Session1:
Getting started in R. A whirlwind introduction.

Dr Nicola Foster

UCL Respiratory

nicola.foster@ucl.ac.uk

https://intro2r.com

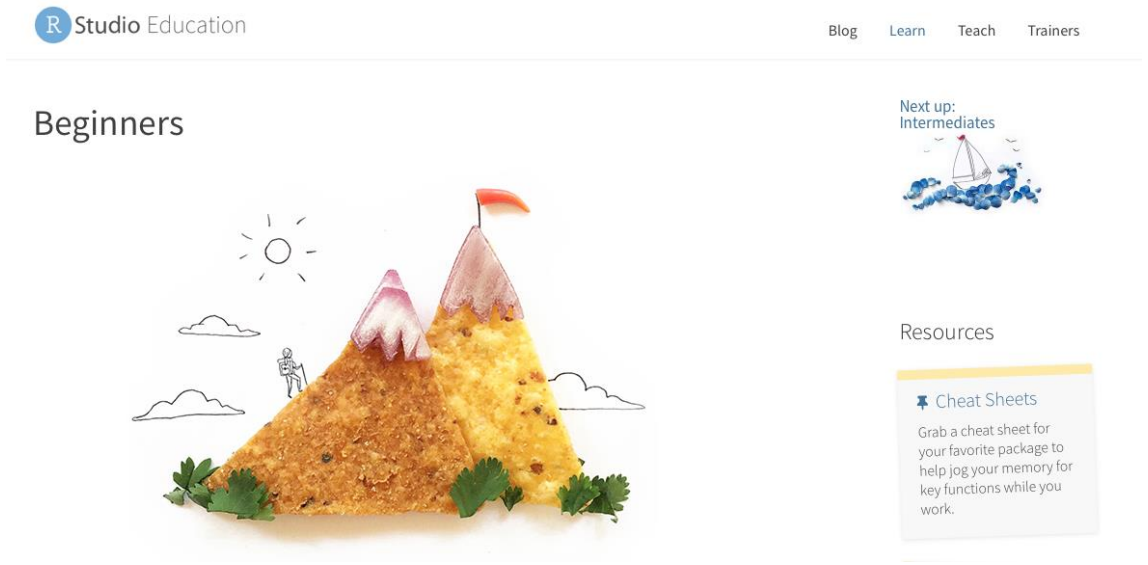
Sep	Getting started in R	Data science introduction session with a focus on setting up to code in R, showing people how to import datasets, summarise data descriptively and where to find help.
Dec	Understanding data	Introduction to statistical thinking, types of data and how data types influence the methods we use, data management while doing the analysis, and data processing tips.
Mar	Uncovering differences between groups	Introduction to regression analyses, identifying covariates for your models, DAGs, correlation and confounding.
Jun	Setting up the experiment	Study designs and sample size calculations. This focus will be on what information we need for doing sample size calculations and looking at some examples. I will also introduce the approach of simulation studies at the study design phase and why they are valuable.

Introduction

- Expressing your analysis in code is only one part of your analysis
  - Still need to think about your experiment
  - Need a good design
  - And able to interpret your findings
- Using a programming language such as R
  - Allow you/ others to reproduce your analysis
  - Easier to review what you've done
  - Saves you time especially on repetitive tasks
  - Allow you to do computationally intensive tasks

