University of BRISTOL

Tom’s simple guide to writing a script

Thomas Battram (he/him)

Credit: Dr Gwen Fernandes on helping prepare slides

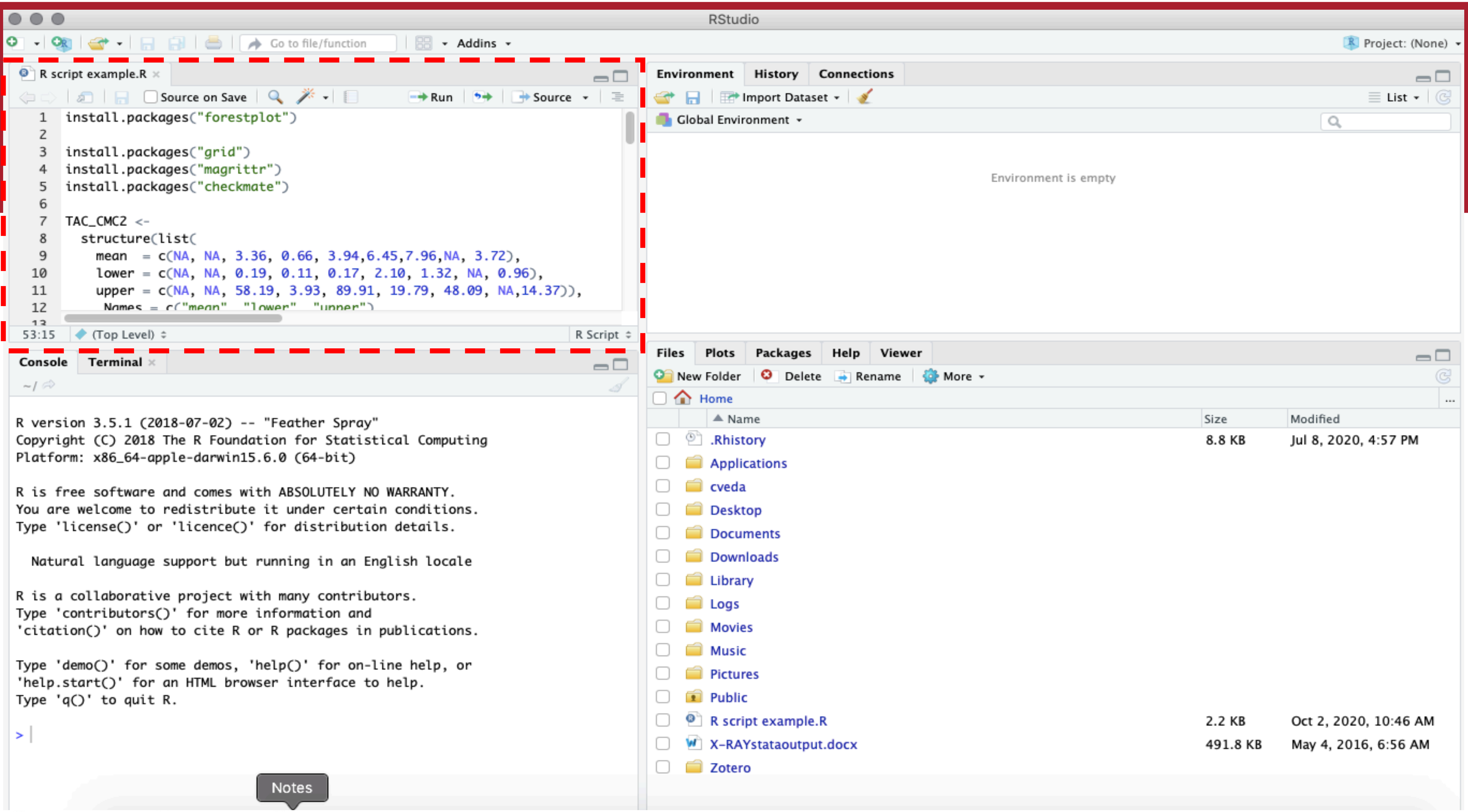
bristol.ac.uk

Outline

- Making a new R script
- Commenting your scripts
- What to think about before starting to script
- How the top of your script should look
- Making your code easier to read
- Saving your scripts
- Bad/good script comparison

Making a new R script

- The usual R Studio screen has four windows:
  - CONSOLE
  - WORKPLACE AND HISTORY
  - FILES, PLOTS, PACKAGES AND HELP
  - **R SCRIPT AND DATA VIEW** (this is where you keep a record of your work. For Stata users, this would be like your do-file, for SPSS users it is like the syntax and for SAS users, the SAS program)

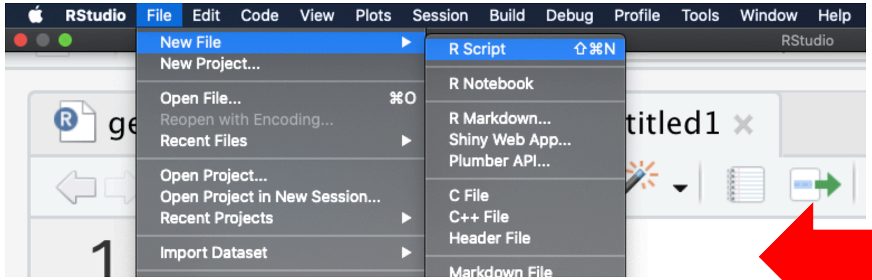


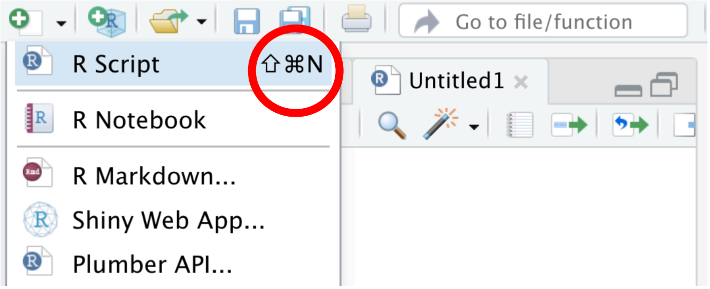
```
install.packages("forestplot")

install.packages("grid")
install.packages("magrittr")
install.packages("checkmate")

TAC_CMC2 <-
  structure(list(
    mean  = c(NA, NA, 3.36, 0.66, 3.94,6.45,7.96,NA, 3.72),
    lower = c(NA, NA, 0.19, 0.11, 0.17, 2.10, 1.32, NA, 0.96),
    upper = c(NA, NA, 58.19, 3.93, 89.91, 19.79, 48.09, NA,14.37)),
     Names = c("mean", "lower", "upper"),
```

