 <- function(){
            temp<-sample(x, 5)
            mean(temp)
        }

x <- 1:10

my.fun2()
```

```
## [1] 4.8
```

Installing and Loading Packages

```
##Remember to use install.packages("") first if the package is not yet installed.

library(ggplot2)
library(FSA)
```

```
## ## FSA v0.8.30. See citation('FSA') if used in publication.
## ## Run fishR() for related website and fishR('IFAR') for related book.
```

```
library(FSAdata)
```

```
## ## FSAdata v0.3.8. See ?FSAdata to find data for specific fisheries analyses.
```

```
library(vegan)
```

```
## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.5-6
```

More Practice

```
####Lets load some FSA data and play with it#####
##Darter LEngth Data from Ontario##
DartData<-DarterOnt
str(DartData)
```

```
## 'data.frame':    54 obs. of  3 variables:
##  $ age  : int  1 2 2 2 2 2 2 2 3 3 ...
##  $ tl   : num  37.8 39.7 43.6 46.7 47.6 50.4 51.2 56.2 60.5 56.2 ...
##  $ river: Factor w/ 2 levels "Salmon","Trent": 1 1 1 1 1 1 1 1 1 1 ...
```

```
##There is two rivers so lets look at just the Trent##
TrentDart<-subset(DartData, river == "Trent")
head(TrentDart)
```

```
##    age   tl river
## 22   1 43.1 Trent
## 23   2 44.2 Trent
## 24   2 47.2 Trent
## 25   2 48.2 Trent
## 26   2 49.1 Trent
## 27   2 59.4 Trent
```

```
##Lets look at average and SD of length in Trent River##
mean(TrentDart$tl)
```

```
## [1] 55.77879
```

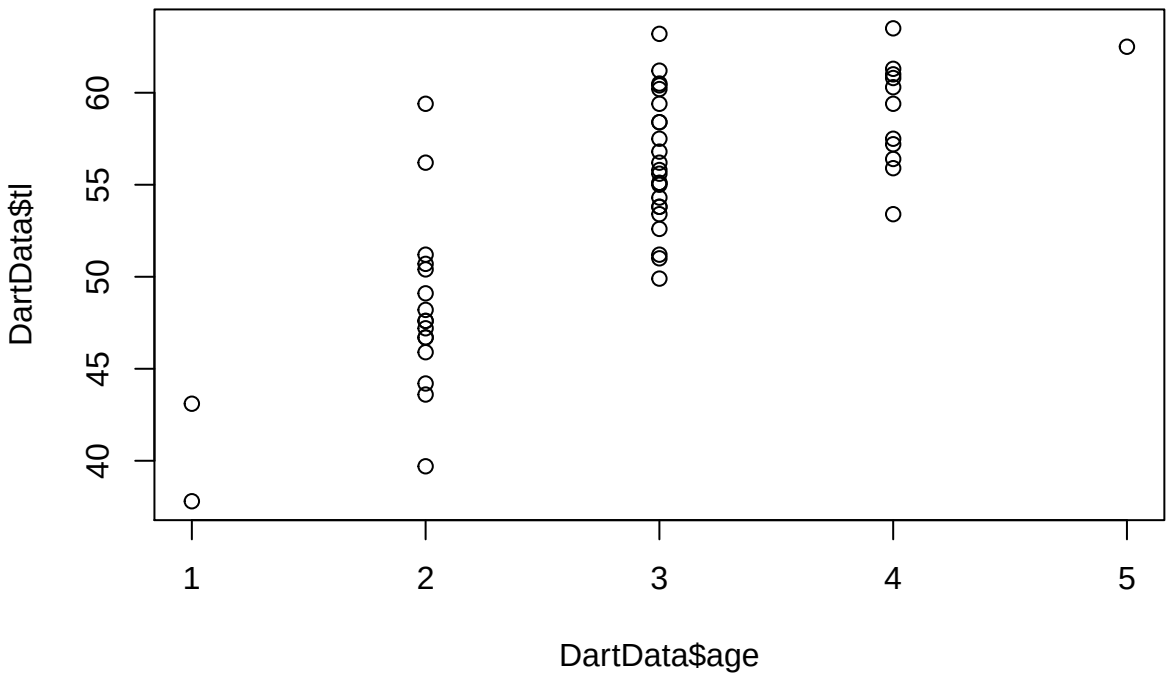
```
sd(TrentDart$tl)
```

