Advanced Psychological Research Methods (PSY4034)

Christopher J. Wilson

September 2021

Contents

Chapter 1

Welcome to the module

Note:To watch the embedded videos on this site, you will need to be logged into Teesside University's Blackboard site and be a part of this course. If you are on the course:

- Press *Ctrl + T* to open a new tab (keeping this one open) and go to https://bb.tees.ac.uk.
- Login to Blackboard. Keep the blackboard tab open and switch back to this page.
- Refresh this page (hit F5 on your keyboard).
- You should now be able to play all of the video content.

1.1   Module Overview

This Level 7 module for first year Doctorate in Clinical Psychology trainees aims to enable you to:

- Refresh and extend your knowledge, skills and critical understanding of advanced research methods using both qualitative and quantitative approaches;
- Creatively apply the principles of quantitative and qualitative research methods to clinical psychology research and practice;
- Refresh and extend your skills in project design, management, analysis and presentation.

The module is also designed to explicitly prepare you for the two Doctorate level research modules which occur in Years Two and Three of the programme, ensuring that you have the requisite knowledge and skills to successfully engage with those modules.

The key foci for this module include:

- critical review of established literature
- project design
- project management
- data analysis
- dissemination of research findings

The module is taught using a variety of techniques to best enhance your knowledge and understanding of the application of research theory and methods in the context of clinical psychology. These include lectures, seminars, guided statistical analysis and tutorials with the latter being used to provide individual guidance and formative feedback. The module has its own site on the University's Virtual Learning Environment http://eat.tees.ac.uk - known as Blackboard), with resources and literature designed to support learning.

1.2   Module timetable and delivery for 2020/2021

The timetable for this module appears on the Clinical Psychology Programme Site/Timetables. The majority of the sessions will take place on Monday mornings. Please note that the timetable should be checked on a regular basis.

1.2.1   Learning and Teaching Strategies

The module is taught using a variety of techniques to best enhance your knowledge and understanding of the application of research theory and methods in the context of clinical psychology. This includes activities designed to encourage independent learning, a key skill for successful performance in research modules in Years Two and Three of the programme. Evidence of independent learning is expected in the assignments for this module. Specific links are made with research informed activity in practice.

You will be provided with two papers for critical review at the start of the module and asked to decide which one you will use for your summative critical review assignment; one of these papers is from a quantitative research tradition and the other is from a qualitative research tradition.

All presentations (with added annotations) are available, along with additional support materials, via an e-learning@tees on the VLE. E-learning is enabled through group activities on the VLE or Microsoft Teams where discussion and problem solving is undertaken in relation to tasks set during teaching sessions. The discussion boards or Microsoft Teams site will be used to ask and answer questions that arise from the taught material and also your independent work.

1.3 Assessment

1.3.1 Formative assessment

Formative feedback is provided throughout the module through practical exercises and in seminars on trainee presentations.

By the end of year one, trainees are expected to have identified a thesis topic and have a completed research proposal. As such, there are a number of formative milestones across the year that will be monitored by the module team. Please see Appendix 1 of this guide for details.

The required format for thesis research proposals can be found in Appendix 2 of the DClinPsy Programme Research Handbook.

The formal formative assessment is of a presentation of the thesis research proposal to be presented during the research panels, which take place in July 2022. The presentation will be 20 minutes' long and will outline the thesis project that the trainee will develop in Years 2 and 3. There will also be 10 minutes allotted for questions from the panel which will have two academic members and one clinical member. This formative assessment is intended as a starting point for the Year 2 and 3 research methods modules. The timing is important as it should enable trainees to start the process of ethics approval for their dissertation. earlier. The trainees will hand in a printed copy of their slides with explanatory notes and references.

Formative Assessment Criteria

The following criteria will be used to assess the assignment:

- Effective justification for the study.
- Clearly defined research question.
- Comprehensive and critical review of the literature (within time constraints).
- Realistic research design.
- Effective consideration of ethical issues.
- Clear plan for writing up and dissemination.
- Fulfillment of professional research ethics requirements.
- Adherence to the relevant guidance for presentation as advised by the Module Tutor.

1.3.2 Summative Assessments

Assessment consists of an ICA and an ECA, each worth 50% of the overall module mark. The deadlines for these assessments can be found in the assessments section on Blackboard or in the programme assessment timetable.

ICA (50%) - A critical review of a published primary research paper (choice to be made by a trainee from papers with different methodologies provided by the tutor). (2,000 words). *Learning outcomes: (KU 1-4, CIS 1-3, KTS 1-3)*

ICA Assessment Criteria (Critical Appraisal of Published Primary Research Paper)

The following criteria will be used to assess the assignment:

- Demonstrate a critical understanding of the role of the reviewed paper for clinical psychologists in service delivery and/or practice.
- Demonstrate a critical and comprehensive understanding of the relevant methodological issues.
- Systematically and critically evaluate stages of the research process.
- Demonstrate a comprehensive and critical understanding of the ethical issues involved in the research.
- Reach effectively argued conclusions.
- Demonstrate independent learning ability through reflection on the critical review process.
- Adhere to the American Psychological Association (APA) guidelines for presentation and referencing.

ECA (50%) - A research project proposal which both addresses limitations identified in the ICA critical review (1) and develops the research further with the use of an alternative methodology (2,000 words).
Learning outcomes: (KU 1-4, CIS 1-3, PPS 1-2, KTS 1-5)

ECA Assessment Criteria (Research Project Proposal)

The following criteria will be used to assess the assignment:

- Identify a project that demonstrates a detailed and critical understanding of the research evidence reviewed and wider methodological issues, including the role of the project for informing service delivery and/or practice.
- Provide detailed and appropriately justified solutions to the design of the research project.
- Consider both methodological and ethical issues in the design of the research project.
- Demonstrate a detailed and critical understanding of the data analysis required for the proposed study.
- Adhere to the American Psychological Association (APA) guidelines for presentation and referencing.

Suggested structure for ECA

This assessment is a research proposal and should be structured to describe the research aims and how the methodological approach suggested will address these aims. The previous assessment will have raised certain critiques of the research paper discussed, and these issues might inform the proposed project. However, it is important that this assessment stands alone as a written piece of work, so there should be sufficient information and reference to literature for the reader to understand the basis of the research question and methodological choices that are made (similar to when someone reads a dissertation or research paper method).