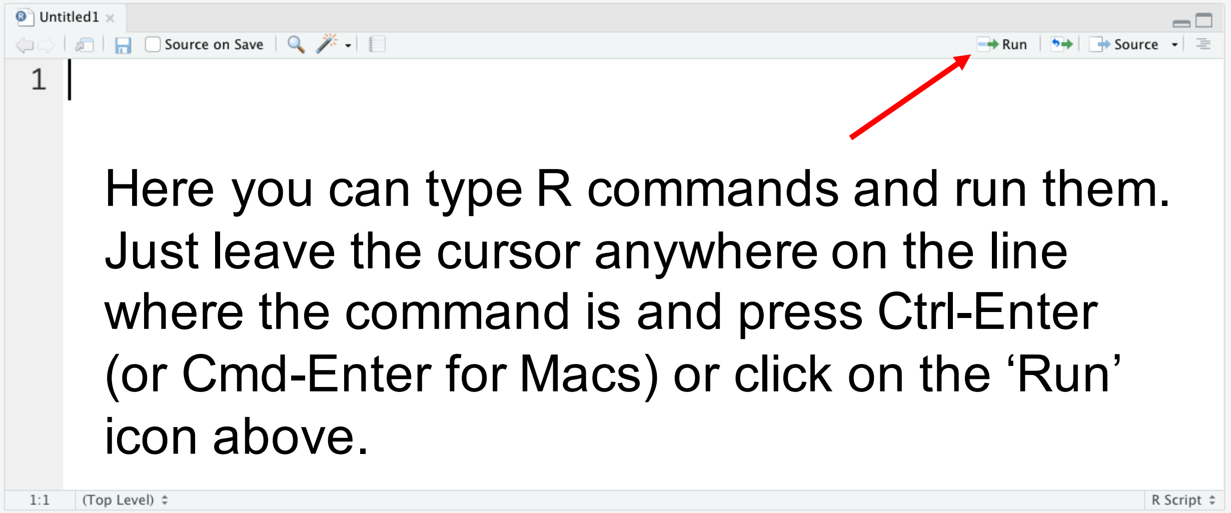
```
R version 3.5.1 (2018-07-02) -- "Feather Spray"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 
```

Name	Size	Modified
.Rhistory	8.8 KB	Jul 8, 2020, 4:57 PM
Applications		
cveda		
Desktop		
Documents		
Downloads		
Library		
Logs		
Movies		
Music		
Pictures		
Public		
R script example.R	2.2 KB	Oct 2, 2020, 10:46 AM
X-RAYstataoutput.docx	491.8 KB	May 4, 2016, 6:56 AM
Zotero		

Making a new R script



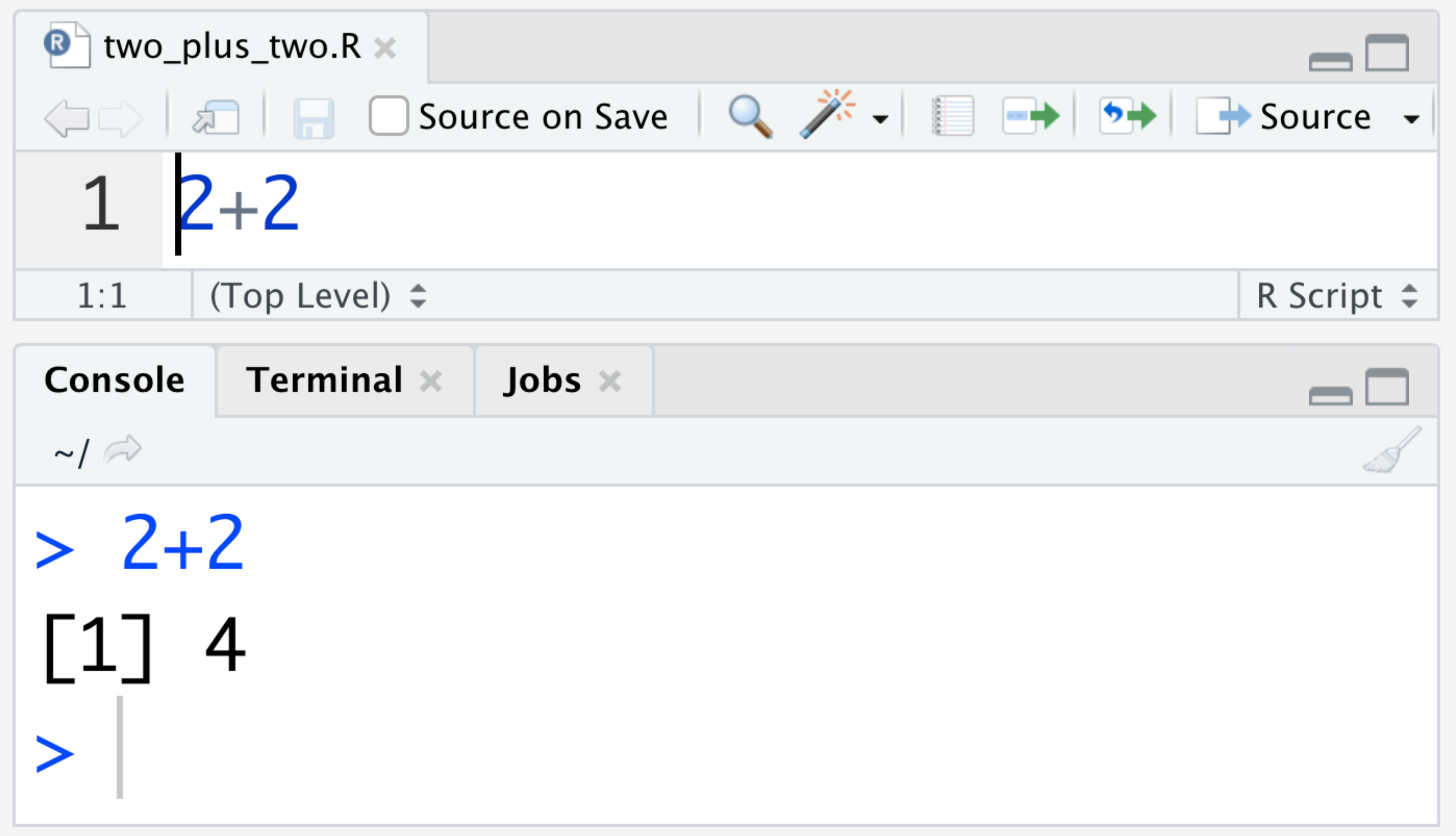




Here you can type R commands and run them. Just leave the cursor anywhere on the line where the command is and press Ctrl-Enter (or Cmd-Enter for Macs) or click on the ‘Run’ icon above.

