# Practicals 4

January Weiner

5/6/2021

## Session 1: Introduction to R

```
# crap. It did not save it, grr
## (my oldschool editor saves automatically, so I don't think about it)

#### asdff asdf asdff ####

5 + 7
```

```
## [1] 12
```

```
a <- 5 + 7
5 ** 7 # obsolete, but works
```

```
## [1] 78125
```

```
5^7 # same as 5 ** 7
```

```
## [1] 78125
```

```
## define function blah
#' @param par1 first parameter. Does absolutely nothing
## By the way, this previous line is a formalized way of documenting functions.
## This comes with the roxygen2 package and is used to document packages.
myfunc <- function(par1) {

  ## asdfasdf


}
```

Here, we play with operators.

```
## what does the following do?
5 %/% 7
```

```
## [1] 0
```

```
5 / 7
```

```
## [1] 0.7142857
```

```
5 %% 7
```

```
## [1] 5
```

```
## to get help on operators, you need to use quotes
?"%*%"
```

```
?"%%" # what is this operator
```

So, what do these operators do?

And now for vectorization. Vectorization is really important: everything in R is either a vector, or a more complex data type. Vectors can have length of 1, or even 0 (yes, that is possible)

```
## everything is a vector!
v <- c(1, 2, 3)
is.vector(v)
```

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

```
v <- TRUE
is.vector(v)
```

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

```
v <- 1
is.vector(v)
```

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

```
## vectorization, you need to
c(1, 2, 3) * 5
```

```
## [1]  5 10 15
```

```
c(1, 2, 3) * c(2, 5) # error message!
```

```
## Warning in c(1, 2, 3) * c(2, 5): longer object length is not a multiple of shorter object length
```

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

```
c(1, 2, 3) * c(1, 5, 10)
```

```
## [1]  1 10 30
```

```
## scalar product
c(1, 2, 3) %*% c(1, 5, 10)
```

```
##      [,1]
## [1,]   41
```

## First statistical tests

We define two variables:

$$x \sim f_N(\mu, \sigma)$$

with $\mu = 0$ and $\sigma = 0$, as well as

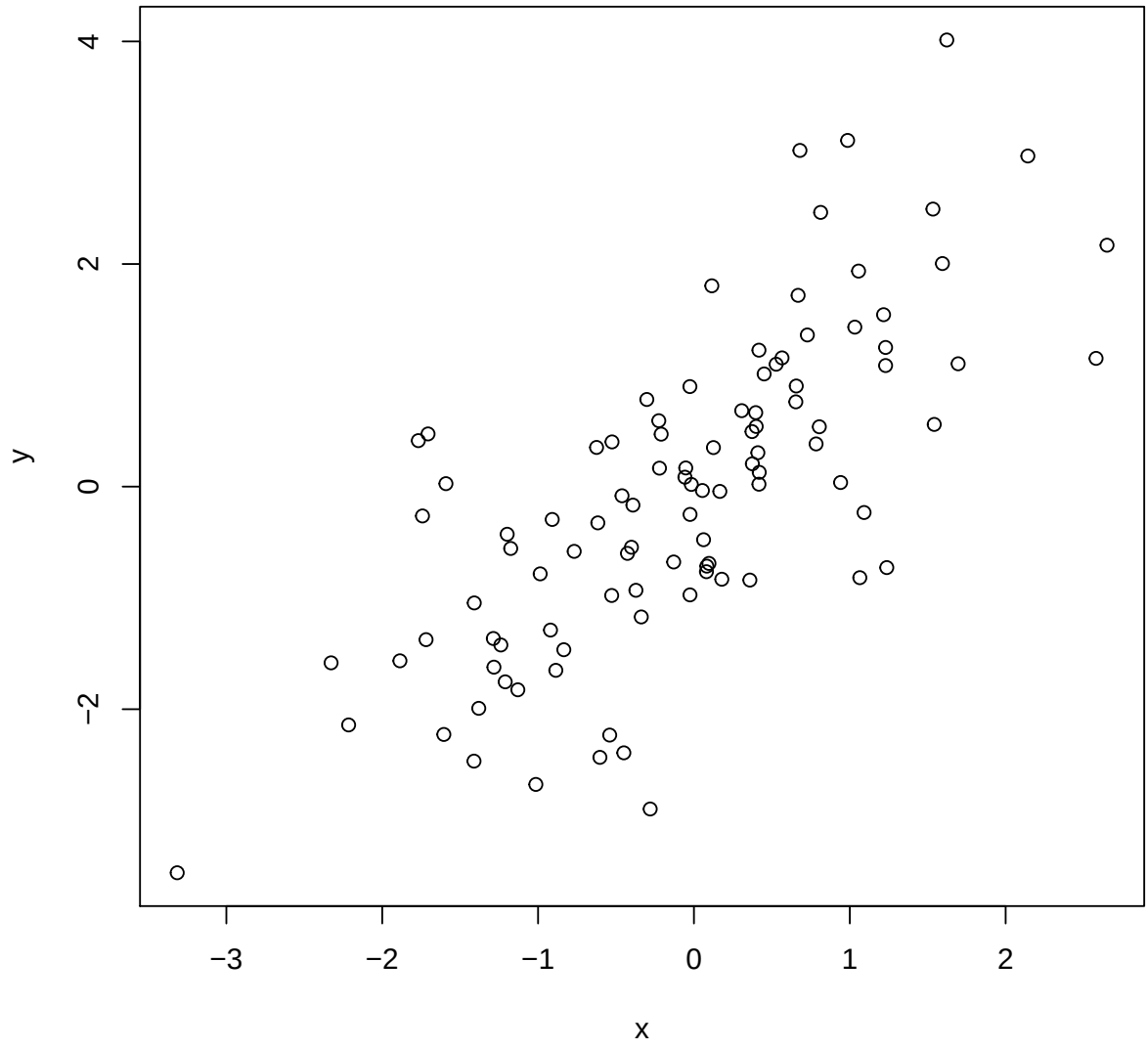
$$y = x + \epsilon$$

where

$$\epsilon \sim f_N(\mu, \sigma)$$

And now, for some plotting. The `fig.height` and `fig.width` parameters for the chunk determine the width and height of the resulting figure.

```
## get random numbers and plot them
x <- rnorm(100)
y <- x + rnorm(100)
plot(x, y)
```




```
##
cor(x, y)
```

```
## [1] 0.727293
```

```
cor(x, y, method="s")
```

```
## [1] 0.739574
```

```r
cor.test(x, y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = 10.49, df = 98, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6193514 0.8082498
## sample estimates:
##      cor
## 0.727293
```

```r
cor.test(x, y, method = "s")
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 43400, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.739574
```