In terms of strucutre, there are some elements that it would be logical to include:

- A brief introduction
- A statement of the research question(s) and (if appropriate) hypotheses
- Information about planned sampling approach / participant inclusion and exclusion criteria
- An overview of the proposed method, including information about measures and techniques that would be used
- An overview of the planned analysis

It would also be useful to include, though not necessarily in an independent section information about:

- the clinical relevance/impact of the research
- ethical considerations that need to be taken into account.

1.3.3 Word limits for assessment

Word limits are as stated above (note that there is no allowance of +10%). The word count refers to the assignment itself and does not include the reference list or tables/graphs. The references cited in the main body of text are included in the word count.

1.3.4 Submitting work

All work should be submitted electronically, using the apporpriate links on the module Blackboard site. A printed hard copy of the assignment is **not** required and should not be submitted. **The university's policy is that all assessments must be submitted by 4 p.m. on the day of the deadline.**

1.3.5 A note about referencing

There is an expectation that all academic assignments conform to current American Psychological Association referencing and citation conventions. Poor referencing will be taken into consideration when marking. It is recommended that you use a digital reference management system (e.g., Refworks, Mendley), which are freely available (and will save you time). The following online resources are also useful:

http://reciteworks.com/ - good for checking fine details (e.g., missing references)

http://www.apastyle.org - detailed guidance for APA style

https://owl.english.purdue.edu/owl/resource/560/01/- additional advice for APA style

1.4   Statistical analysis software

You may be familiar with SPSS from your undergraduate statistics teaching. Please note that we do not use SPSS for teaching and instead use R Statistics. The reason for this is that R is a free statistical package, meaning that it can be accessed in NHS settings that do not have funding for SPSS. This will enable you to run statistical analyses whilst on placement where required, and also enables you to conduct statistical analyses as a qualified Psychologist without incurring any software costs. R is also more flexible than SPSS and has greater functionality. During the teaching you will be shown how to set up and install R, and how to run statistical analyses in this software.

1.4.1   Downloading R and R Studio software

You can obtain R and R Studio from the following links:

https://cran.r-project.org/

https://rstudio.com/

1.5   Texbooks that can be accessed online

e-books can be accessed from the library website: https://www.tees.ac.uk/depts /lis/

1.5.1   Research Methods and Statistics

Coolican, 2019. Research Methods and Statistics in Psychology. Taylor & Francis Group

Barker, C., Pistrang, N., & Elliott, R. (2015). Research methods in clinical psychology: An introduction for students and practitioners (3rd ed.). Chichester, West Sussex: Wiley Blackwell.

Weiner, I. B., Schinka, J. A., & Velicer, W. F. (2012). Handbook of psychology, research methods in psychology (2. Aufl. ed.). Somerset: Wiley.

1.5.2   Working with R and RStudio to do analysis

Navarro, D. (2017) Learning statistics with R.

Phillips N. D. (2018) YaRrr! The Pirate's Guide to R

Horton, Pruim and Kaplan (2015) A Student's Guide to R

Mather, M. (2019) R for Academics

Wickham and Grolemund (2019). R For Data Science

Allaire and Grolemund (2019). R Markdown: The Definitive Guide

Basics of RStudio

Data Import

Data Transformation

Data Visualisation with GGPlot

1.6 Academic Support and Guidance

Please contact the module team if you have any questions, concerns or any other areas you wish to discuss.

1.6.1 Module Team Contact Details

Module Leader: Dr Christopher Wilson: christopher.wilson@tees.ac.uk

Module Team: Dr Alan Bowman: A.Bowman@tees.ac.uk

Guest lecturers from Schools within the University and Local NHS clinicians also contribute to some teaching

1.7 Considering your thesis

Your doctoral thesis is one of the largest pieces of work you will undertake during your training and it is important to start thinking about it early on.

You are advised to read around the area of your thesis topic on a continuing basis, and make use of tutorials with your supervision team when needed.

It is also advised that you consider research governance and ethics as you develop your project. Please speak to your academic supervisor about this as they will be able to advise or direct you to someone with appropriate expertise to address queries.

As specified in the Research Handbook, **please note that a revised version of your research proposal forms one of your year two summative assignments**. The deadline for this assignment is **early on in the start of second year** and it is therefore advised that you work on your revised proposal as soon as you receive feedback from the panels and have discussed this with your academic supervisor.

Chapter 2

Research Methods Concepts Revision

2.1 Basic concepts that you should already know

2.2 Probability and hypothesis testing

One of the most important things to remember about hypothesis testing in statistics is why we use the approaches we do. That is, we need statistical approaches to test hypotheses because we can only collect data from samples of the population but our research questions and hypotheses apply to whole populations. For that reason, we need a way to estimate how well the *sample* reflects the *population.*

It is common for us to want to know what the mean (average) response of the population is on certain measures. For example, we might ask the question "what is the average score on this measure of happiness?" In reality we can only measure a subset (sample) of the population, so we test as many people as we can. Below is a sample of 20 participants:

Some sample data for happiness score

participant

intervention

happiness

1

2

7.622706

2

1

11.298334

3

2

12.669884

4

2

12.247820

5

1

6.681176

6

1

12.089051

7

1

10.655260

8

1

7.895136

9

1

11.948733

10

1

10.215708

11

1

6.703790

12

1

9.523169

13

1

10.076470

14

1

11.584766

15

2

11.656510

16

1

9.741287

17

2

10.722183

18

1

8.340372

19

2

8.757775

20

2

10.281363

2.3   Variance in sample data influences our confidence in population estimates

We can see from the table above that the mean of the sample is 10.0355747. However, this is not to say that the population mean is 10.0355747. For one thing, we can see that the range of scores in the sample is between 6.6811761 and 12.6698843. The standard deviation of the sample is 2.

The fact that there is so much variance from person to person within our sample indicates that we are likely to be incorrect if we assume that the sample mean is the same as the population mean. The more variance there is within the sample data, the less confident we can be that the sample mean is an accurate representation of the population mean.

Another thing that affects our ability to generalise from sample to population is that the sample size is only 20. Larger samples are less influenced by individual outliers, so the larger the sample size is, the more confident we can be that the sample mean is representative of the population mean (provided that the participant sample is representative of the population and recruited in a way to minimise bias).