bristol.ac.uk

Making a new R script

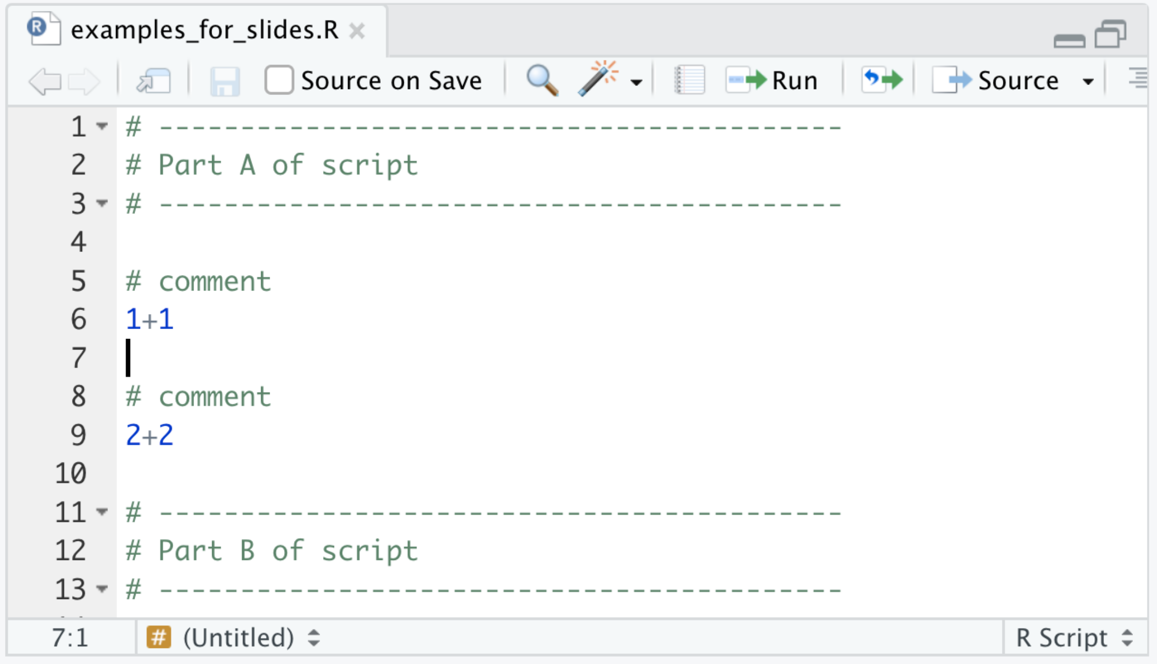


```
2+2
```

```
> 2+2
[1] 4
>
```

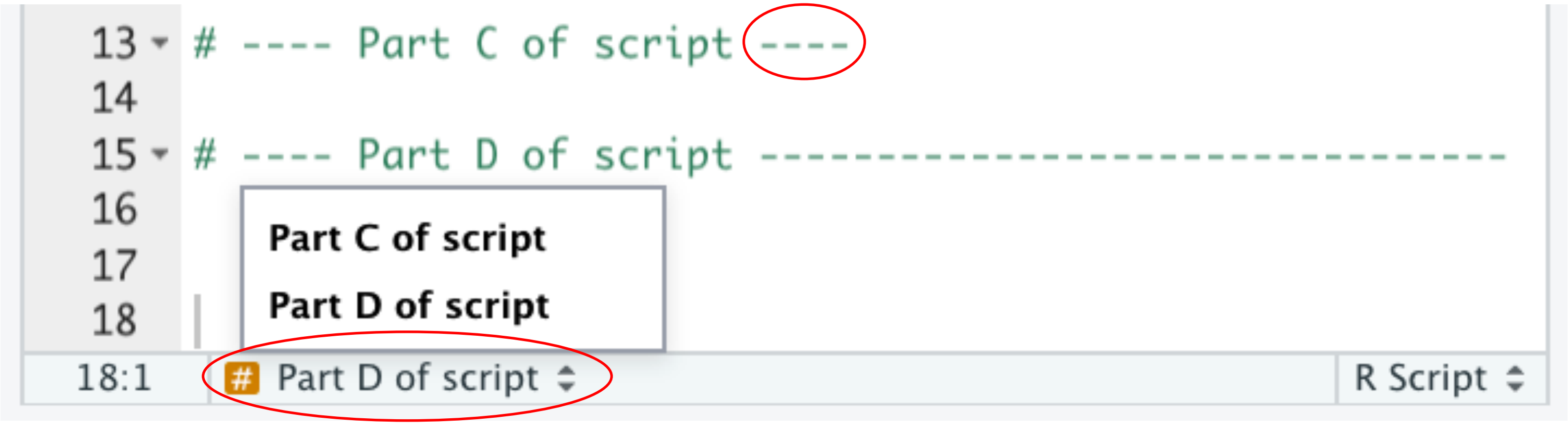
Commenting

- –Code with a # before it does not get run
- –This is useful for making your scripts much easier to read!
- –Comment on WHY and WHAT (to start with)
  - ➢Start with many comments! Also, use comments to split up the script to make it clearer



```
# ----------------------------------------
# Part A of script
# ----------------------------------------

# comment
1+1

# comment
2+2

# ----------------------------------------
# Part B of script
# ----------------------------------------
```

RStudio sections



```
# ---- Part C of script ----

# ---- Part D of script --------------------------------
```

