History of R

- Statistical programming language S developed at Bell Labs since 1976 (at the same time as UNIX)
- Intended to support research and data analysis projects
- Exclusively licensed to Insightful (“S-Plus”)
- R: Open source platform similar to S developed by R. Gentleman and R. Ihaka (U of Auckland, NZ) during the 1990s
- Since 1997: international “R-core” developing team
- Updated versions available every couple months

What R is and what it is not

- R is
  - a programming language
  - a statistical package
  - an interpreter
  - Open Source
- R is not
  - a database
  - a collection of “black boxes”
  - a spreadsheet software package
  - commercially supported

R as a calculator

- R can be used as a calculator:

```
> 5 + (6 + 7) * pi^2
[1] 133.3049
> log(exp(1))
[1] 1
> log(1000, 10)
[1] 3
> sin(pi/3)^2 + cos(pi/3)^2
[1] 1
> Sin(pi/3)^2 + cos(pi/3)^2
Error: couldn't find function "Sin"
```

Basic (atomic) data types

- Logical

```
> x <- T; y <- F
> x; y
[1] TRUE
[1] FALSE
```

- Numerical

```
> a <- 5; b <- sqrt(2)
> a; b
[1] 5
[1] 1.414214
```

- Character

```
> a <- "1"; b <- 1
> a; b
[1] "1"
[1] 1
> a <- "character"
> b <- "a"; c <- a
> a; b; c
[1] "character"
[1] "a"
[1] "character"
```

Vectors

- Vector: Ordered collection of data

```
> x <- c(5.2, 1.7, 6.3)
> log(x)
[1] 1.6486586 0.5306283 1.8405496
> y <- 1:5
> z <- seq(1, 1.4, by = 0.1)
> y + z
[1] 2.0 3.1 4.2 5.3 6.4
> length(y)
[1] 5
> mean(y + z)
[1] 4.2
```

Matrices

- Matrix: Rectangular table of data of the same type

```
> m <- matrix(1:12, 4, byrow = T); m
      [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
[4,]   10   11   12
> y <- -1:2
> m.new <- m + y
> t(m.new)
     [,1] [,2] [,3] [,4]
[1,]    0    4    8   12
[2,]    1    5    9   13
[3,]    2    6   10   14
> dim(m)
[1] 4 3
> dim(t(m.new))
[1] 3 4
```

Missing values

- R is designed to handle statistical data and therefore can deal with missing values
- Numbers that are “not available”

```
> x <- c(1, 2, 3, NA)
> x + 3
[1] 4  5  6 NA
```

- “Not a number”

```
> log(c(0, 1, 2))
[1]      -Inf 0.0000000 0.6931472
> 0/0
[1] NaN
```

Subsetting

- It is often necessary to extract a subset of a vector or matrix
- R offers a couple of neat ways to do that

```
> x <- c("a", "b", "c", "d", "e", "f",
"g", "h")
> x[1]
> x[3:5]
> x[-(3:5)]
> x[c(T, F, T, F, T, F, T, F)]
> x[x <= "d"]
> m[,2]
> m[3,]
```

Importing/Exporting Data

- Importing data
  - R can import data from other applications
  - Packages are available to import microarray data, Excel spreadsheets etc.
  - The easiest way is to import tab or comma delimited files

```
> my.data<-read.table("file",sep=",")
```
*)

```
> CensusData <- read.table(file = "acs_or.csv",
header = TRUE, sep =  ",")
```

- Exporting data
  - R can also export data in various formats
  - Tab delimited is the most common

```
> write.table(x, "filename")
```
*)

*) make sure to include the path or to first change the working directory

Analyzing/Summarizing data

- First, let's take a look…

```
> CensusData[1:10,]
```

- Mean, Variance, Standard deviation, etc.

```
> mean(CensusData[,3])
> mean(log(CensusData[,3]))
> var(CensusData[,3])
> sd(CensusData[,3])
> cor(CensusData[,2:3])
> colMeans(CensusData[2:3])
```

Plotting

- Scatter plot

```
> plot(CensusData[,"income_husband"], CensusData[,"income_wife"], xlab = "husband", ylab = "wife")
```

- Histogram

```
> hist(CensusData[,7])
> hist(CensusData[,7],nclass = 50, main = "Histogram of Electricity (on log scale)")
```

- Boxplot

```
> boxplot(CensusData[,2:3])
> boxplot(CensusData[,2:3], outline = F, boxwex = 0.5, col = 3, main = "Boxplot of CensusData")
```

Getting help… and quitting

- Getting information about a specific command

```
> help(rnorm)
> ?rnorm
```

- Finding functions related to a key word

```
> help.search("boxplot")
```