> The **standard error** of the mean can be calculated to estimate how far the mean of the sample data is likely to be from the true population mean. It uses the concepts of variance and sample size to make this estimate. Standard error is calculated by dividing standard deviation by the square root of the sample size ($SE = \frac{SD}{\sqrt{[n]}}$)

In R, we can calculate the standard error of the happiness data like so:

```
 standardError <- sd(happiness)/sqrt(length(happiness)) ## Calculate standard error

standardError #Display the standard error
```

```
## [1] 0.4132489
```

The standard error of our sample mean is 0.41. This suggests that using the sample mean is likely to be 0.41 away from the population mean.

2.4   We can use confidence intervals to make educated guesses about the population mean

Using the standard error, we can also create **Confidence intervals**, which are a range of values, within which the population mean is likely to fall. For example, we know from normal distribution that 95% of the population lies between +/- 1.96 standard deviations of the mean. If we use our sample mean ($\bar{x}$) in place of the population mean and include the standard error to account for errors in our

estimate, we come up with the following formula for 95% confidence intervals of the mean:

Lower confidence interval $= \bar{x} - 1.96 * SE$

Upper confidence interval $= \bar{x} + 1.96 * SE$

```r
mean(happiness) - 1.96 * standardError # Lower confidence interval
```
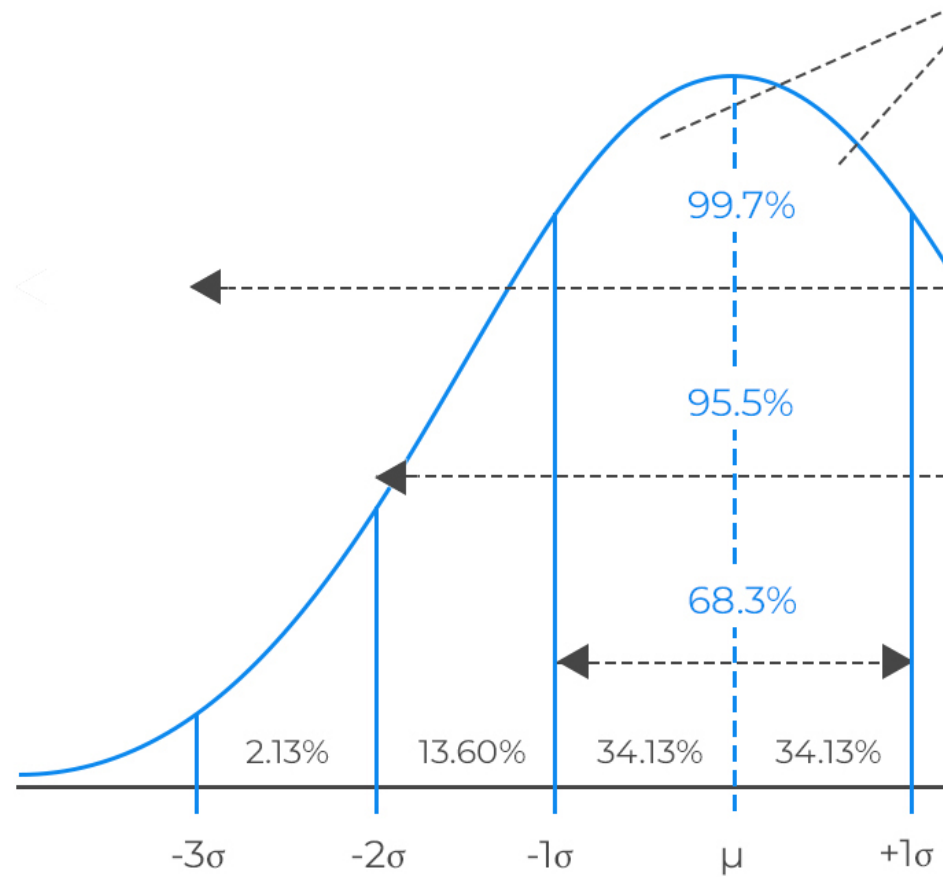
```
## [1] 9.225607
```

```r
mean(happiness) + 1.96 * standardError # Upper confidence interval
```

```
## [1] 10.84554
```

Shape of the normal distrib

99.7%

95.5%

68.3%

| 2.13% | 13.60% | 34.13% | 34.13% |

-3σ   -2σ   -1σ   μ   +1σ

No. of standard deviations from

The value of 1.96 come from the normal distribution, where 95% of the population lies between +/- 1.96 standard deviations of the mean. If we did not already know this, we could use the *qnorm()* function in R to calculate the value:

```r
# Calculate the number of standard deviations that contains 0% to 97.5% of the data (100% - 2.5%)
# We can then say that 95% of the data lies between + or - the answer:

qnorm(0.975)
```

```
## [1] 1.959964
```

However, with smaller samples, since we are less confident about generalising to the population, we use the t-distribution to calculate that value. The shape of a t-distribution changes based on the sample size, so the smaller the sample size is, the wider the range that 95% of values lie between. We can calculate the 95% value for a particular sample size in R using the *qt()* function:

```r
# The qt function relates to t-distribution

qt(0.975,df=20-1)
```

```
## [1] 2.093024
```

```r
# for qt, we need to specify the degress of freedom, which is sample size minus 1
```

We can see that when we have a sample size of 20, 95% of values in our predicted population distribution will lie between + and - 2.0930241 standard deviations. Therefore, we can calculate more accurate confidence intervals using this value:

```r
mean(happiness) - qt(0.975,df=20-1) * standardError # Lower confidence interval
```

```
## [1] 9.170635
```

```r
mean(happiness) + qt(0.975,df=20-1) * standardError # Upper confidence interval
```

```
## [1] 10.90051
```

This tells us: if we were to take infinite number of similar samples, about 95% of their confidence intervals would contain the population mean. Therefore, we think it is reasonable to estimate that the population mean is somewhere in this range.

> Often people say that a 95% confidence interval means that there is a 95% chance that the population mean is between the lower and upper confidence interval. This is **not** an accurate statement, but it is often used as a shorthand to help people conceptualise what confidence intervals are.

2.5   We can also make confidence intervals of differences between means

Often when we test hypotheses, we are testing the difference between two samples. For example, we might have 2 groups who have undergone different psychological interventions and want to know whether the difference we see our participant samples is likely to generalise to the population.