3 steps before starting

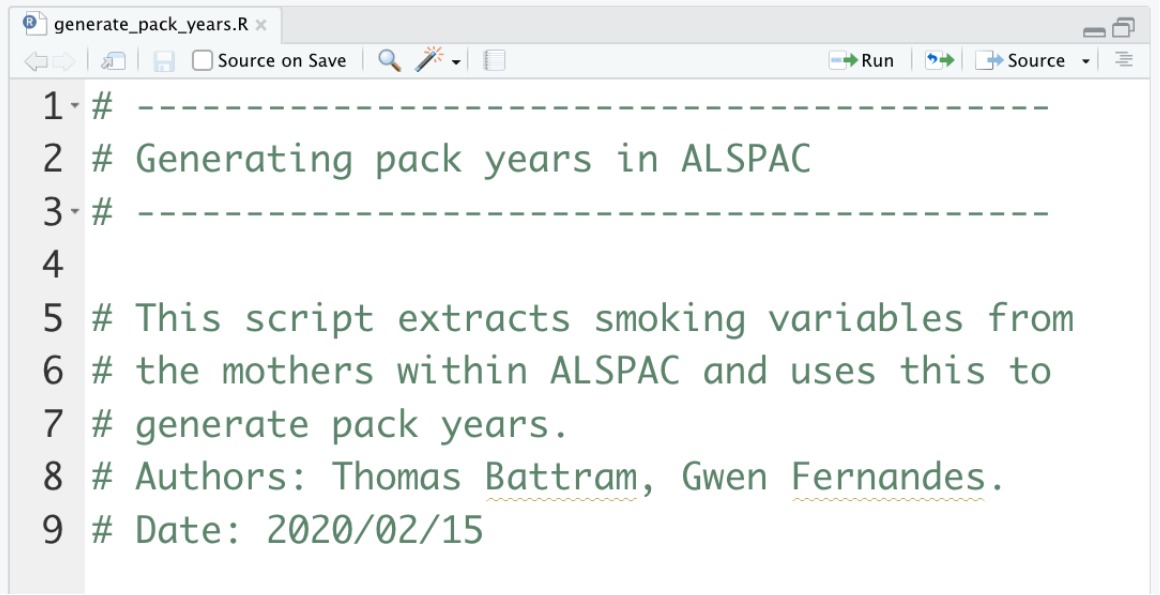
1. Why am I writing this script?

2. What do I need to write this script?

3. How am I going to write this script?

Step 1: Why am I writing this script?

- Think about it and write it down!
  - ➢e.g. clean a dataset OR assess association between x and y
- Give the script a good name, write a descriptive title + add a couple of lines that describe the purpose of the script



```
# ------------------------------------------
# Generating pack years in ALSPAC
# ------------------------------------------

# This script extracts smoking variables from
# the mothers within ALSPAC and uses this to
# generate pack years.
# Authors: Thomas Battram, Gwen Fernandes.
# Date: 2020/02/15
```

Step 2: What do I need to write this script?

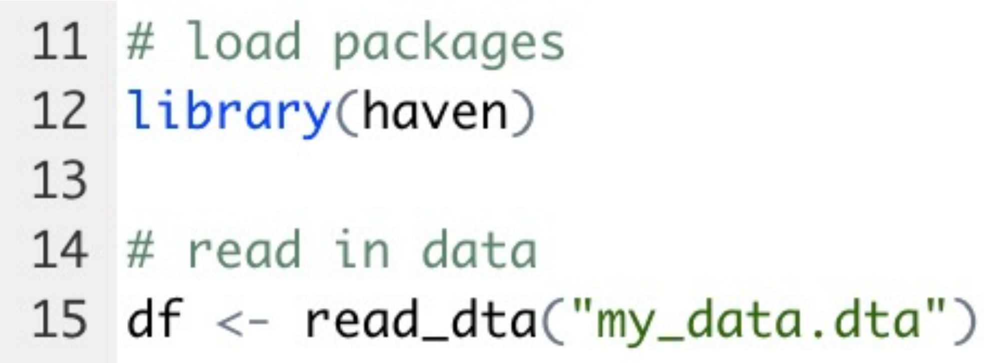
- Datasets!
- Using only base R can make things difficult… Packages!
- Packages are made by others and are there to make your life easier

For example, reading data into R can be tricky depending on how the data is stored, but packages can make this easier!

Reading data into R

Depending on what form the data is in, you have to use different functions to read in the data. Data you get may be in:

- excel spreadsheets
- comma seperated value (csv) files
- tab separated value (tsv) files
- spss files
- stata files
- images (e.g. .png files)



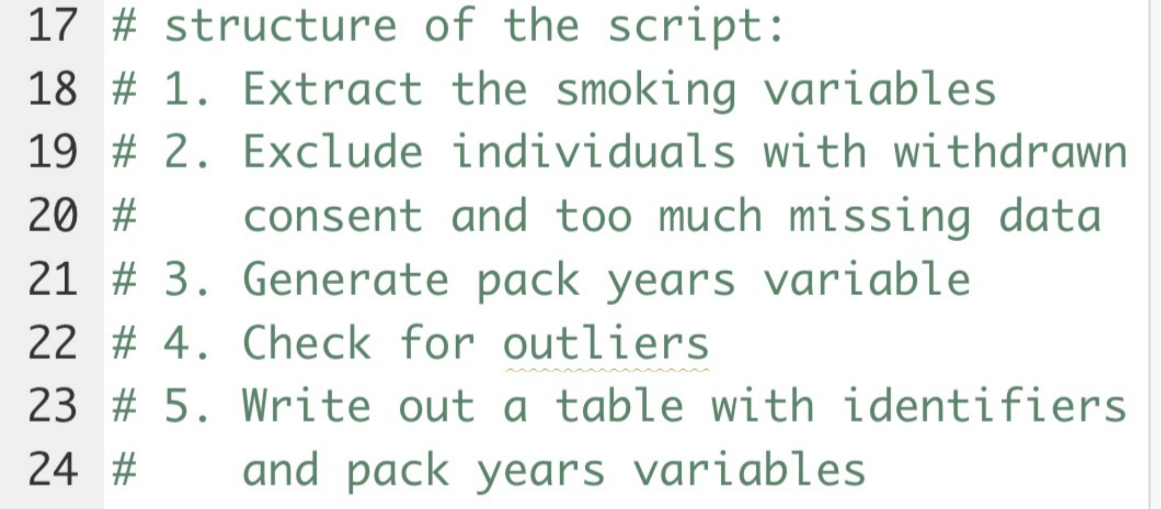
```
# load packages
library(haven)

# read in data
df <- read_dta("my_data.dta")
```

These need different functions to read them in, some of which are only available with certain packages.