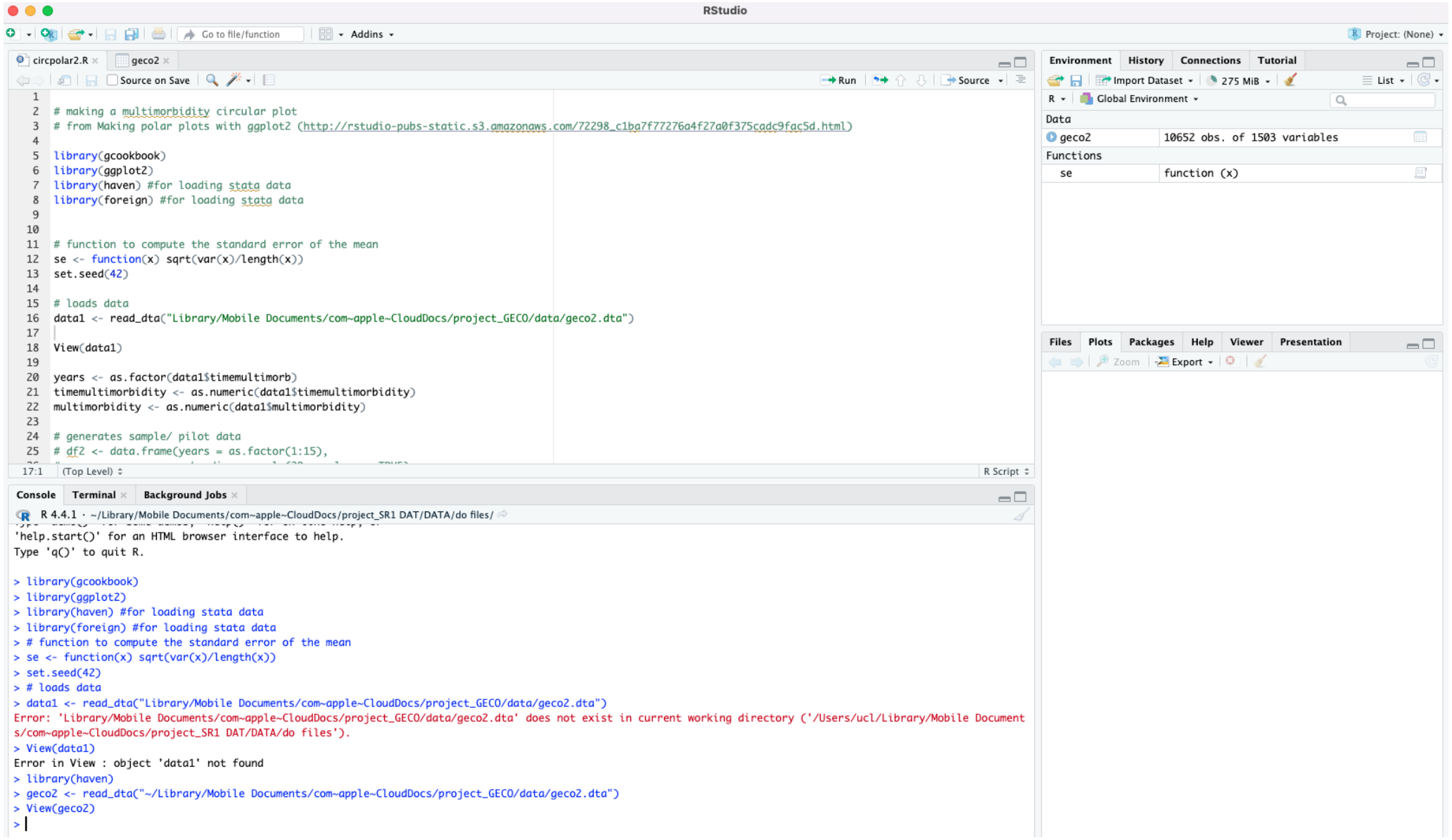
Installing R

- R is free and open-source software, download from Comprehensive R Archive Network (CRAN) website https://cran.r-project.org
- Also need an integrated development environment (IDE) to run base R – use R studio