```
## [1] 5.574818
```

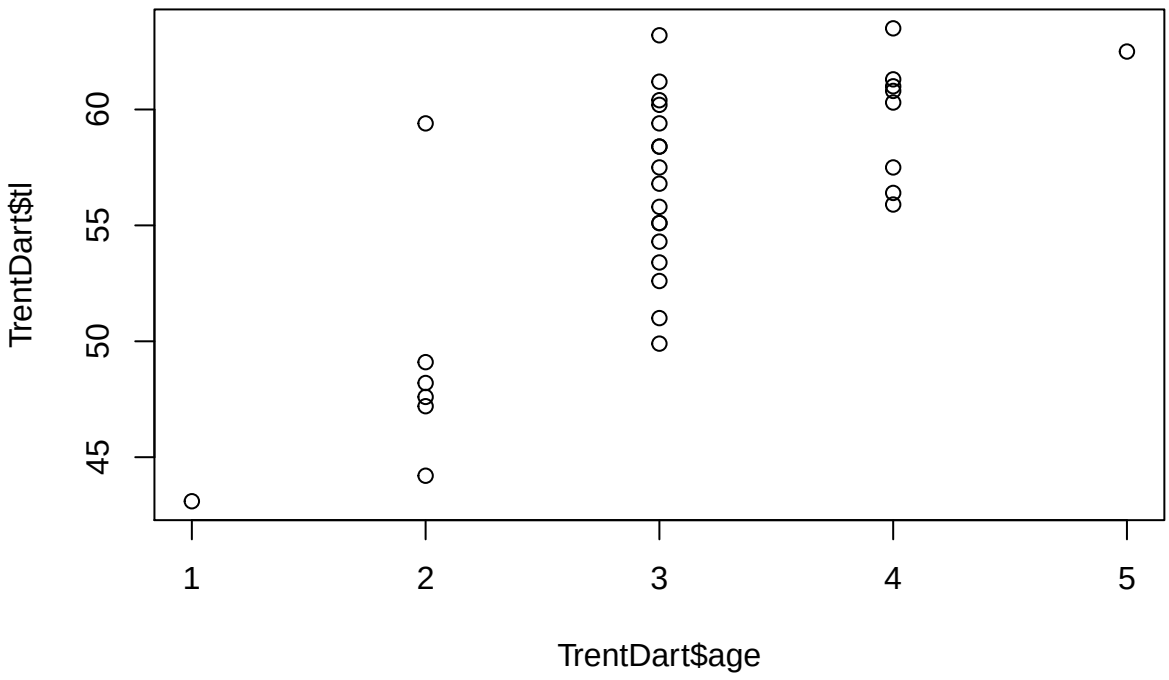
```
##Average age in Trent River##

mean(TrentDart$age)
```

```
## [1] 3.060606
```

```
###Lets make a plot of age vs length at both sites combined##
plot(DartData$age,DartData$tl)
```




```
##How about at Trent River Only###
plot(TrentDart$age, TrentDart$tl)
```




```
###Creel Data Will be loaded###
CreelData<-CreelMN
str(CreelData)
```

```
## 'data.frame':    14550 obs. of  2 variables:
##  $ species: Factor w/ 6 levels "Crappie","LMB",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ harvest: int  6 6 6 6 6 6 6 6 6 6 ...
```

```
#View different species name#
levels(CreelData$species)
```

```
## [1] "Crappie" "LMB"     "NOP"     "Sunfish" "WAE"     "YEP"
```

```
###Subset for LMB###
LMBCreelData<-subset(CreelData, species == "LMB")
```