WISh course - simple linear regression

Andrew McGovern

29 December 2016

Introduction

Linear regression cannot be considered an easy topic. This tutorial is designed to make building some regression models as easy as possible. We will start with *simple* linear regression.

Difficulty rating: ~~Easy~~, Medium, ~~Hard~~

Areas covered:

1. Importing data from an excel file
2. Scatter plots
3. Testing for correlations
4. Simple linear regression

Libraries

Firstly we will load the library required to import data from an csv file; *Hmisc*.

```
library("Hmisc")
```

```
## Warning: package 'Hmisc' was built under R version 3.2.3
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Warning: package 'survival' was built under R version 3.2.3
```

```
## Loading required package: Formula
```

```
## Warning: package 'Formula' was built under R version 3.2.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.2.3
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

Load the data

Now we can load the data from the excel file “PlasmaVolume.xlsx”. We can then veiw it using the print command. This works well for small datasets but if you have a lot of data then *View(dataFrame)* is better.

```
# first row contains variable names, comma is separator
# assign the variable id to row names

PlasmaData <- read.table("PlasmaVolume.csv", header=TRUE, sep=",")

print(PlasmaData)
```

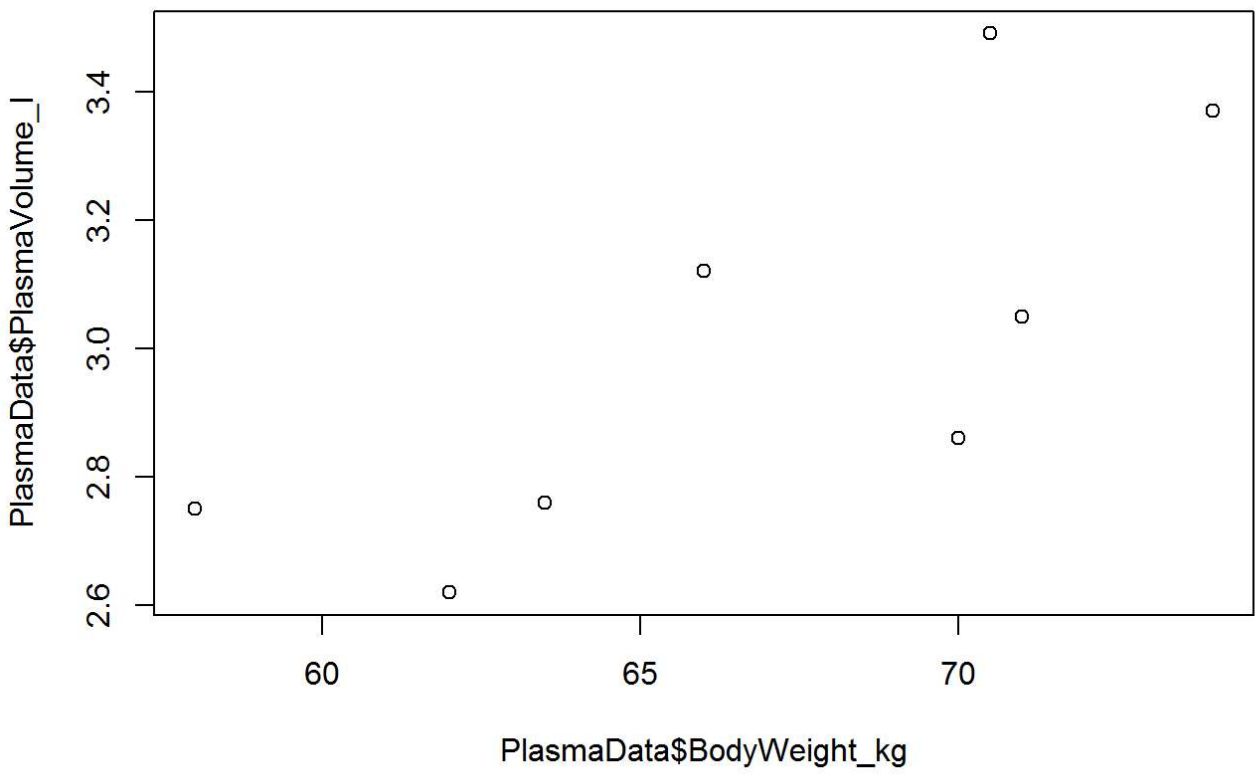
```
##   Subject BodyWeight_kg PlasmaVolume_l
## 1       1          58.0           2.75
## 2       2          70.0           2.86
## 3       3          74.0           3.37
## 4       4          63.5           2.76
## 5       5          62.0           2.62
## 6       6          70.5           3.49
## 7       7          71.0           3.05
## 8       8          66.0           3.12
```

We have two variables body weight in kg and plasma volume in litres. We are going to attempt to predict plasma volume from height using linear regression.

Plot the data

We should start by looking at the relationship between our outcome variable and predictor variable:

```
# Create a scatter plot
plot(PlasmaData$BodyWeight_kg,PlasmaData$PlasmaVolume_l)
```



Measure the correlation

We can use *cor.test(x,y)* to get a measure of the correlation between body weight and plasma volume

```
# Check for a linear correlation
cor.test(PlasmaData$BodyWeight_kg,PlasmaData$PlasmaVolume_l, method="pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  PlasmaData$BodyWeight_kg and PlasmaData$PlasmaVolume_l
## t = 2.8566, df = 6, p-value = 0.02893
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1170885 0.9536551
## sample estimates:
##       cor
## 0.7591266
```

We can see a correlation coefficient of 0.7591266 (95% CI 0.1170885 to 0.9536551) and this is significant; p-value = 0.02893.

Linear regression

Now we can build our first linear regression model.

```
# Build a linear regression model
fit <- lm(PlasmaVolume_l~BodyWeight_kg,data=PlasmaData)
# Look at the model outputs
summary(fit)
```

```
##
## Call:
## lm(formula = PlasmaVolume_l ~ BodyWeight_kg, data = PlasmaData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.27880 -0.14178 -0.01928  0.13986  0.32939
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.08572    1.02400   0.084   0.9360
## BodyWeight_kg  0.04362    0.01527   2.857   0.0289 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2188 on 6 degrees of freedom
## Multiple R-squared:  0.5763, Adjusted R-squared:  0.5057
## F-statistic:  8.16 on 1 and 6 DF,  p-value: 0.02893
```

```
# Look at the 95% confidence intervals for the estimates
confint(fit)
```

```
##                      2.5 %     97.5 %
## (Intercept)   -2.419908594 2.59135716
## BodyWeight_kg  0.006255005 0.08097567
```

The regression co-efficient for body weight is 0.0436 l/kg i.e. for every 1kg increase in weight the plasma volume increases by 0.044 litres.

Note that the regression coefficients in linear regression have units. The size of the regression coefficients therefore depends on the units of the dependant and independant variables. If we convert the plasma volume measurements into mililitres (ie. x by 1000) the regression co-efficients will be 1000 times larger:

```
# Create a new variable with Plasma volume in ml
PlasmaData$PlasmaVolume_ml <- PlasmaData$PlasmaVolume_l * 1000

# Rerun the model with the new ml variable
fit <- lm(PlasmaVolume_ml~BodyWeight_kg,data=PlasmaData)
# Look at the model outputs
summary(fit)
```

```
##
## Call:
## lm(formula = PlasmaVolume_ml ~ BodyWeight_kg, data = PlasmaData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -278.80 -141.78  -19.28  139.86  329.39
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      85.72    1024.00   0.084   0.9360
## BodyWeight_kg    43.62      15.27   2.857   0.0289 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 218.8 on 6 degrees of freedom
## Multiple R-squared:  0.5763, Adjusted R-squared:  0.5057
## F-statistic:  8.16 on 1 and 6 DF,  p-value: 0.02893
```

```
# Look at the 95% confidence intervals for the estimates
confint(fit)
```

```
##                      2.5 %     97.5 %
## (Intercept)   -2419.908594 2591.35716
## BodyWeight_kg     6.255005   80.97567
```

Now our regression co-efficient for body weight is 43.6 ml/kg. The result of the regression is unaffected otherwise - the p value remains unchanged.

Well done you have performed your first regression analysis in R!