Session_09

Correlation analysis in R

First, let's create random numbers for a data set and explore the relationship between two randomly created variables. A random seed (in R `set.seed` function) is a number that initialises a pseudorandom number generator. Within the `set.seed` function, we simply have to specify a numeric value to set a seed.

Have a look at the following R code:

```
set.seed(35843)              # set random seed
x <- rnorm(100)              # create x variable
head(x)
```

```
## [1]  0.3863374 -0.8510980 -0.4309409 -0.3548813  0.1697038  1.8075671
```

Next, we have to create a second variable:

```
y <- rnorm(100) + x        # create and print the head of y variable
head(y)
```

```
## [1]  0.006255856 -1.072393886  0.766724567  1.025803298  0.749353490
## [6]  1.760101736
```

Let's create a data frame with these two variables and then use these data to calculate Pearson's correlation:

```
numbers <- data.frame(x,y)
head(numbers)
```

```
##            x           y
## 1  0.3863374  0.006255856
## 2 -0.8510980 -1.072393886
## 3 -0.4309409  0.766724567
## 4 -0.3548813  1.025803298
## 5  0.1697038  0.749353490
## 6  1.8075671  1.760101736
```

```
corr.1 <- round(cor(x, y), 2)   # Pearson correlation + round
corr.1                          # the output to two decimal places
```

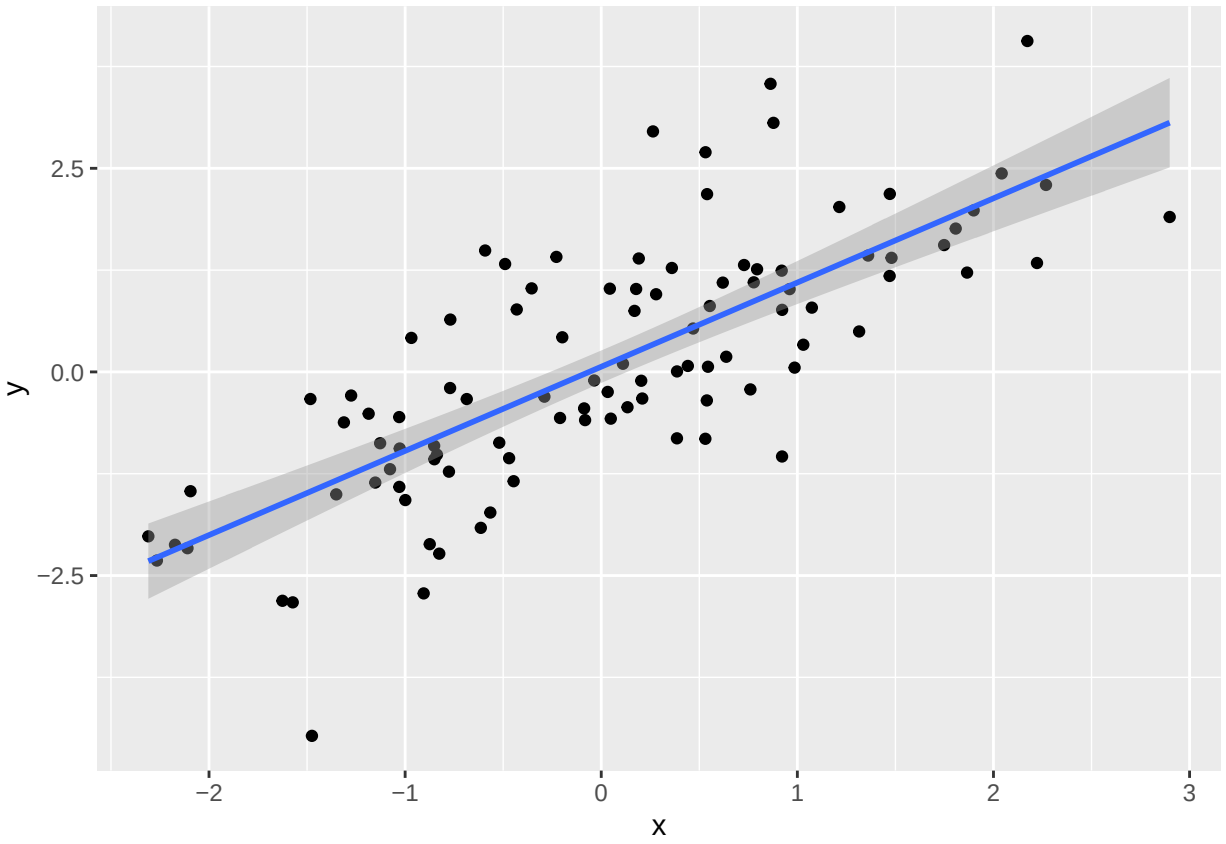
```
## [1] 0.76
```

The output is *0.76*, this is our Pearson correlation coefficient. Since the number is positive, it's a **positive** correlation, i.e. our two variables move in the same direction and when one variable increases, the other one increases as well. The number *0.76* indicates that it is a **strong** correlation.

