

A Brief Introduction to R

Matt Bourque

UIC Department of Mathematics, Statistics, and Computer Science

October 15, 2010, UIC CSGSA Friendly Friday

What is R?

R is a free computing environment for statistical computing and graphics. It is a GNU project version of software called S, which was originally developed at Bell Labs. There is a huge number of packages available for R, including ones for

- linear and nonlinear modelling
- standard statistical test
- time series analysis
- graphics

In this talk, I'll focus on some basic graphics capability, because that's the most fun.

Reading data

The basic data structure in R is the data frame, which is an array. Each column of a data frame has a class, such as “numeric” or “factor.” The `read.csv` command can read in data from a CSV file.

```
> cars <- read.csv("cars.csv", header = T)
> cars[1:5, ]
```

	weight	mpg	gpm
1	4.360	16.9	5.917160
2	4.054	15.5	6.451613
3	3.605	19.2	5.208333
4	3.940	18.5	5.405405
5	2.155	30.0	3.333333

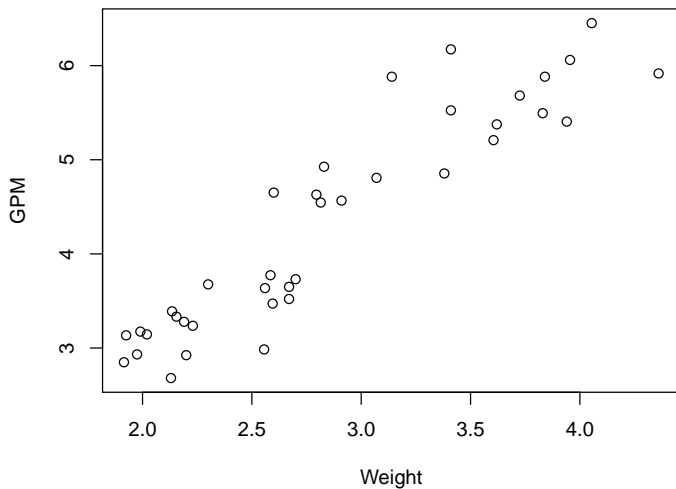
A scatterplot

Unsurprisingly, the `plot` function is used for plotting data. We can get a scatter plot of car weight vs. gallons per mile with

```
> plot(cars$weight, cars$gpm, xlab = "Weight",  
+      ylab = "GPM", main = "Gallons per Mile vs. Weight")
```

A scatterplot

Gallons per Mile vs. Weight



A little statistics

The `lm` command fits a linear model. You can specify the model, we'll just model the GPM response as

$$G_i = \alpha \cdot \text{weight}_i + \beta$$

```
> cars.mod <- lm(gpm ~ weight, data = cars)
> cars.mod
```

Call:

```
lm(formula = gpm ~ weight, data = cars)
```

Coefficients:

(Intercept)	weight
-0.00623	1.51485

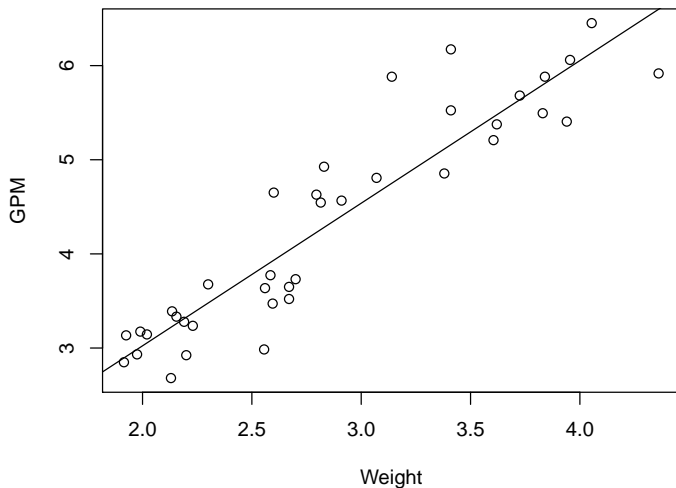
Plotting the fit

And the `abline` command takes the linear model and gets the slope and intercept out to plot the least squares fit line.

```
> plot(cars$weight, cars$gpm, xlab = "Weight",  
+      ylab = "GPM", main = "Gallons per Mile vs. Weight")  
> abline(cars.mod)
```

Plotting the fit

Gallons per Mile vs. Weight



Another example

Apparently, the female cuckoo lays her eggs in the nests of other birds. A paper published in *Biometrika* in 1901 studied whether cuckoo eggs in nests of different birds have different sizes (the better to fool the foster parent birds). We read in the data.

```
> eggs <- read.table("eggs.dat", header = T)
> eggs[11:16, ]
```

```
  length species
11   23.1 sparrow
12   23.5 sparrow
13   23.0 sparrow
14   23.0 sparrow
15   21.8  robin
16   23.0  robin
```

Factors

Since R is for statistics, it has a data class called a “Factor” with “levels”:

```
> attributes(eggs$species)

$levels
[1] "robin"    "sparrow"  "wren"

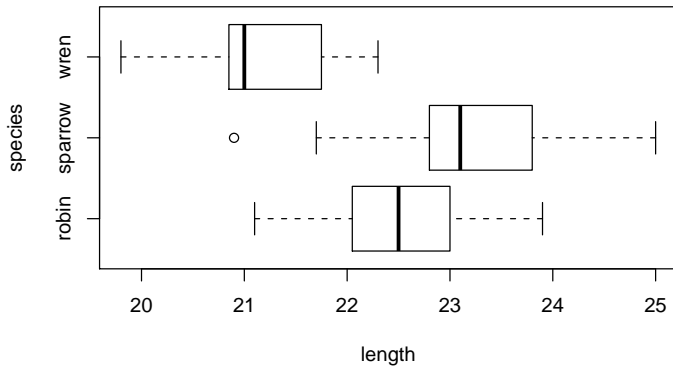
$class
[1] "factor"
```

Egg Boxplot

The `plot` function does different things based on what kind of arguments it gets. If you give it a formula with a factor, it defaults to a boxplot.

```
> plot(length ~ species, data = eggs, horizontal = T)
```

Egg Boxplot



Hypothesis testing

All the basic statistics tests are available as built-in commands. Many more exotic or special-purpose stuff is available in packages. We can do a pairwise test on equality of means of egg length in the cuckoo data:

```
> pairwise.t.test(eggs$length, eggs$species)
```

Pairwise comparisons using t tests with pooled SD

data: eggs\$length and eggs\$species

	robin	sparrow
sparrow	0.075	-
wren	4.2e-05	2.9e-07

P value adjustment method: holm

A few other things...

- built-in functions for drawing from various distributions
- you can write complicated scripts: `function`, `for`, `while`, etc.
- plays nicely with \LaTeX through Sweave package
- big, active, helpful community