Step 3: How am I going to write this script?

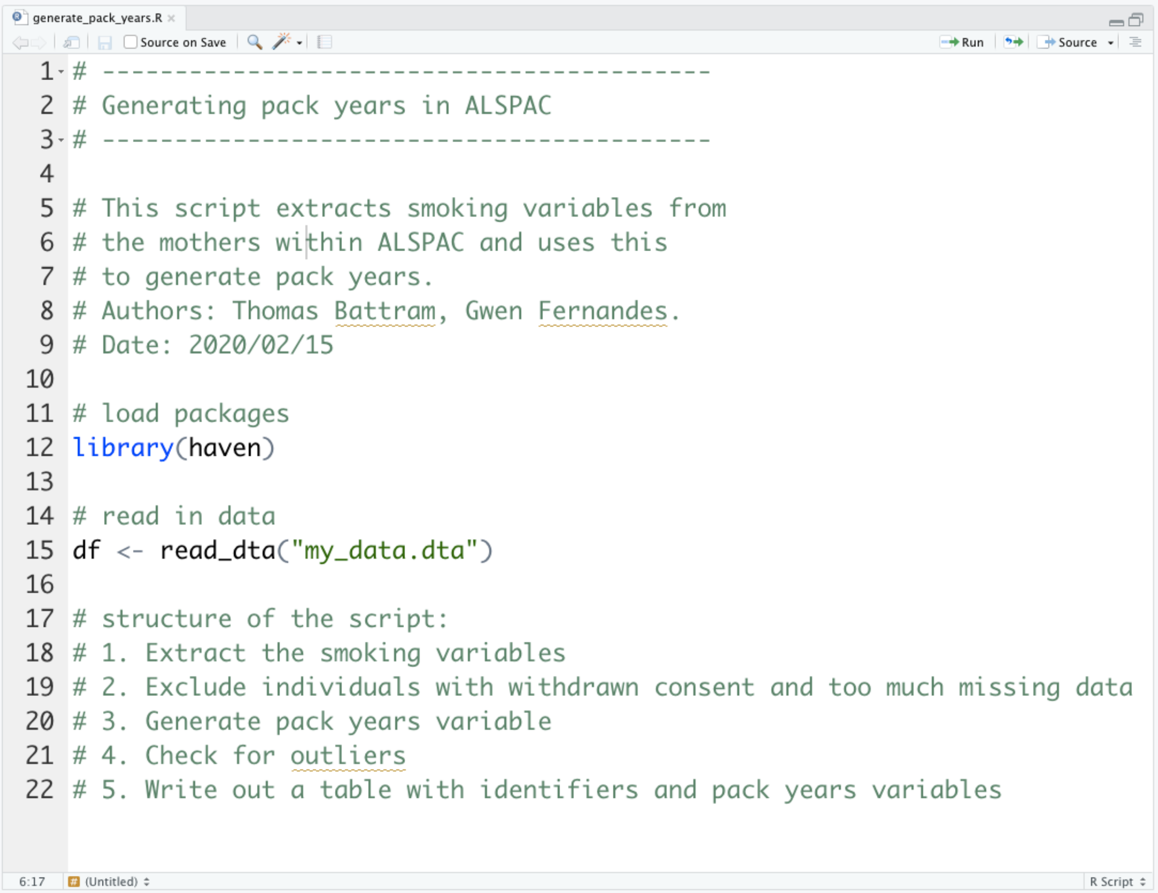
– Linked to why you are writing it and what you need to write it!

– Write out each step



```
# structure of the script:
# 1. Extract the smoking variables
# 2. Exclude individuals with withdrawn
#     consent and too much missing data
# 3. Generate pack years variable
# 4. Check for outliers
# 5. Write out a table with identifiers
#     and pack years variables
```

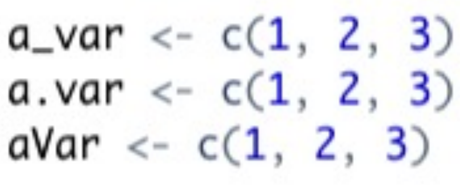
Top of the script



```r
# ------------------------------------------
# Generating pack years in ALSPAC
# ------------------------------------------

# This script extracts smoking variables from
# the mothers within ALSPAC and uses this
# to generate pack years.
# Authors: Thomas Battram, Gwen Fernandes.
# Date: 2020/02/15

# load packages
library(haven)

# read in data
df <- read_dta("my_data.dta")

# structure of the script:
# 1. Extract the smoking variables
# 2. Exclude individuals with withdrawn consent and too much missing data
# 3. Generate pack years variable
# 4. Check for outliers
# 5. Write out a table with identifiers and pack years variables
```

