STU22005 Laboratory Session 1

Professor Caroline Brophy

Week beginning 8th February 2021

Instructions

- In this lab sheet, there will be a series of notes for you to work through. For each section, you should run the R code yourself on your own computer and verify the output provided. Ask your laboratory demonstrator any questions you have as you work through the material.
- In the last section, there are several questions where you will write your own code. Your laboratory demonstrator is there to help if you get stuck in this section.
- Create a script for all the code that you write from this laboratory. Create a new folder on your computer for this module and save the script into the folder. Save your code regularly throughout the class.

Refresher on the basics of R

Open R or R Studio on your computer.

Reading a vector of data into R and exploring the data

Read the following vector of data values into R. This code stores a vector of data values in `x`. Remember that R is case sensitive.

```
# Store the data values in a vector x
x <- c(4.5, 6.5, 6.4, 8.9, 4.1, 6.4, 6.3, 9.1, 12.1, 1.4, 1.4, 4.6, 1.6, 9.8, 7.2, 6.5,
       4.1, 6.5, 11.6, 2.9)
```

View the vector in the output window.

```
# View x
x
```

```
##  [1]  4.5  6.5  6.4  8.9  4.1  6.4  6.3  9.1 12.1  1.4  1.4  4.6  1.6  9.8  7.2
## [16]  6.5  4.1  6.5 11.6  2.9
```

Find the summary statistics for the vector of values

```
# Find the average
mean(x)
```

```
## [1] 6.095
```

```r
# Find the standard deviation
sd(x)
```

```
## [1] 3.130239
```

```r
# Find the number of observations
length(x)
```

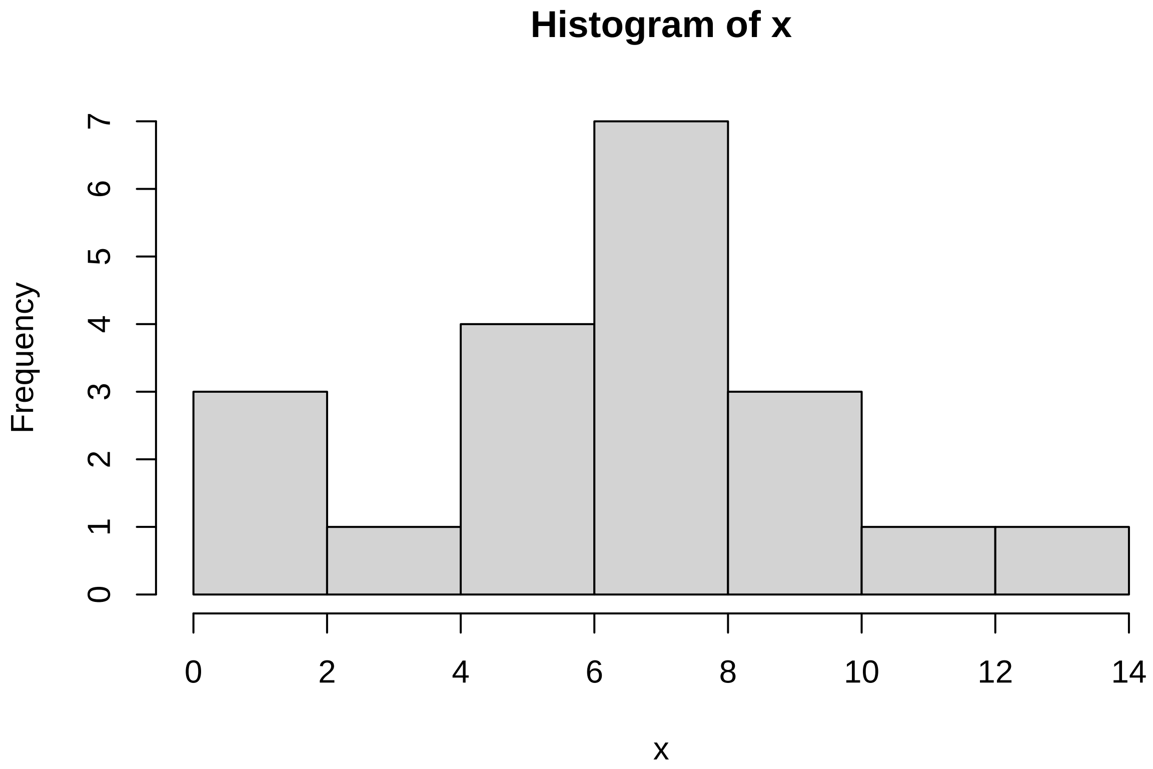
```
## [1] 20
```

```r
# Find various summary statistics
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.400   4.100   6.400   6.095   7.625  12.100
```

Plot the data values in a histogram.

```r
# Generate a histogram of x
hist(x)
```