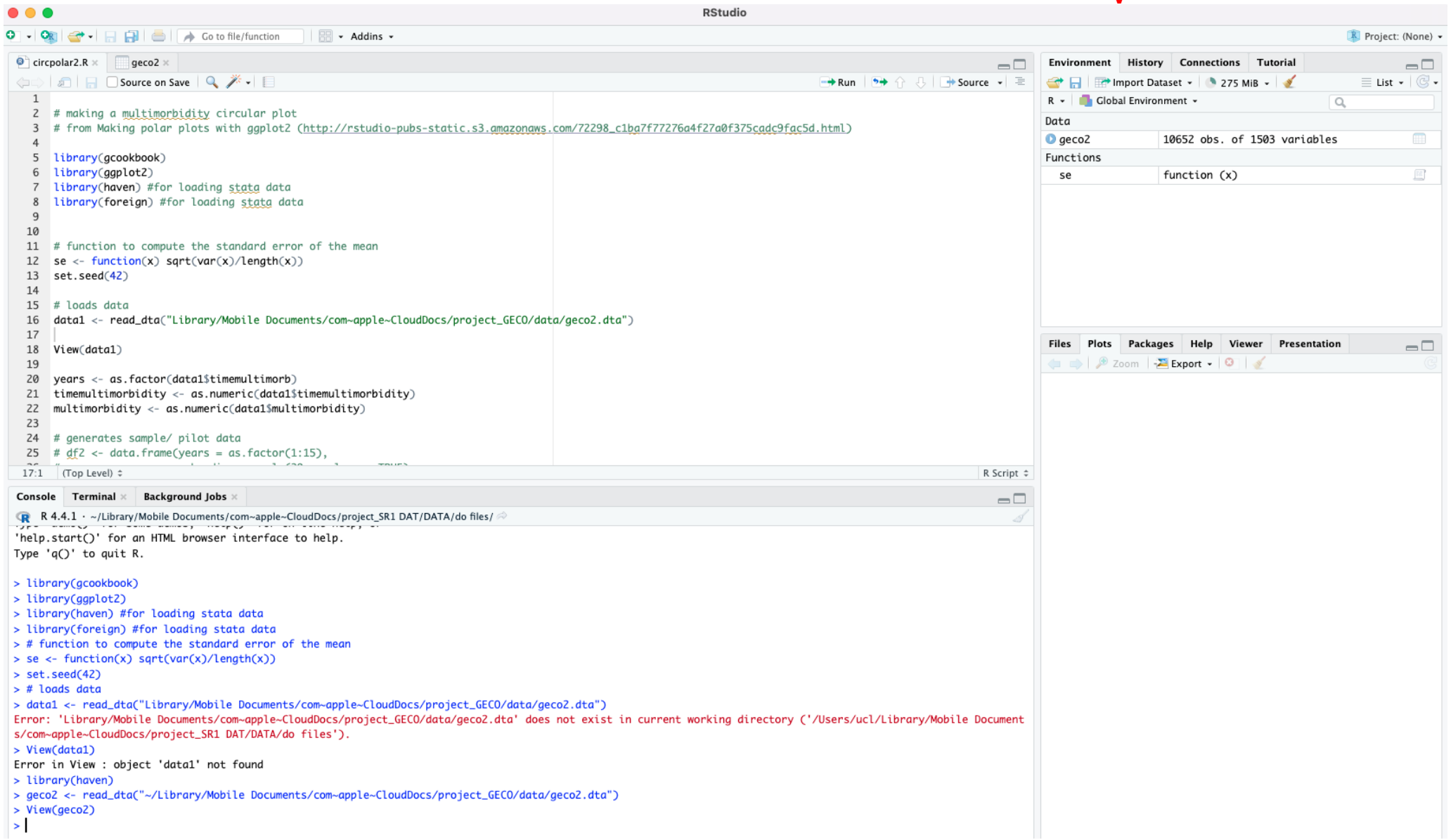
The R environment 1

Script ->

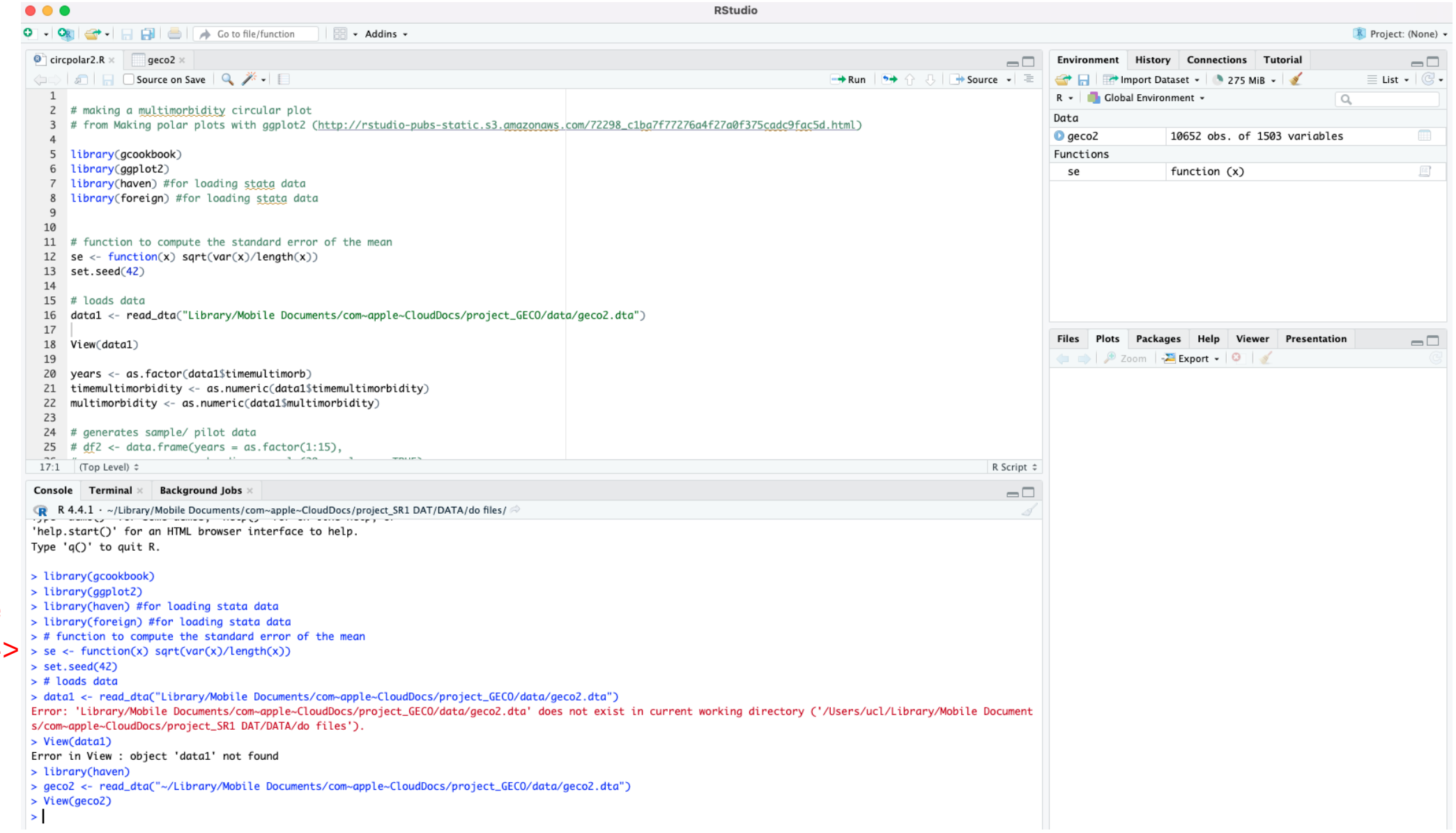


The R environment 2

Objects
|
V

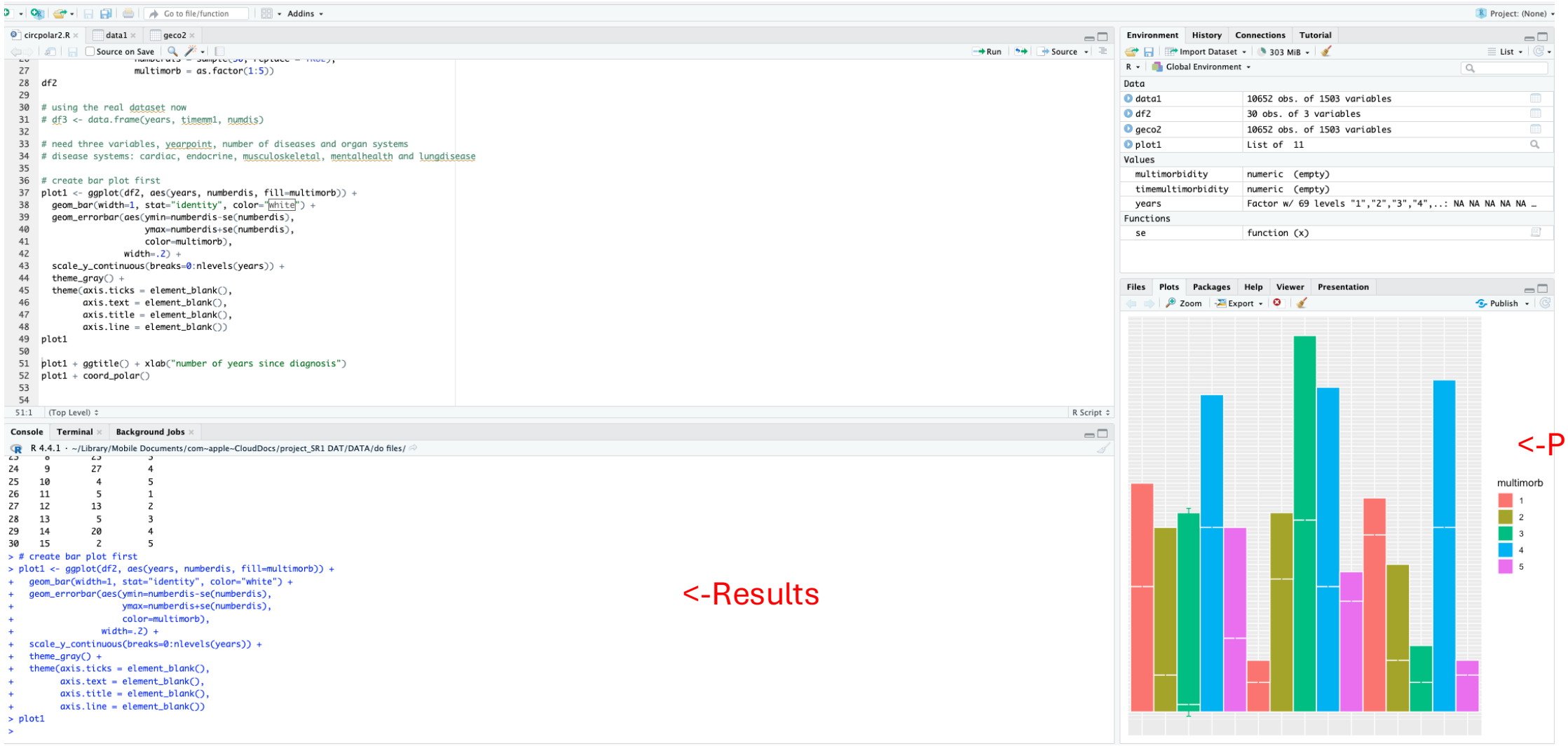


The R environment 3



Console/ code implemented ->

The R environment 4



<-Results

<-Plots

Packages and Objects

- Packages are collections of R functions, data and compiled code that you will use to perform certain tasks.
- Base R refers to the standard set of packages, others you can download, install and load into your library.

```
> install.packages("ggplot2")
```

```
> library(ggplot2)
```

- All things in R – functions (equations), datasets (vectors, matrices, lists, and data frames), results are objects
- Script is the way to make objects. Can be stored and reviewed.

Functions

```r
# function to compute the standard error of the mean
se <- function(x) sqrt(var(x)/length(x))
```

- Equations
- Instead of writing a calculation over and over – write a repetitive calculation as a function then apply

Variables

```
years <- as.factor(data1$timemultimorb)
timemultimorbidity <- as.numeric(data1$timemultimorbidity)
multimorbidity <- as.numeric(data1$multimorbidity)