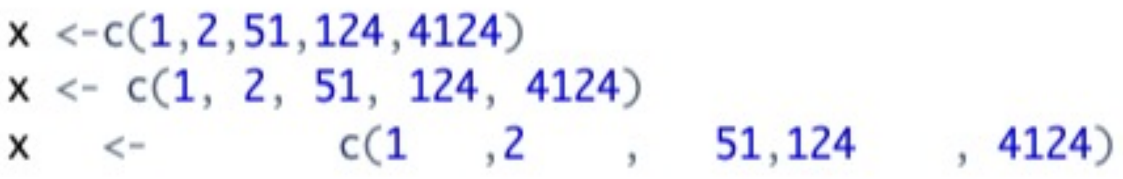
Make your code easy to read

1. Use a consistent style when writing code



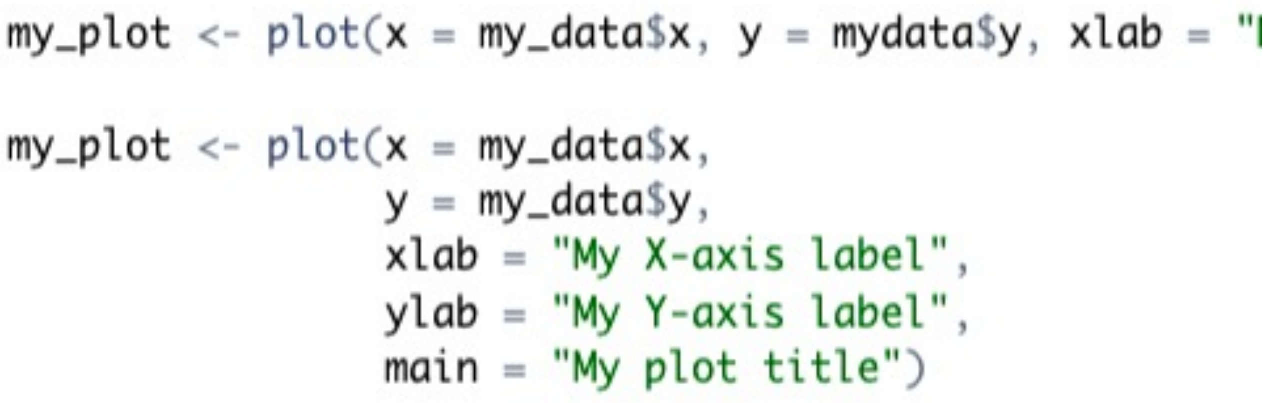
```
a_var <- c(1, 2, 3)
a.var <- c(1, 2, 3)
aVar <- c(1, 2, 3)
```

2. Use spaces appropriately



```
x <-c(1,2,51,124,4124)
x <- c(1, 2, 51, 124, 4124)
x    <-          c(1    ,2      ,     51,124      , 4124)
```

3. Use indents appropriately
   a) No long lines of code!

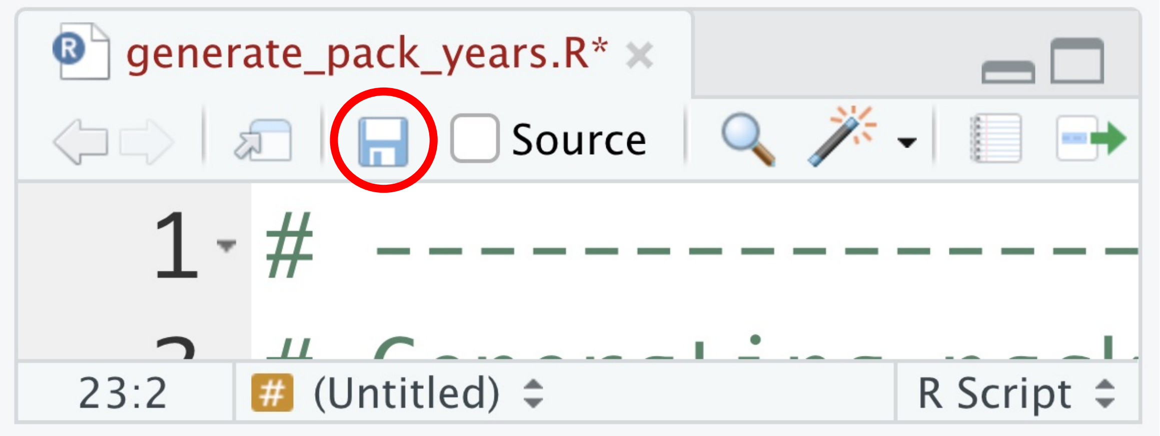


```
my_plot <- plot(x = my_data$x, y = mydata$y, xlab = "

my_plot <- plot(x = my_data$x,
                y = my_data$y,
                xlab = "My X-axis label",
                ylab = "My Y-axis label",
                main = "My plot title")
```

Saving scripts

- It’s important to save scripts as you go and you can always come back to them and send them to other people.

- Give the script a good name and save it regularly!



```
# ----------------
```