Some sample data for happiness score with participants divided into 2 groups

participant

intervention

happiness

1

2

7.622706

2

1

11.298334

3

2

12.669884

4

2

12.247820

5

1

6.681176

6

1

12.089051

7

1

10.655260

8

1

7.895136

9

1

11.948733

10

1

10.215708

11

1

6.703790

12

1

9.523169

13

1

10.076470

14

1

11.584766

15

2

11.656510

16

1

9.741287

17

2

10.722183

18

1

8.340372

19

2

8.757775

20

2

10.281363

Using the same approach as in the previous section, we can estimate a confidence interval based on the difference in means and the sample size:

```
# Calculate the number of standard deviations for 95% of the data
qt(0.975,df=20-2) # since there are 2 intervention groups, degrees of freedom is now 2
```

```
## [1] 2.100922
```

```
 group1 <- happinessSample %>% filter(intervention ==1) %>% summarise(mean = mean(happ

  group2 <- happinessSample %>% filter(intervention ==2) %>% summarise(mean = mean(happ

# calculate mean of difference
meanDifference <- group1$mean - group2$mean
seDifference <- sqrt(((group1$sd^2)/19) + ((group2$sd^2)/19))


# calculate 95% CI of this
meanDifference - seDifference * qt(c(0.975), 20-2) # lower CI
```

```
## [1] -2.077858
```

```
meanDifference + seDifference * qt(c(0.975), 20-2) # upper CI
```

```
## [1] 0.4474324
```
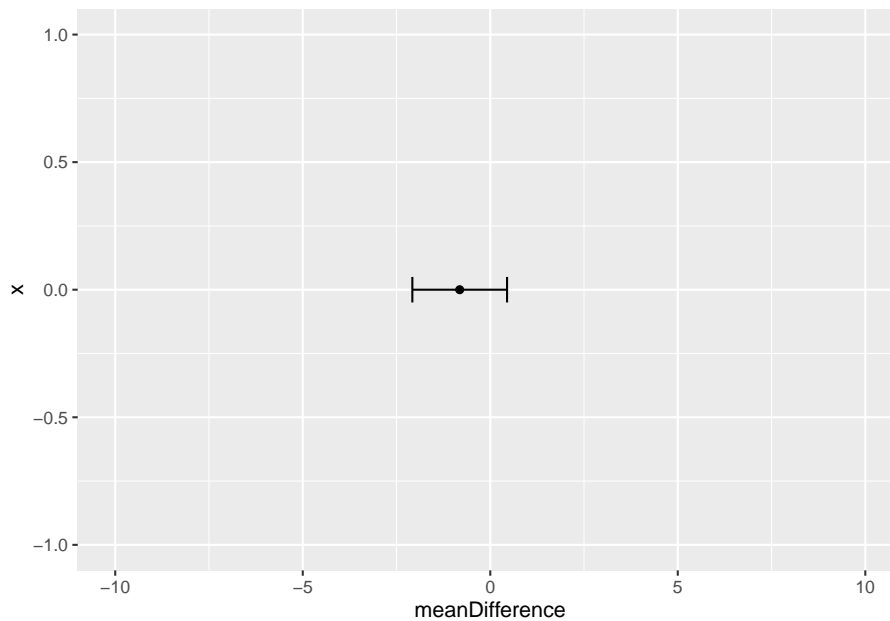
This tells use that the 95% confidence interval of the difference is between -2.0778583 and 0.4474324. An important part of interpreting this, is to notice whether any point between these values is equal to zero. If the confidence interval of a difference contains a zero value, this means that in future research, with similar samples, it would be possible to see zero difference between the groups. If, on the other hand, the confidence interval does not cross zero, then it is likely that in future research, with similar samples, we would see some difference between the means.

> The fact of whether confidence intervals cross zero (or not) is linked directly to the idea of hypothesis testing and statistical significance.

2.6 The null hypothesis and statistical significance

Using the same study from the previous example: we know that the null hypothesis can be phrased as "in the population, there is no difference between groups." We then see how the confidence interval of a difference can help us test the null hypothesis: if the null hypothesis were not true, then it is unlikely that the confidence interval of the difference would contain zero.

Therefore, if confidence intervals overlap, then there is a possibility of no difference existing between the populations. As such, we are unable to reject the null hypothesis.

Chapter 3

Introduction to R and R Studio

3.0.1 Overview of RStudio.cloud

3.1 By the end of this section, you should be able to:

- Download R and R studio
- Identify the R script, R console, Data environment and file browser in R studio
- Write and run R code from a script
- Install and load R packages

3.2 Why learn / use R?

3.2.1 Some information about R

- R is developed and used by scientists and researchers around the world
- Open source = no cost
- Constant development
- Connects to other data science/research tools
- Worldwide community: training widely available
- Encourages transparency and reproducibility
- Publication-ready outputs

3.2.2 Moving from other software to R

- Workflow is different

- – Organise files and data differently
- – Workspace can contain data and outputs
- – Can manage multiple datasets within a workspace
- Learning curve can be steep initially
  - – e.g. Variables and coding, scripts
- Need to know what you want
  - – e.g. building your regression model / ANOVA error terms

3.3   R has many advantages

- Using scripts means analysis is easy to follow and reproduce
- R scripts are small, online collaboration, no SPSS "older version" problems
- Data can be organised and reorganised however you need it (tidyr)
- Packages are available for "cutting edge" analysis: e.g. Big Data & Machine Learning
- A robust language for precise plots and graphics (ggplot)
- R analysis code can be embdeded into documents and presentations (R Markdown)

3.4   Download R and R Studio

Click on these links to download:

- R project
- RStudio

3.5 The R Studio environment
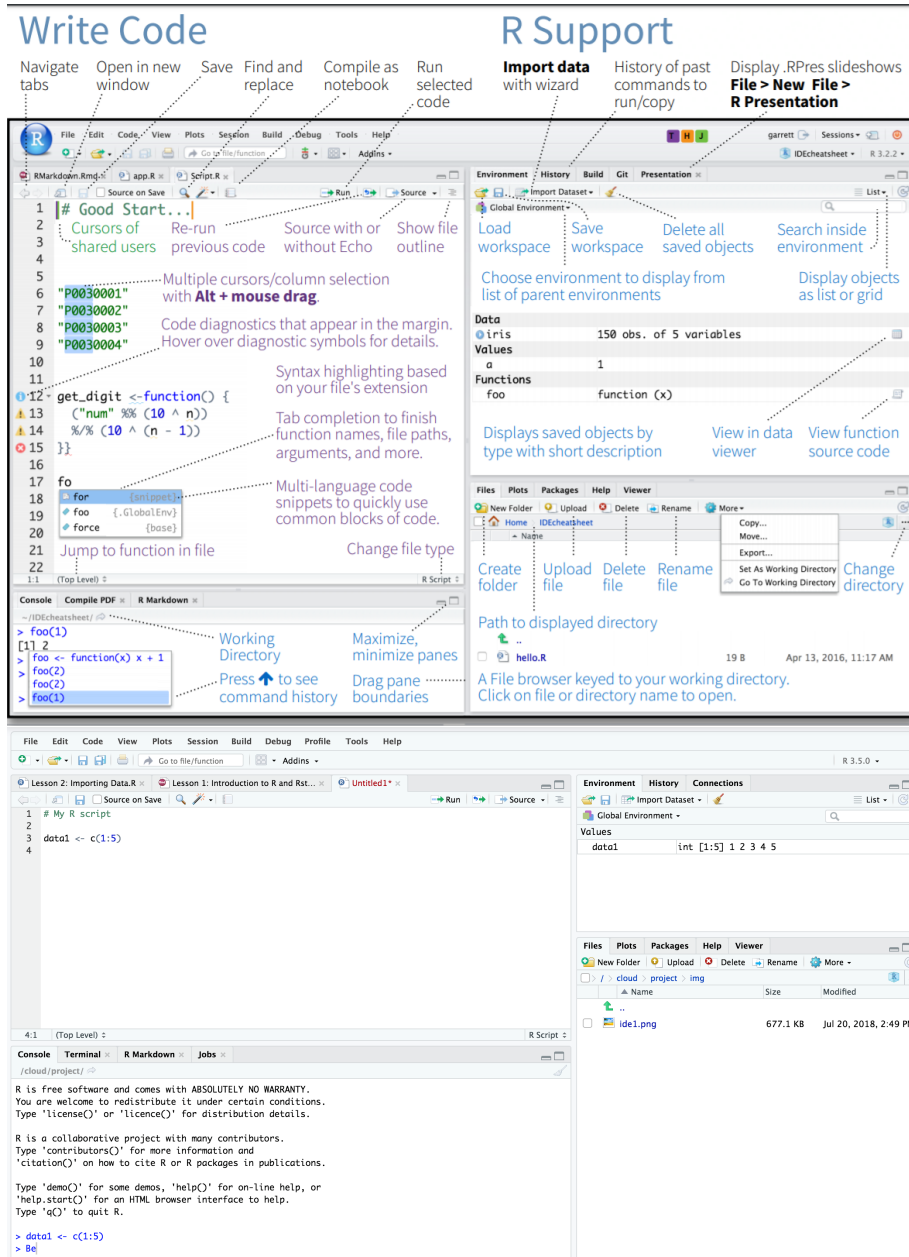
The interface for R Studio looks daunting at first. However, there are 4 main sections, 2 on the left and 2 on the right.

- MAIN TOP: R Script files or R Document Files

- – Where we usually type our code as a script before we run it. Script files are usually saved so we can work on them and rerun the code again later (.R files).
- MAIN BOTTOM: Console
  - – Shows the output of our R code. We can type R code directly into the console and the answer will ouput immediately. However, it is more convenient to use script files.
- RIGHT TOP: Environment
  - – Contains all of the objects (e.g. data, analysis, equations, plots) that are currently stored in memory. We can save all of this to a file and load it later (.RData files).
- RIGHT BOTTOM: File Browser
  - – The folder that R is working from is called 'the working directory' and it will automatically look for files there if we try to import something (e.g. a data file). Using the more button on the file browser allows you to set your desired working directory.

3.6 Working with a script

Scripts can be opened from the **File** menu.
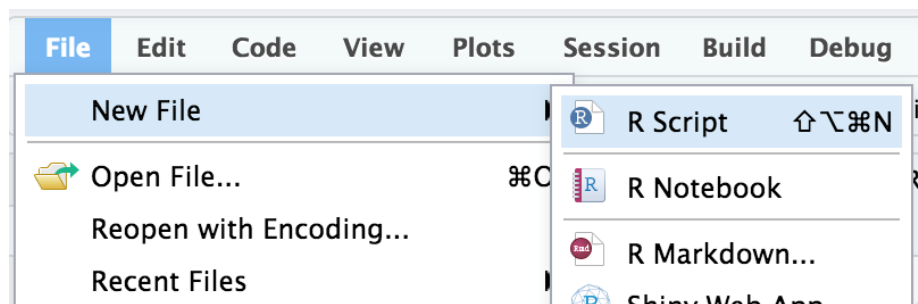
Figure 3.1: Creating a new script

The purpose of scripts is to allow you to type your analysis code and save it for use later. Scripts include, for example:

- Code for importing data into R
- Your analysis code (e.g. t-test or descriptive statistics)
- Code for graphs and tables
- Comments and notes (preceded by the '#' symbol)

To run a script, you click the **Run** button. You can choose to:

- Run the whole script
- Run the selected line of code

When you run the script, you will normally see output in the **console.**
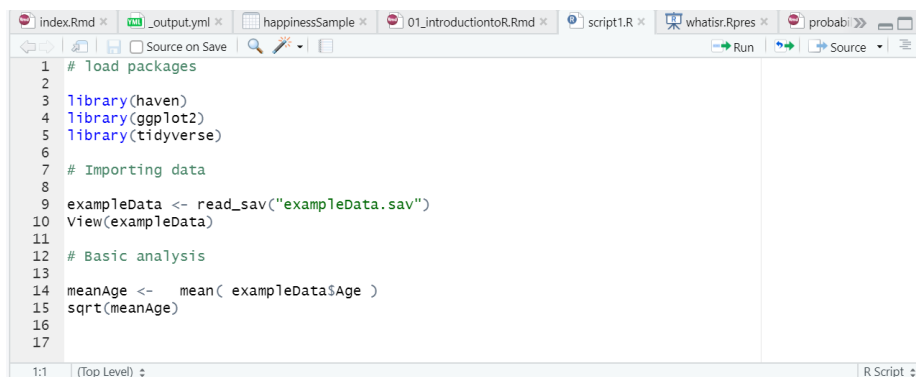
Figure 3.2: Example of an R script