Let's visualise our variables. Scatterplots are a great way to check quickly for correlation between pairs of *continuous data*.

```
# Basic scatter plot
library(ggplot2)
ggplot(numbers, aes(x=x, y=y)) + geom_point() +
  geom_smooth(method=lm)    # adding a line
```

```
## `geom_smooth()` using formula 'y ~ x'
```



We can use the `cor` function to calculate a correlation matrix for an entire data frame with several variables:

```
set.seed(12345)         # set seed and create example data
data <- data.frame(x1 = rnorm(100),
                   x2 = rnorm(100),
                   x3 = rnorm(100),
                   x4 = rnorm(100))
head(data)         # print head of example data
```

```
##           x1         x2         x3          x4
## 1  0.5855288  0.2239254 -1.4361457  0.52228217
## 2  0.7094660 -1.1562233 -0.6292596  0.00979376
## 3 -0.1093033  0.4224185  0.2435218 -0.44052620
## 4 -0.4534972 -1.3247553  1.0583622  1.19948953
## 5  0.6058875  0.1410843  0.8313488 -0.11746849
## 6 -1.8179560 -0.5360480  0.1052118  0.03820979
```

In this example, we will see how to get Pearson correlation coefficient between a particular data frame variable with all the other variables in this data frame.

To achieve this, we can apply the `cor` and `colnames` functions as shown below:

```
data_cor <- cor(data[ , colnames(data) != "x1"], data$x1) # calculate correlation
data_cor <- round(data_cor,2)  # round the output to 2 decimal places
data_cor
```

```
##     [,1]
## x2  0.10
## x3 -0.04
## x4  0.14
```

Next, we can use the `cor` function to create a correlation matrix for for all our variables:

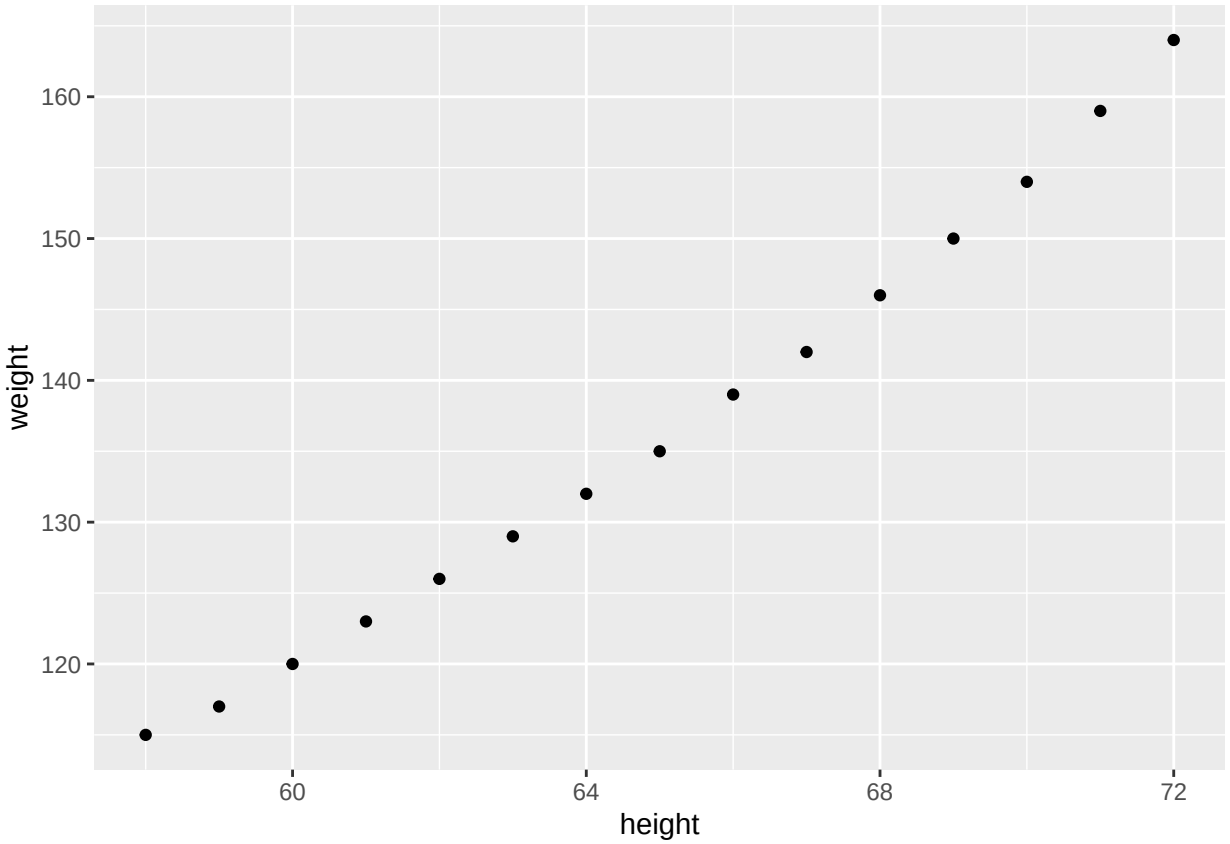
```
data.corr.1 <- round(cor(data),2) # Pearson correlation + round the output
data.corr.1
```

```
##       x1    x2    x3    x4
## x1  1.00  0.10 -0.04  0.14
## x2  0.10  1.00 -0.13  0.15
## x3 -0.04 -0.13  1.00 -0.29
## x4  0.14  0.15 -0.29  1.00
```