```

- Important for many functions that you specify your variables as numerical or factors
- Factors are categorical variables in R

Vectors

```
> count <- c(1, 3, 4, 8, 0, 2)
```

- Use vectors to summarise, manipulate and sort data using R
- Can combine multiple logical expressions using Boolean expressions
- Change values of some elements in a vector

Data frames

```r
# generates sample/ pilot data
df2 <- data.frame(years = as.factor(1:15),
                  numberdis = sample(30, replace = TRUE),
                  multimorb = as.factor(1:5))
```

- most commonly used data structure to store data in is the data frame
- Two-dimensional object made of rows and columns, whereby each row corresponds to an individual observation
- Other ways of storing data include matrices, arrays and lists

Working with data

- Does happiness increase with increase in age?

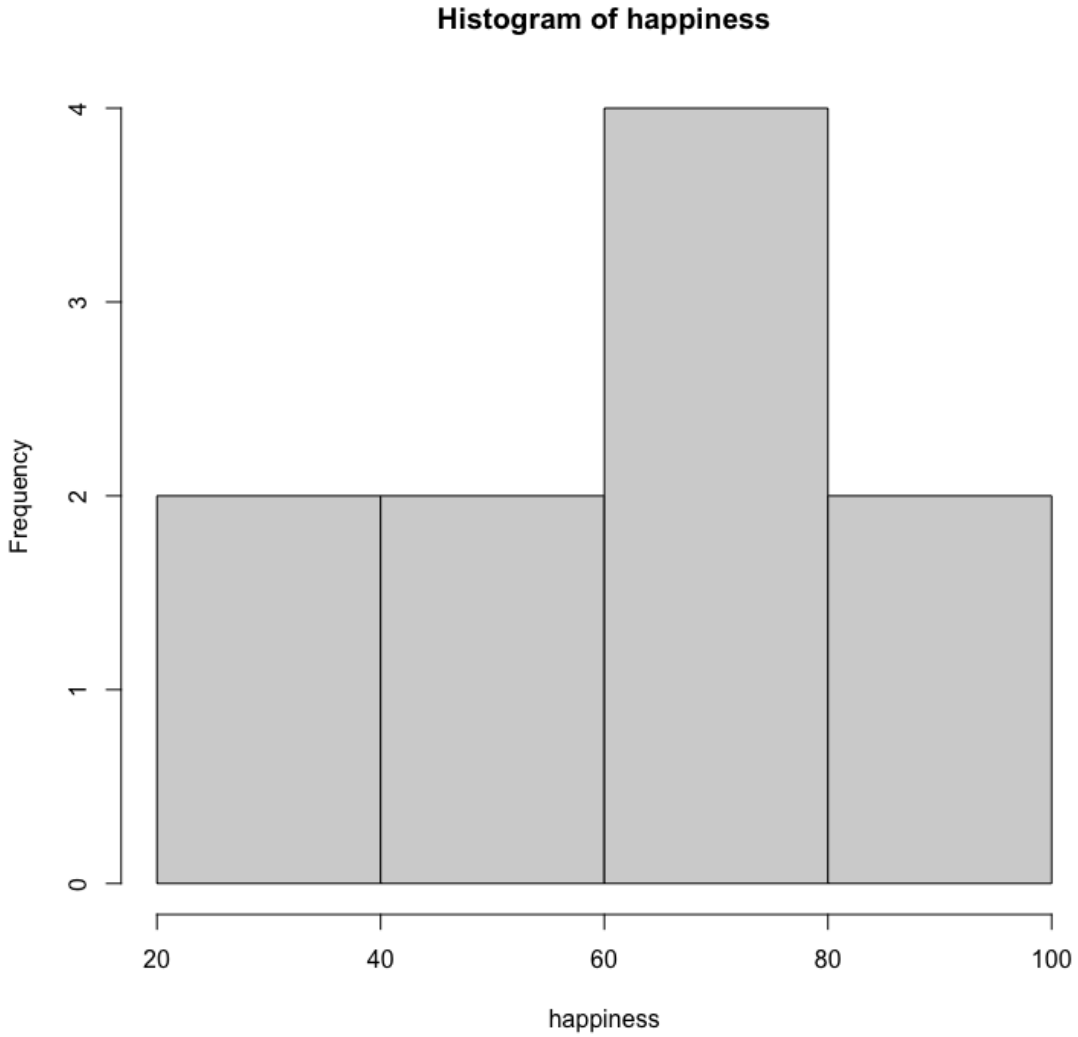
```
# set the working directory
setwd("/Users/ucl/Documents/")

library(readxl)

# imports data
data1 <- read_excel("~/Desktop/happiness.xlsx")
View(data1)

ages <- as.numeric(data1$ages)
gender <- as.factor(data1$gender)
happiness <- as.numeric(data1$happiness)
```