Figure 3.3: The run button

```
Console   Terminal ×   Jobs ×                                                            ▬ ☐
C:/Users/wilso/OneDrive - Teesside University/Research Methods Book/Book/
> describe(happinessSample)
              vars  n mean    sd median trimmed  mad min max range skew kurtosis    se
participant      1 20 10.5 5.92   10.5   10.50 7.41   1  20    19 0.00    -1.38 1.32
happiness        2 20 18.4 4.37   16.5   18.25 4.45  12  27    15 0.34    -1.32 0.98
intervention*    3 20  1.5 0.51    1.5    1.50 0.74   1   2     1 0.00    -2.10 0.11
> |
```
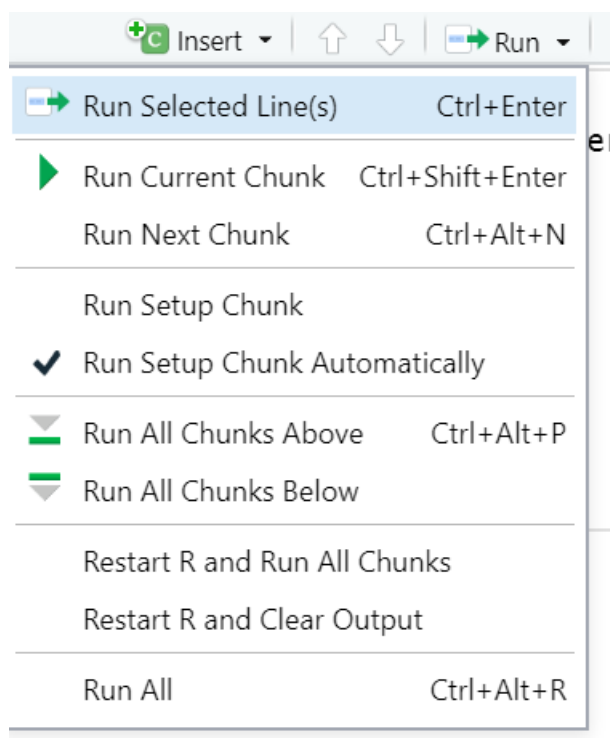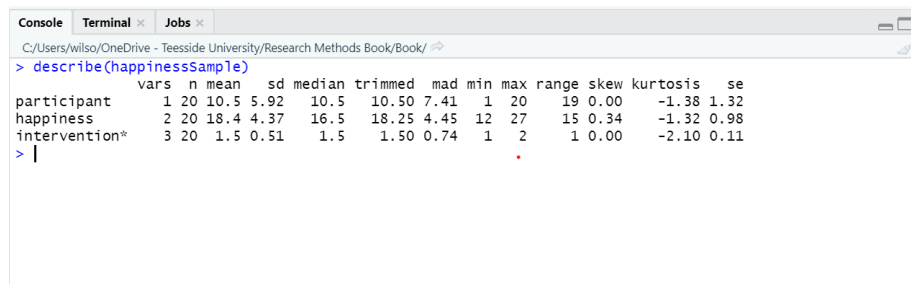
Figure 3.4: Output appears in the console

If your script contains code for a plot (graph), it will appear in the **Plots** window in the bottom right.

3.7   Installing and loading packages

install Packages from RStudio, Inc. on Vimeo.

Packages add functionality to R and allow us to do new types of analysis.

- They can be installed via the menu (Tools -> Install Packages)

- The can also be installed using code:

      install.packages()

For example, TidyR is a package that contains functions for sorting and organising data. To install the package:

or use the code:

```
install.packages("tidyr")
```

Once a package is has been installed, you need tp load it using the *library()* command. For example:

```
library("tidyr")
```
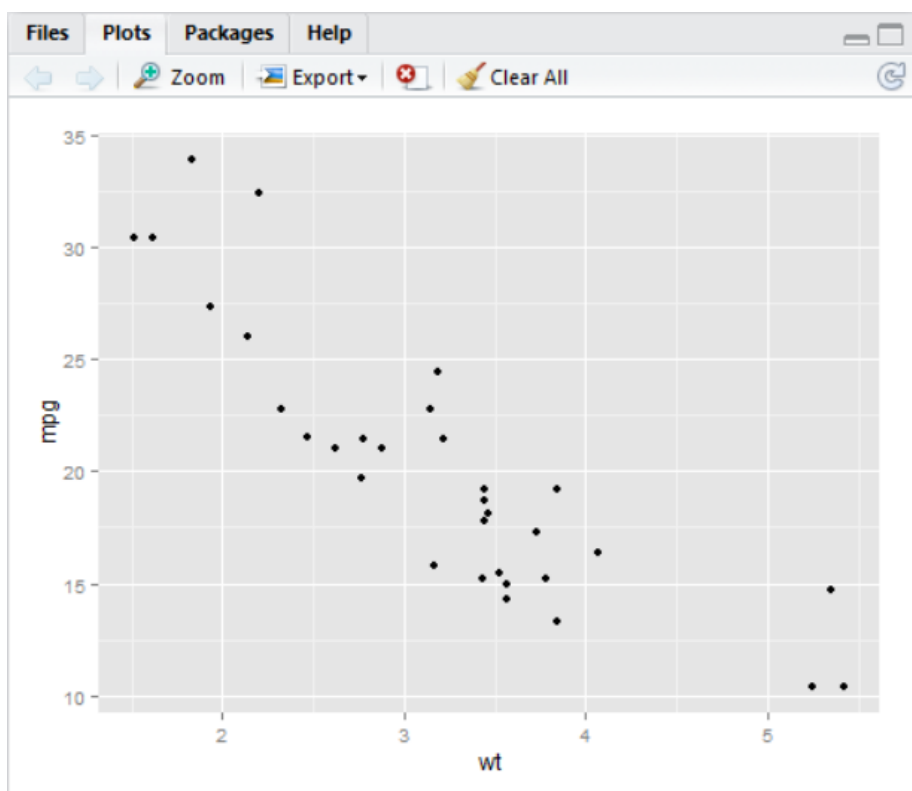
Figure 3.5: Plots appear in the plot window
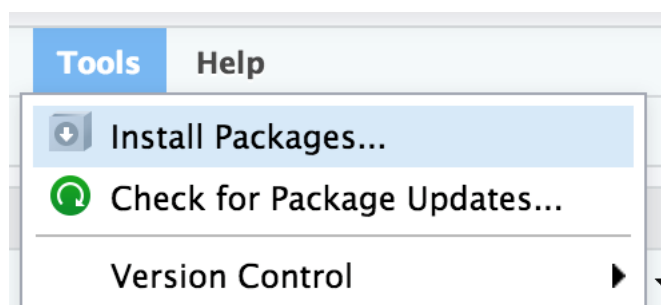
Figure 3.6: Installing a package in RStudio