CTRL (or CMD) + S

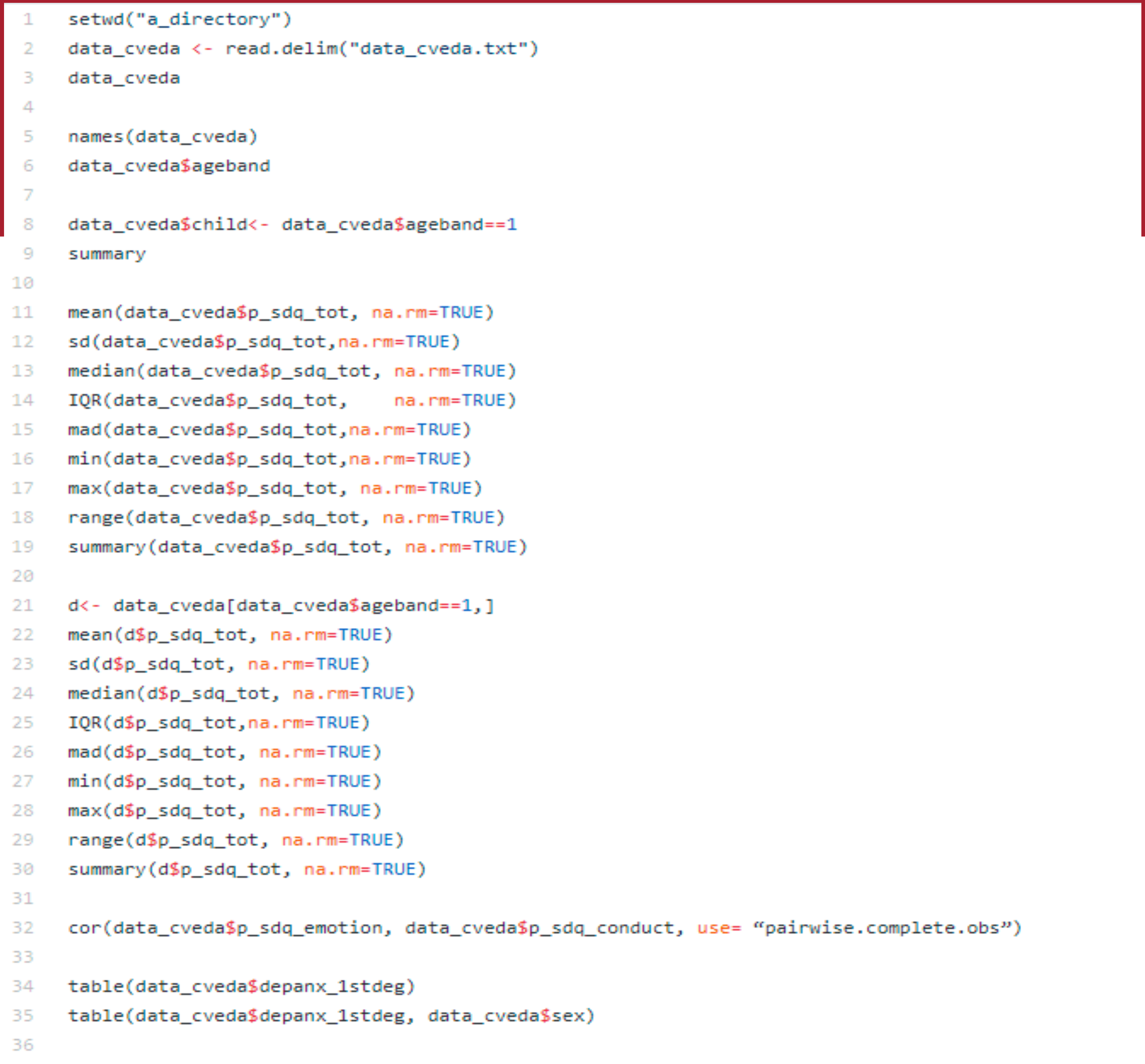
Summary

3 questions to think about before writing a script:

1. Why am I writing this script?
2. What do I need to write this script?
3. How am I going to write this script?

- Comment your code a lot (using #)
- Make your code easy to read
- Save your scripts regularly

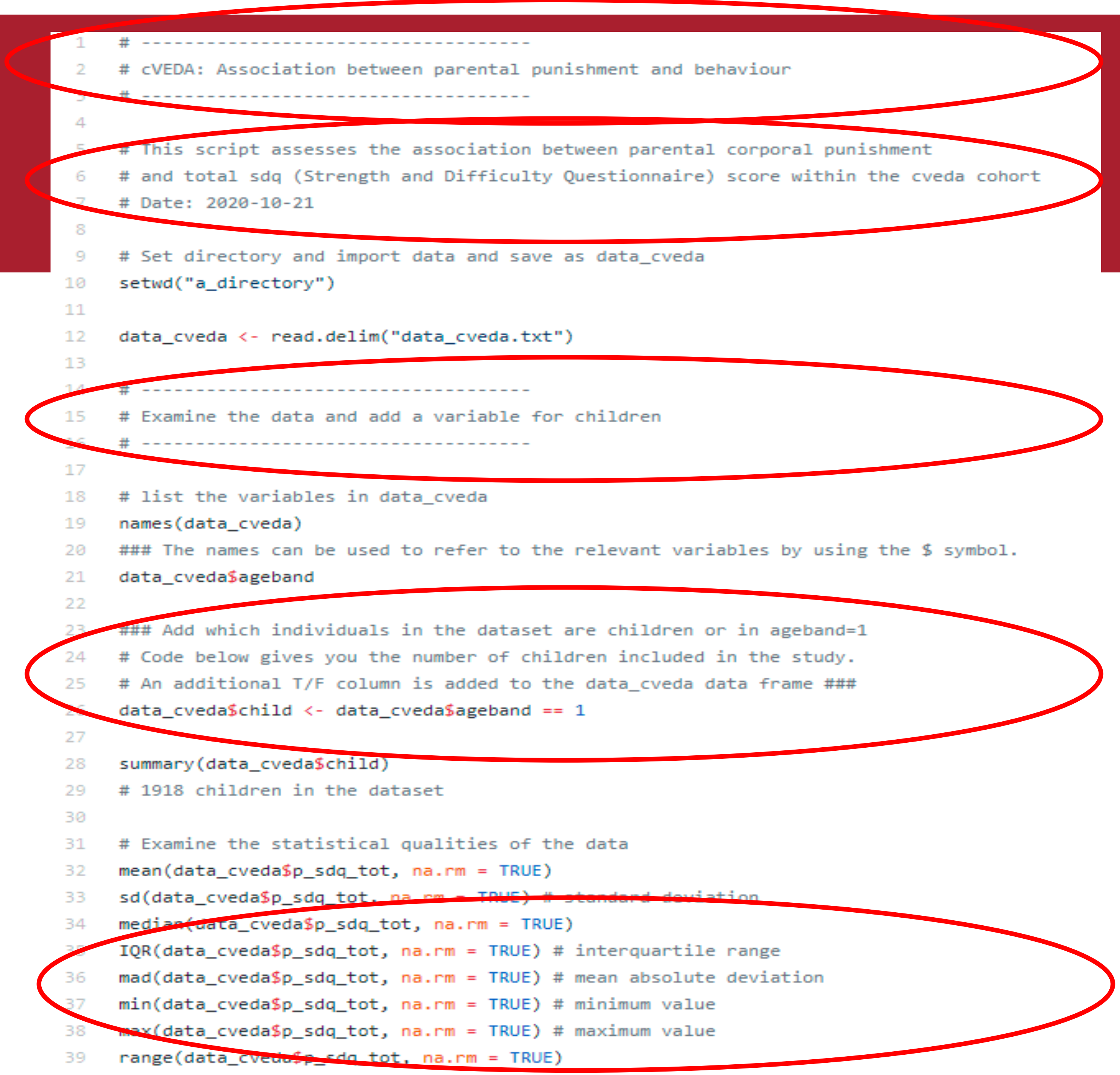
Bad Script



```
setwd("a_directory")
data_cveda <- read.delim("data_cveda.txt")
data_cveda

names(data_cveda)
data_cveda$ageband

data_cveda$child<- data_cveda$ageband==1
summary

mean(data_cveda$p_sdq_tot, na.rm=TRUE)
sd(data_cveda$p_sdq_tot,na.rm=TRUE)
median(data_cveda$p_sdq_tot, na.rm=TRUE)
IQR(data_cveda$p_sdq_tot,    na.rm=TRUE)
mad(data_cveda$p_sdq_tot,na.rm=TRUE)
min(data_cveda$p_sdq_tot,na.rm=TRUE)
max(data_cveda$p_sdq_tot, na.rm=TRUE)
range(data_cveda$p_sdq_tot, na.rm=TRUE)
summary(data_cveda$p_sdq_tot, na.rm=TRUE)

d<- data_cveda[data_cveda$ageband==1,]
mean(d$p_sdq_tot, na.rm=TRUE)
sd(d$p_sdq_tot, na.rm=TRUE)
median(d$p_sdq_tot, na.rm=TRUE)
IQR(d$p_sdq_tot,na.rm=TRUE)
mad(d$p_sdq_tot, na.rm=TRUE)
min(d$p_sdq_tot, na.rm=TRUE)
max(d$p_sdq_tot, na.rm=TRUE)
range(d$p_sdq_tot, na.rm=TRUE)
summary(d$p_sdq_tot, na.rm=TRUE)

cor(data_cveda$p_sdq_emotion, data_cveda$p_sdq_conduct, use= “pairwise.complete.obs”)

table(data_cveda$depanx_1stdeg)
table(data_cveda$depanx_1stdeg, data_cveda$sex)
```