https://statsandr.com/blog/descriptive-statistics-in-r/

Summarising data

```
> #summarising data
> head(data1) #first 6 observations
# A tibble: 6 x 4
     id  ages gender happiness
  <dbl> <dbl> <chr>      <dbl>
1     1    26 male          50
2     2    30 female       100
3     3    38 female        80
4     4    43 male          20
5     5    17 female        30
6     6    78 male         100
> mean(ages) # average age
[1] 41.8
> sd(ages) # standard deviation of age
[1] 27.94359
> range(ages) #range of ages
[1] 10 98
> summary(data1) #statistics for all numeric variables in dataset
       id             ages          gender            happiness
 Min.   : 1.00   Min.   :10.00   Length:10          Min.   : 20.0
 1st Qu.: 3.25   1st Qu.:23.75   Class :character   1st Qu.: 52.5
 Median : 5.50   Median :34.00   Mode  :character   Median : 80.0
 Mean   : 5.50   Mean   :41.80                      Mean   : 68.0
 3rd Qu.: 7.75   3rd Qu.:52.00                      3rd Qu.: 80.0
 Max.   :10.00   Max.   :98.00                      Max.   :100.0
> table(gender)
gender
female   male
     6      4
> by(data1, gender, summary) #statistics by group
gender: female
       id             ages          gender            happiness
 Min.   : 2.000   Min.   :10.00   Length:6           Min.   : 30.00
 1st Qu.: 3.500   1st Qu.:18.50   Class :character   1st Qu.: 65.00
 Median : 6.500   Median :26.50   Mode  :character   Median : 80.00
 Mean   : 6.167   Mean   :28.83                      Mean   : 71.67
 3rd Qu.: 8.750   3rd Qu.:36.00                      3rd Qu.: 80.00
 Max.   :10.000   Max.   :55.00                      Max.   :100.00
------------------------------------------------------------------------
gender: male
       id            ages          gender            happiness
 Min.   :1.00   Min.   :26.00   Length:4           Min.   : 20.0
 1st Qu.:3.25   1st Qu.:38.75   Class :character   1st Qu.: 42.5
 Median :5.00   Median :60.50   Mode  :character   Median : 65.0
 Mean   :4.50   Mean   :61.25                      Mean   : 62.5
 3rd Qu.:6.25   3rd Qu.:83.00                      3rd Qu.: 85.0
 Max.   :7.00   Max.   :98.00                      Max.   :100.0
```