Chapter 4

Working with data in R

4.1 By the end of this section, you will be able to:

- Import data into R from excel, SPSS and csv files
- Save data to objects
- Identify different data structures and variable types
- Convert variables from one type to another
- Order, filter and group data
- Summarise data
- Create new variables from data

4.2 In this section, we will use the Tidyverse set of packages

- A 'toolkit' of packages that are very useful for organsing and manipulating data
- We will use the haven package to import SPSS files
- We will use the dplyr to organise data
- Also includes the ggplot2 and tidyR packages which we will use later

To install:

```
install.packages("tidyverse")
```

(See the previous section on installing packages)

4.3   Import data into R from excel, SPSS and csv files

We can import data from a range of sources using the **Import Dataset** button in the **Environment** tab:
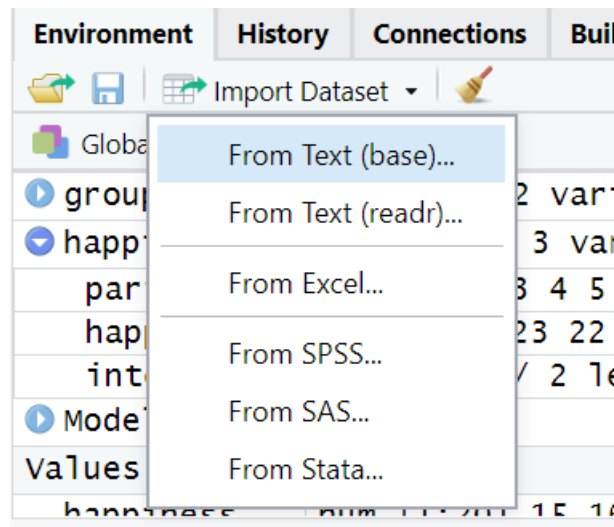
Figure 4.1: Importing data

It is also possible to import data using code, for example:
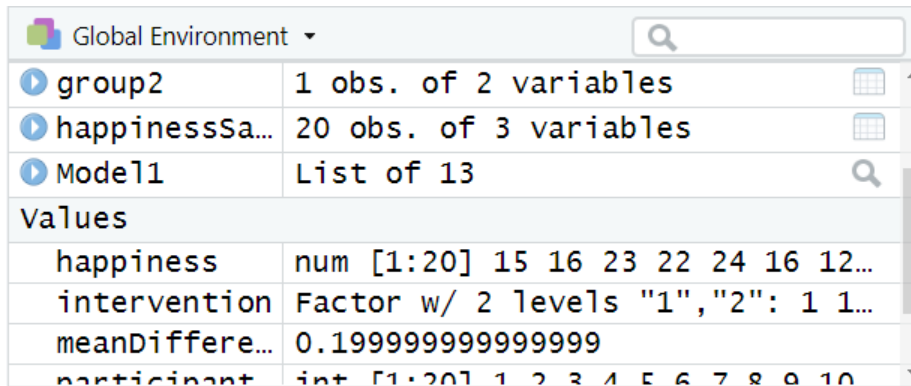
```
# importing a .csv file

library(readr)
studentData <- read_csv("Datasets/studentData.csv")

#importing an SPSS file

library(haven)
mySPSSData <- read_sav("Datasets/salesData.sav")
```

Once the data are imported, it will be visible in the environment:

Figure 4.2: Imported data in the environment

4.4 Restructuring and reorganising data in R (long versus wide data)

4.5 Understanding objects in R

In R, an **object** is anything that is saved to memory. For example, we might do some analysis:

```
mean(happiness)
```

However, in the example above, the result would appear in the console but not be saved anywhere. To store the result for reuse later, we save it to an object:

```
happinessMean <- mean(happiness)
```

In the above code (reading left to right):

- We name the object "happinessMean." This name can be anything we want.
- The arrow means that the result of the code on the right will be saved to the object on the left.
- The code on the right of the arrow calculates the mean of *happiness* data
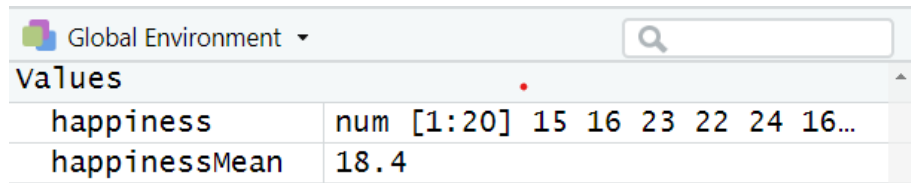
When this code is run, *happinessMean* will be stored in the environment window:

To recall an object from the environment, we can simply type its name. For example:

```
happinessMean
```

```
## [1] 10.03557
```

Its important to note that anything can be stored as an object in R

Figure 4.3: Result of a calculation in the environment

and recalled later. This includes, dataframes, the results of statistical calculations, plots etc.

4.6 Identify different data structures and variable types

4.6.1 Data structures

There are many different types of data that R can work with. The most common type of data for most people tends to be a **dataframe**. A **dataframe** is what you might consider a "normal" 2-dimensional dataset, with rows of data and columns of variables:

	participant	happiness	intervention
1	1	15	1
2	2	16	1
3	3	23	1
4	4	22	1
5	5	24	1
6	6	16	1
7	7	12	1

Figure 4.4: A dataframe example

R can also use other data types.