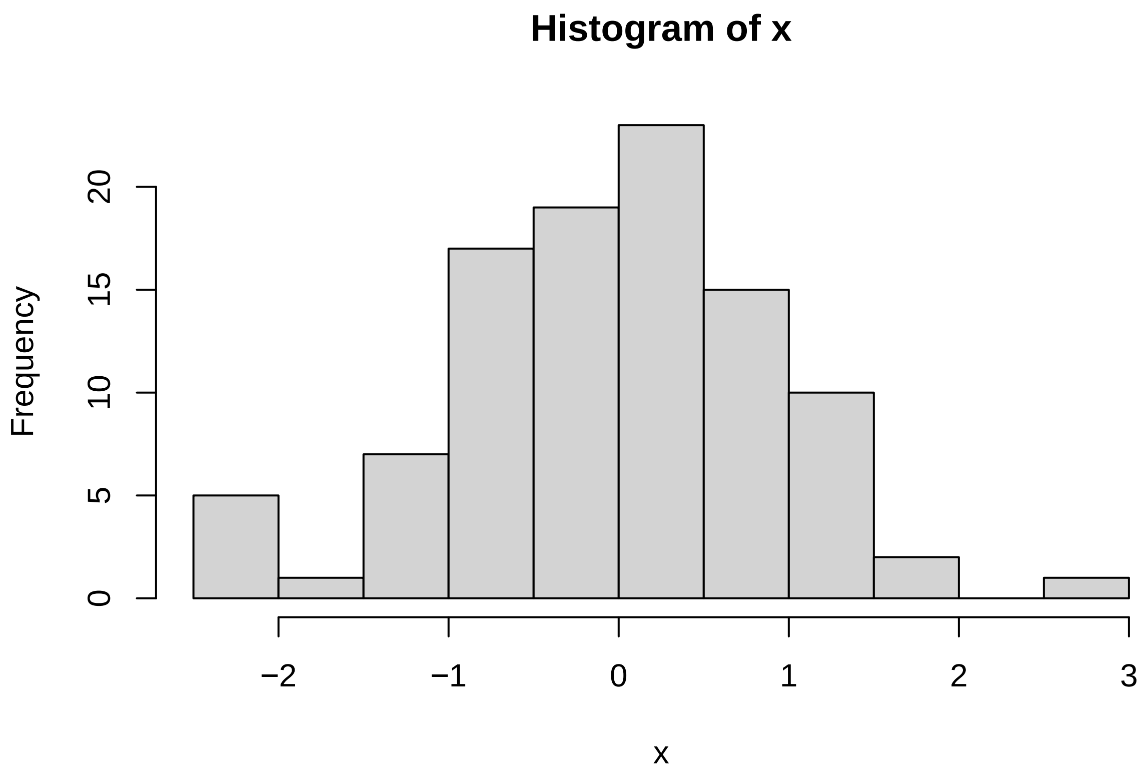
Simulating data

Simulating data from many different distributions can be carried out easily in R. The following code will generate 100 independent realisations from a N(0,1) and store these in the vector x. Make your output reproducible by using the `set.seed()` function.

```r
# Set a seed for reproducibility
set.seed(6124)
# Store the value of a scalar in n
n <- 100
# Generate a vector of independent standard normal variables.
x <- rnorm(n)
```

Construct a histogram of the values.

```r
# Generate a histogram of x
hist(x)
```



To find out more about how to generate data from the normal distribution, run the following code.

```r
?rnorm
```

In general, if you want to find out more about an R function you run ?function_name.

Reading a dataset into R

The dataset “Lab1a.csv” is on Blackboard with the lab sheet for this session. Save this file into the folder you created for the module. Set the working directory for this R session by using the code, for example: `setwd("C:/STU22005")`.

Read the dataset into R.

```
# Read the data into R
data1 <- read.csv("Lab1a.csv")
data1
```

```
##    var1
## 1     3
## 2     6
## 3     3
## 4     2
## 5     5
## 6     6
## 7     8
## 8     1
## 9     2
## 10    9
```

You have now created a data frame called `data1` which includes the variable `var1`. To access `var1`, we use `data1$var1`. For example, to find the mean of `var1`:

```
# mean
mean(data1$var1)
```

```
## [1] 4.5
```

Putting your code into practice

For the remainder of this laboratory session: DO NOT USE COPY / PASTE. Write out your own code. Try to remember the code yourself, but you can look back at the earlier code whenever you need to. Typing out your own code gives you a better chance of understanding what each part of the code does, and will help your R programming skills to develop.

It is very good practice to include good comments for each part of your R code. Please get into the habit of this every time you write R code.

Questions

1. In Section 2.3 of the lecture notes, we looked at a Pharmaceutical example. Go back to the lecture notes on Blackboard and find these values. Read the data values into a vector in R. Construct a histogram of the data and find the summary statistics.
2. Simulate 1000 values from a standard normal distribution and construct a histogram for the values. Use a different seed to before.
3. Simulate 200 values from a normal distribution with $\mu = 30$ and $\sigma^2 = 9$. Hint: view the help files for `rnorm`. Use a different seed to before.
4. Repeat the previous step for 20,000 values. What do you notice when you compare the two histograms?
5. The dataset "Lab1b.csv" is stored on Blackboard with the lab sheet for this session. Save this dataset into your module folder and read it into R. Hint: to view the first five lines of the dataset, you can run the code `head(nameofdataset)`. Compute summary statistics and generate histograms for each variable. How would you describe the distribution of each variable?
6. For the histograms you generated so far in this lab session, go back and improve the "look" of them. For example, choose what title to put on, choose appropriate labels for axes, and modify the size of the text where necessary. Hint: you can run `?hist` to find out more argument options for this function.