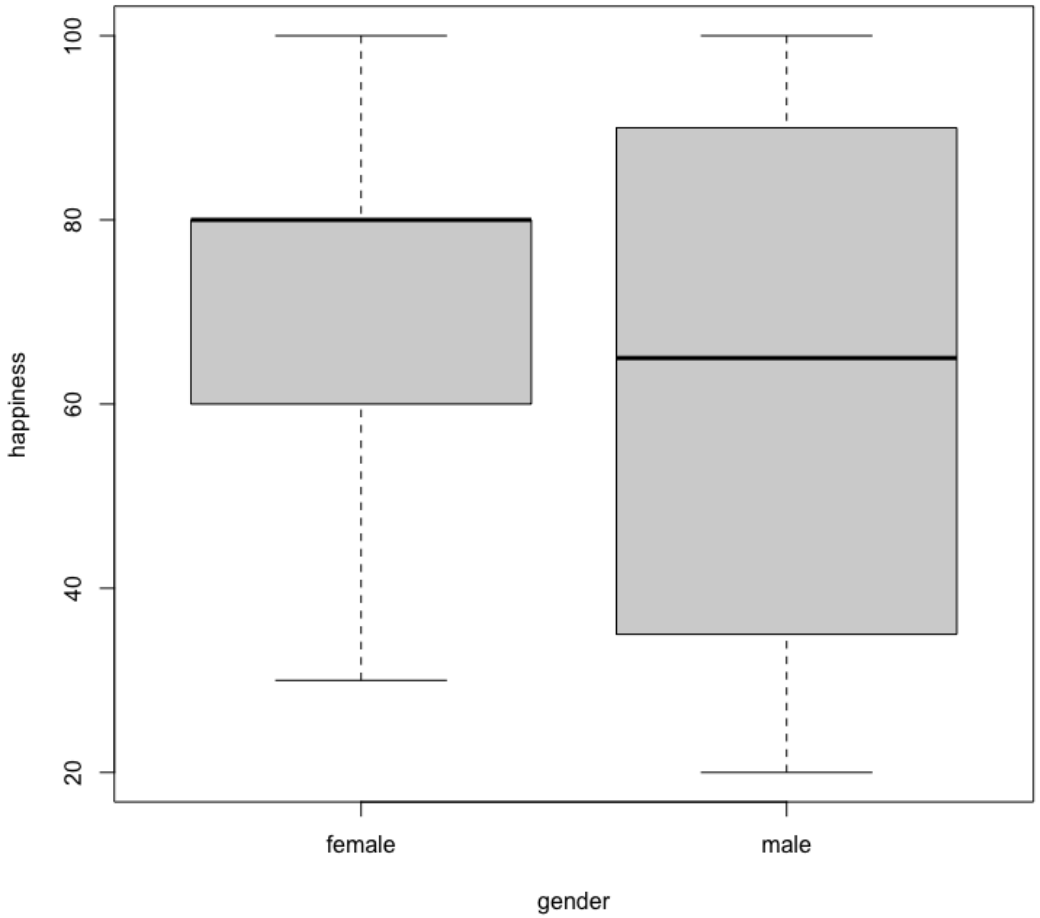
Plotting 1



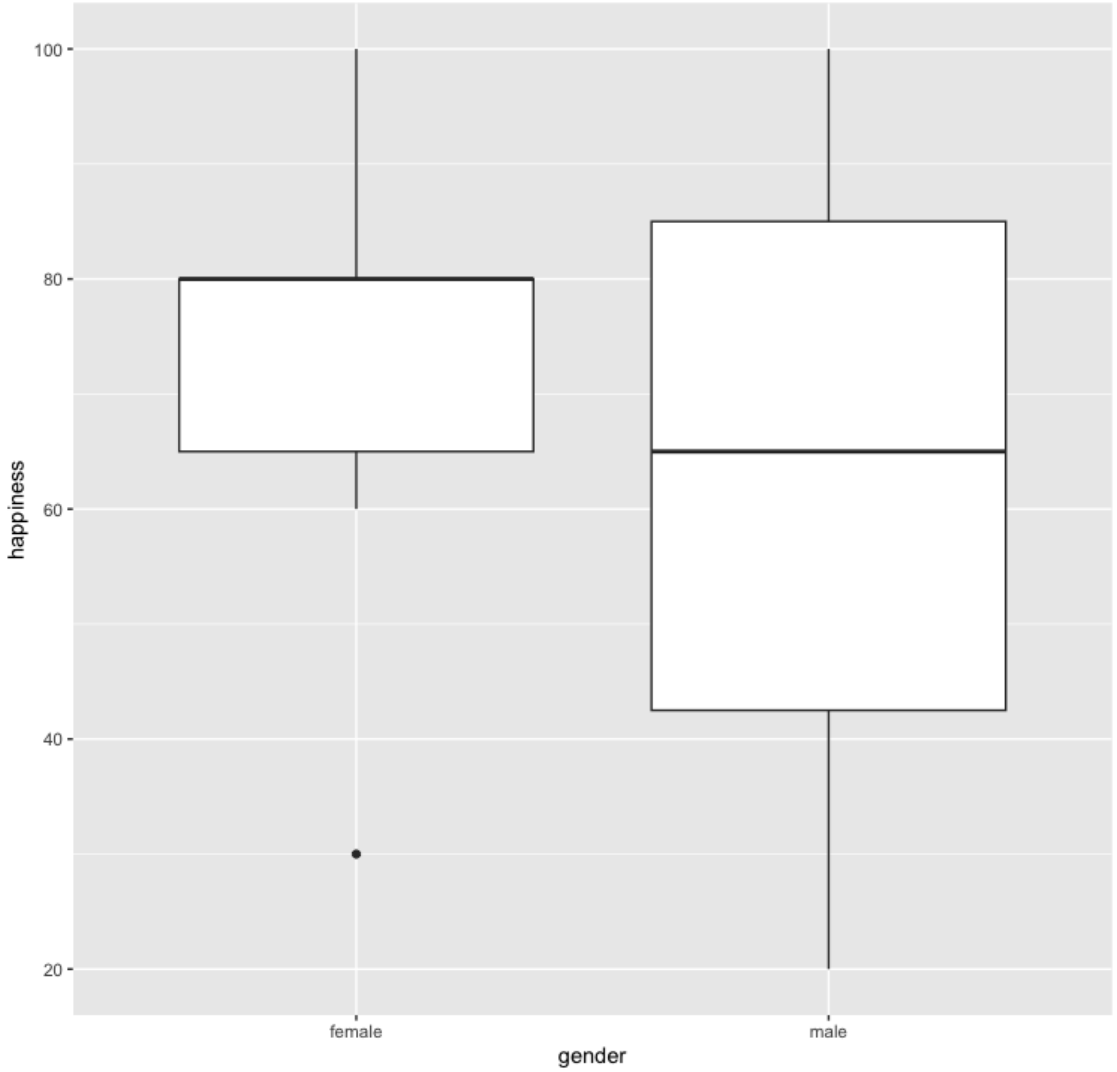

```r
#plotting
hist(happiness)
boxplot(happiness~gender)

# same graph plotted using ggplot2
ggplot(data1) +
  aes(x=gender, y=happiness) +
  geom_boxplot()
```