Correlations of `height` and `weight`

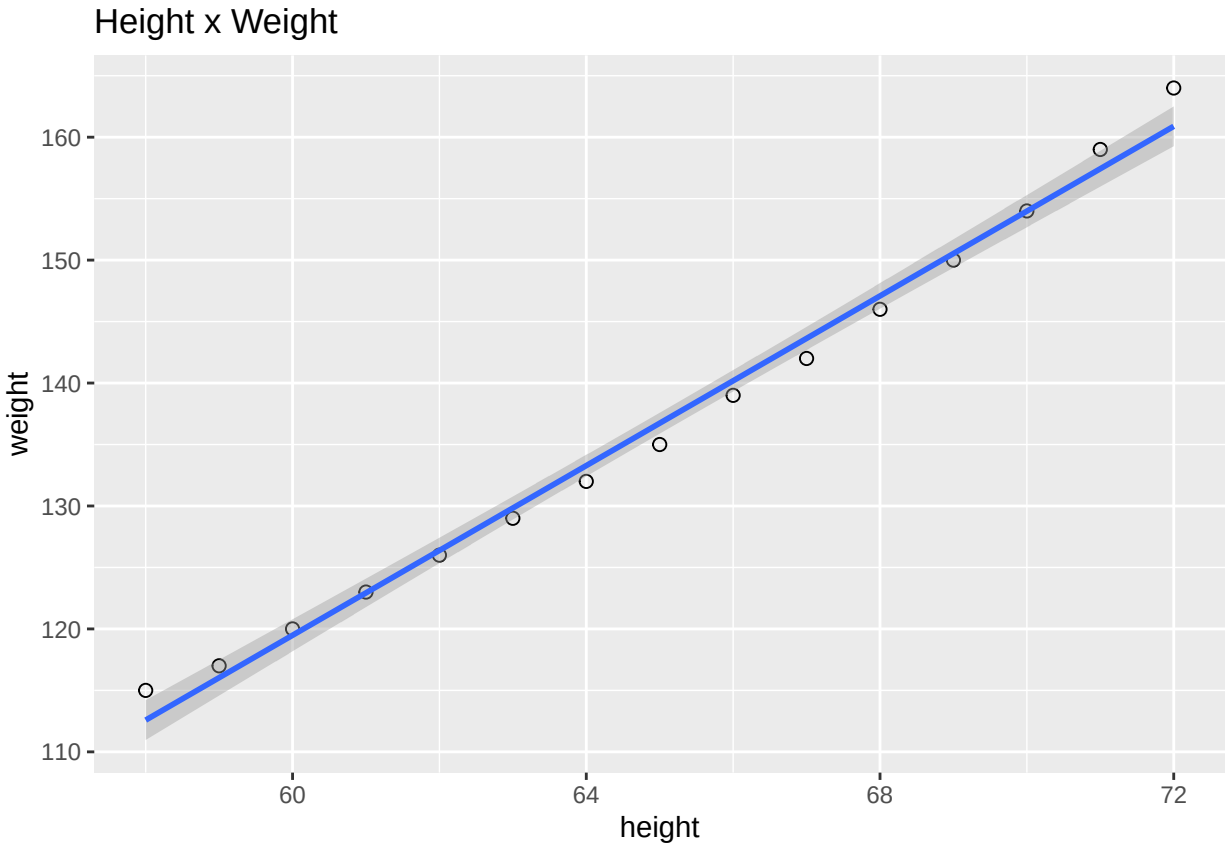
Let's look at one more data set.
Download the `women` data set and explore the correlation between `weight` and `height` by creating a scatterplot:

```
data(women)
# Basic scatter plot
ggplot(women, aes(x=height, y=weight)) + geom_point()
```




```
ggplot(women, aes(x=height, y=weight)) +
  geom_point(size=2, shape=21) +   # changing the dots size and shape
  geom_smooth(method=lm) +         # adding the line
  labs(title = "Height x Weight")  # adding the title
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Let's calculate Pearson correlation coefficient:

```
corr.2 <- cor(women$height, women$weight, method = 'pearson')
corr.2 <- round(corr.2, 2) # rounding the number to two decimals
corr.2
```

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

How can you interpret this result? Any ideas why the result is like that?