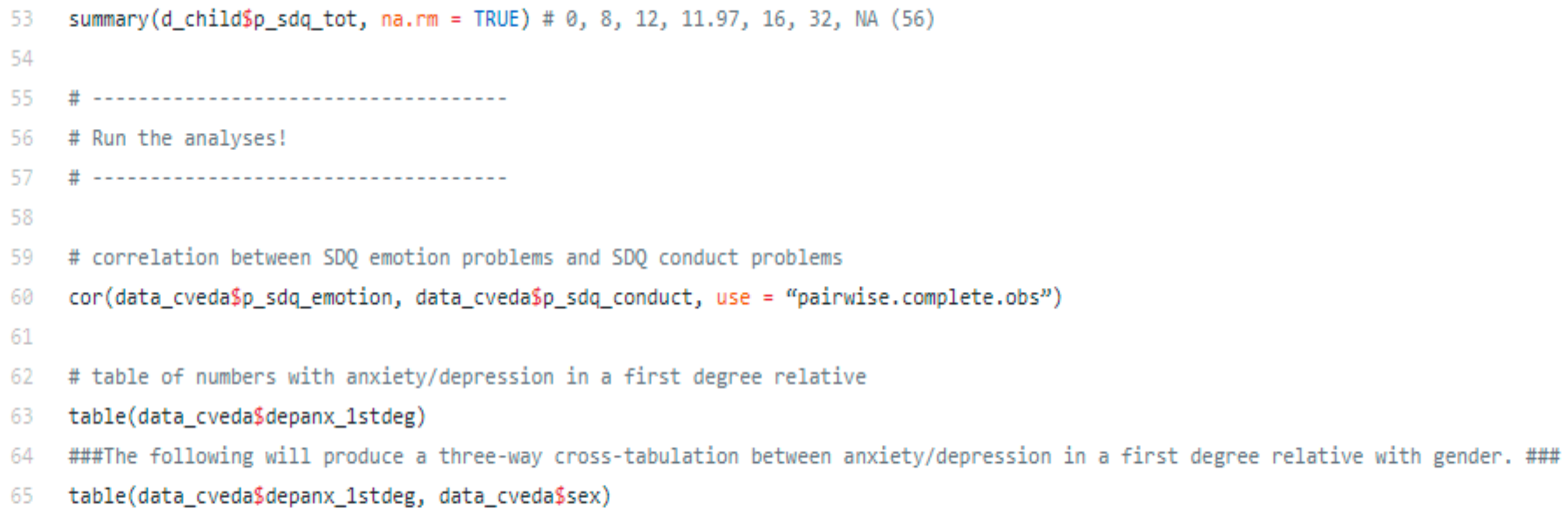
1. No title
2. No description
3. No sections
4. No comments at all!
5. Spacing could potentially be improved

Good Script



```
# ------------------------------------
# cVEDA: Association between parental punishment and behaviour
# ------------------------------------

# This script assesses the association between parental corporal punishment
# and total sdq (Strength and Difficulty Questionnaire) score within the cveda cohort
# Date: 2020-10-21

# Set directory and import data and save as data_cveda
setwd("a_directory")

data_cveda <- read.delim("data_cveda.txt")

# ------------------------------------
# Examine the data and add a variable for children
# ------------------------------------

# list the variables in data_cveda
names(data_cveda)
### The names can be used to refer to the relevant variables by using the $ symbol.
data_cveda$ageband

### Add which individuals in the dataset are children or in ageband=1
# Code below gives you the number of children included in the study.
# An additional T/F column is added to the data_cveda data frame ###
data_cveda$child <- data_cveda$ageband == 1

summary(data_cveda$child)
# 1918 children in the dataset

# Examine the statistical qualities of the data
mean(data_cveda$p_sdq_tot, na.rm = TRUE)
sd(data_cveda$p_sdq_tot, na.rm = TRUE) # standard deviation
median(data_cveda$p_sdq_tot, na.rm = TRUE)
IQR(data_cveda$p_sdq_tot, na.rm = TRUE) # interquartile range
mad(data_cveda$p_sdq_tot, na.rm = TRUE) # mean absolute deviation
min(data_cveda$p_sdq_tot, na.rm = TRUE) # minimum value
max(data_cveda$p_sdq_tot, na.rm = TRUE) # maximum value
range(data_cveda$p_sdq_tot, na.rm = TRUE)
```

Good Script



```
summary(d_child$p_sdq_tot, na.rm = TRUE) # 0, 8, 12, 11.97, 16, 32, NA (56)

# ------------------------------------
# Run the analyses!
# ------------------------------------

# correlation between SDQ emotion problems and SDQ conduct problems
cor(data_cveda$p_sdq_emotion, data_cveda$p_sdq_conduct, use = “pairwise.complete.obs”)

# table of numbers with anxiety/depression in a first degree relative
table(data_cveda$depanx_1stdeg)
###The following will produce a three-way cross-tabulation between anxiety/depression in a first degree relative with gender. ###
table(data_cveda$depanx_1stdeg, data_cveda$sex)
```

Any Questions?