Plotting 2

```r
#plotting
hist(happiness)
boxplot(happiness~gender)

# same graph plotted using ggplot2
ggplot(data1) +
  aes(x=gender, y=happiness) +
  geom_boxplot()
```

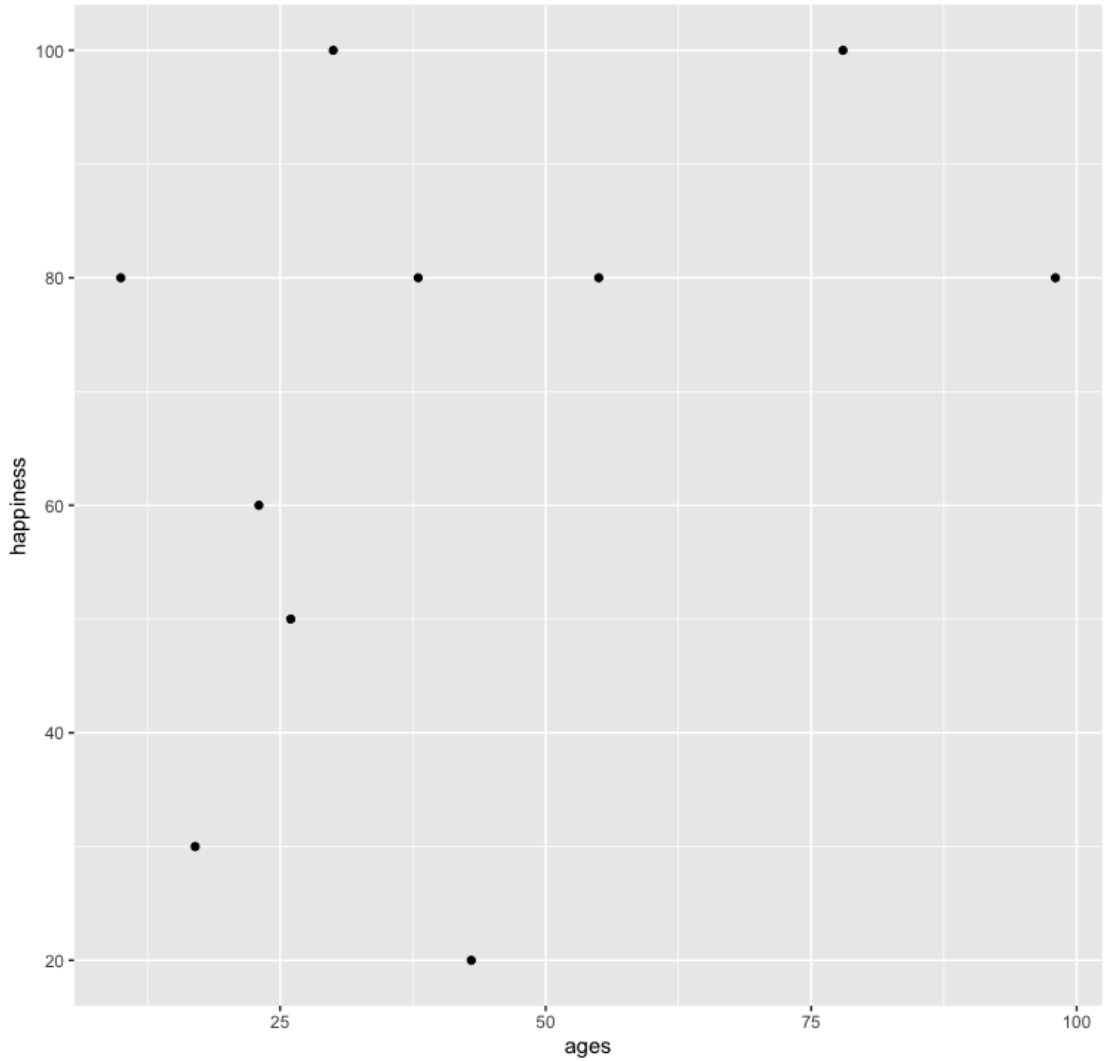


Plotting 3 (ggplot)




```r
#plotting
hist(happiness)
boxplot(happiness~gender)

# same graph plotted using ggplot2
ggplot(data1) +
  aes(x=gender, y=happiness) +
  geom_boxplot()
```

Linear regression model

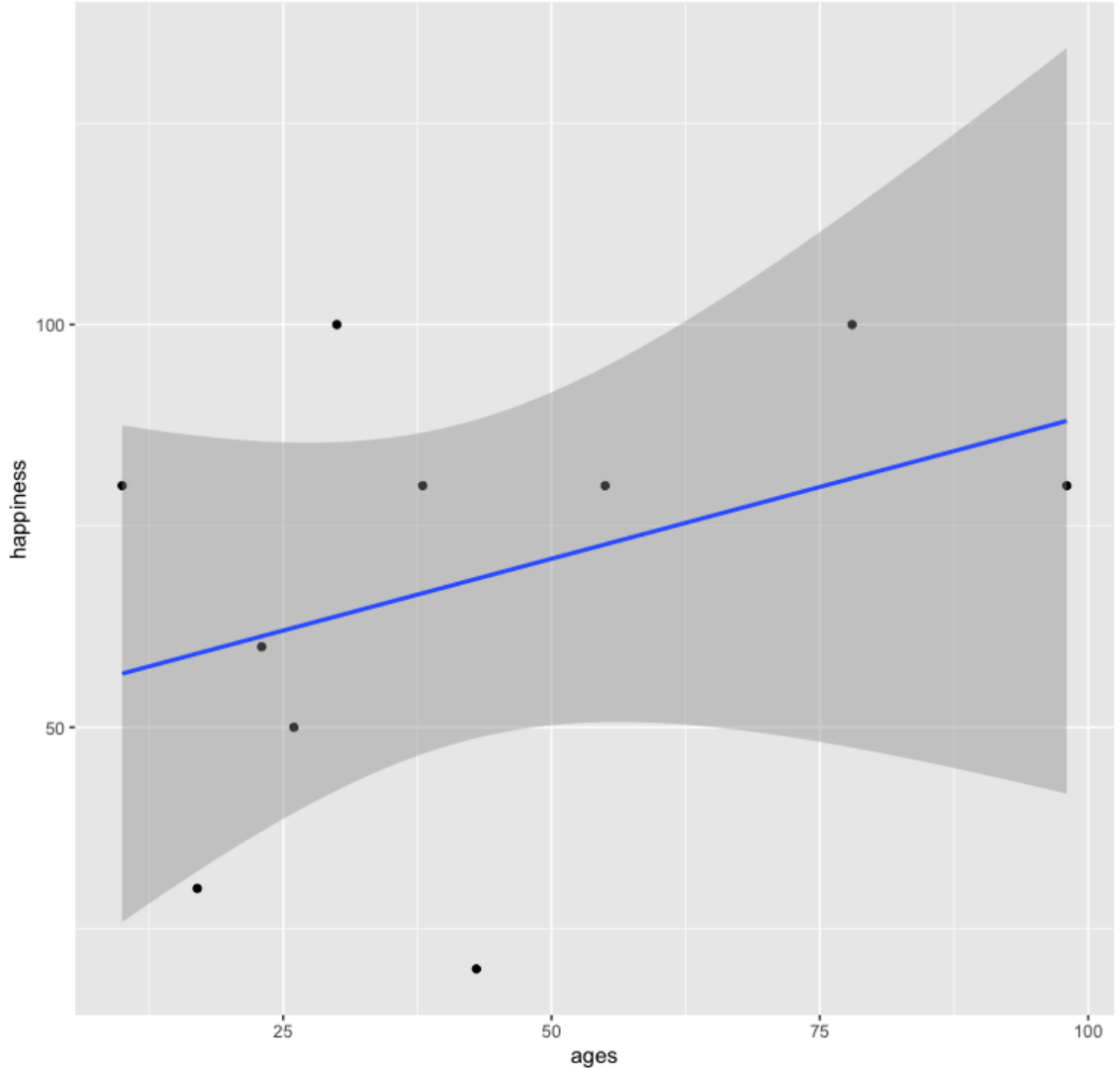



```r
#linear regression analysis
#response variable ~ explanatory variable(s)
ggplot(mapping = aes(x=ages, y=happiness), data=data1) +
  geom_point()

lm1 <- lm(happiness ~ gender+ages, data=data1)
lm1
summary(lm1)
```

```
> lm1 <- lm(happiness ~ gender+ages, data=data1)
> lm1

Call:
lm(formula = happiness ~ gender + ages, data = data1)

Coefficients:
(Intercept)   gendermale         ages
    51.0657     -32.3279       0.7145

> summary(lm1)

Call:
lm(formula = happiness ~ gender + ages, data = data1)

Residuals:
    Min      1Q  Median      3Q     Max
-33.212  -9.961  -2.857  19.513  27.500

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  51.0657    14.6163   3.494   0.0101 *
gendermale  -32.3279    19.8834  -1.626   0.1480
ages          0.7145     0.3674   1.944   0.0929 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.66 on 7 degrees of freedom
Multiple R-squared:  0.3701,    Adjusted R-squared:  0.1901
F-statistic: 2.056 on 2 and 7 DF,  p-value: 0.1984
```

Linear regression model

```
ggplot(mapping = aes(x=ages, y=happiness), data=data1) +
  geom_point() +
  geom_smooth(method="lm", se=TRUE)
```



Resources

- An introduction to R and good place to start https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf
- Introduction to R with focus on basic data analysis http://r4ds.had.co.nz/
- Advanced R where the focus is more on the programming parts https://adv-r.hadley.nz/
- Imputation https://stefvanbuuren.name/fimd/
- Data visualization using ggplot2 http://moderngraphics11.pbworks.com/f/ggplot2-Book09hWickham.pdf

UCL courses

- Introduction to R https://www.ucl.ac.uk/short-courses/search-courses/introduction-r-online
- Regressions with R: an overview https://www.ucl.ac.uk/short-courses/search-courses/regressions-r-overview
- Introduction to tidyverse and ggplot2 https://www.ucl.ac.uk/short-courses/search-courses/introduction-tidyverse-and-ggplot2
- R: further topics https://www.ucl.ac.uk/short-courses/search-courses/r-further-topics
- Introduction to Bayesian Inference and Modelling https://www.ucl.ac.uk/short-courses/search-courses/introduction-bayesian-inference-and-modelling

Coding communities

- NHS-R
- R-Ladies (London)
- UCL R group

UCL R User Group

UCL Home » Advanced Research Computing » Community & Events

UCL R User Group

The UCL R user group is a friendly group for R users to meet, hear a talk and discuss R questions.

24 Sep 2024	Getting started in R	Data science introduction session with a focus on setting up to code in R, showing people how to import datasets, summarise data descriptively and where to find help.
17 Dec 2024	Understanding data	Introduction to statistical thinking, types of data and how data types influence the methods we use, data management while doing the analysis, and data processing tips.
25 Mar 2025	Uncovering differences between groups	Introduction to regression analyses, identifying covariates for your models, DAGs, correlation and confounding.
17 Jun 2025	Setting up the experiment	Study designs and sample size calculations. This focus will be on what information we need for doing sample size calculations and looking at some examples. I will also introduce the approach of simulation studies at the study design phase and why they are valuable.