A vector is a one-dimensional set of values:

```
# a vector example
```

```
scores <- c(1,4,6,8,3,4,6,7)
```

A matrix is a multi-dimensional set of values. The below example is a 3-dimensional matrix, there are 2 groups of 2 rows and 3 columns:

```
## , , 1
##
##      [,1] [,2] [,3]
## [1,]    1    3    5
## [2,]    2    4    6
##
## , , 2
##
##      [,1] [,2] [,3]
## [1,]    7    9   11
## [2,]    8   10   12
```

> We will primarily work with dataframes (and sometimes vectors), as this is how the data in psychology research is usually structured.

4.6.2 Variable types

With numerical data, there are 4 key data types:

- Nominal (a category, group or factor)
- Ordinal (a ranking)
- Interval (scale data that can include negative values)
- Ratio (scale data that cannot include negative values)

Collecting data – main levels of data

- There are four different levels of numerical data:

Nominal	Ordinal	Interval	Ratio
• Categories • Can be counted • Cannot be ranked • Cannot be measured • Male/Female. Old/Young, Yes/No	• Ranks • Can be counted • Can be ranked • Cannot be measured • 1st, 2nd, 3rd	• Scale with exact values • Can be counted • Can be ranked • Can be measured • Can go below zero • E.g. temperature or difference score	• Scale with exact values • Can be counted • Can be ranked • Can be measured • Cannot go below zero • E.g. A real number (time, count)

R can use all of these variable types:

- **Nominal** variables are called **factors**
- **Ordinal** variables are called **ordered factors**
- **Interval and ratio** variables are called **numeric** data and can sometimes be called integers (if they are only whole numbers) or doubles (if they all have decimal points)

R can also use other data types such as text (**character**) data.

4.6.3   Convert variables from one type to another

When we first import data into R, it might not recognise the data types correctly. For example, in the below data, we can see the **intervention** variable :

```
##    participant intervention happiness
## 1            5            1  6.681176
## 2           11            1  6.703790
## 3            1            2  7.622706
## 4            8            1  7.895135
## 5           18            1  8.340372
## 6           19            2  8.757775
## 7           12            1  9.523169
## 8           16            1  9.741287
## 9           13            1 10.076470
## 10          10            1 10.215708
```

In the **intervention** variable, the numbers 1 and 2 refer to different intervention groups. Therefore, the variable is a **factor** variable. To ensure that R understands this, we can resave the intervention variable as a factor using the *as.factor()* function:

```
happinessSample$intervention <- as.factor(happinessSample$intervention)
```
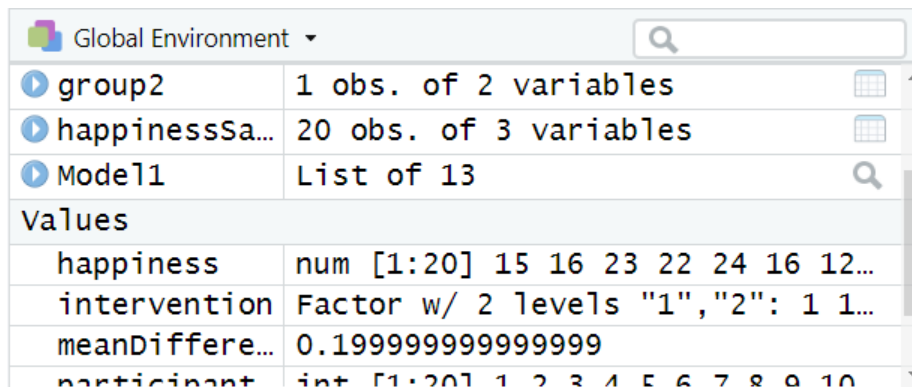
4.7 Working with dataframes

Dataframes are the more standard data format that were are used to (think of how a dataset looks in SPSS or Excel).

In a dataframe, variables are columns and each row usually reperesents one measurement or one participant.

4.7.1 View dataframe

To view a dataframe, we can click on it in the environment window and it will display:

Figure 4.5: Clicking on datasets in hte environment will open them up for viewing

4.7.2 Refer to variables (columns) in a dataframe

Columns in a dataframe are accessed using the "$" sign. For example, to access the *happiness* column in the *happinessSample* dataframe, we would type:

```
happinessSample$happiness
```

```
##  [1]  7.622706 11.298334 12.669884 12.247820  6.681176 12.089051 10.655260  7.895135 11.948733
## [12]  9.523169 10.076470 11.584766 11.656510  9.741287 10.722183  8.340372  8.757775 10.281363
```

As we can see above, the result is then displayed.

	participant	happiness	intervention
1	1	15	1
2	2	16	1
3	3	23	1
4	4	22	1
5	5	24	1
6	6	16	1
7	7	12	1

Figure 4.6: Viewing a dataframe

4.8   Order, filter and group data

If you have the **tidyverse** package loaded, it is easy to organise and filter data.

```
arrange(happinessSample, happiness)
```

```
##    participant intervention happiness
## 1            5            1  6.681176
## 2           11            1  6.703790
## 3            1            2  7.622706
## 4            8            1  7.895135
## 5           18            1  8.340372
## 6           19            2  8.757775
## 7           12            1  9.523169
## 8           16            1  9.741287
## 9           13            1 10.076470
## 10          10            1 10.215708
## 11          20            2 10.281363
## 12           7            1 10.655260
## 13          17            2 10.722183
## 14           2            1 11.298334
## 15          14            1 11.584766
## 16          15            2 11.656510
## 17           9            1 11.948733
## 18           6            1 12.089051
## 19           4            2 12.247820
## 20           3            2 12.669884
```

```
arrange(happinessSample, desc(happiness)) # Arrange in descending order
```

```
##    participant intervention happiness
## 1            3            2 12.669884
## 2            4            2 12.247820
## 3            6            1 12.089051
## 4            9            1 11.948733
## 5           15            2 11.656510
## 6           14            1 11.584766
## 7            2            1 11.298334
## 8           17            2 10.722183
## 9            7            1 10.655260
## 10          20            2 10.281363
## 11          10            1 10.215708
## 12          13            1 10.076470
## 13          16            1  9.741287
## 14          12            1  9.523169
## 15          19            2  8.757775
## 16          18            1  8.340372
## 17           8            1  7.895135
## 18           1            2  7.622706
## 19          11            1  6.703790
## 20           5            1  6.681176
```

- Show clients with a happiness score of less than 4

```
filter(happinessSample, happiness < 4)
```

```
## [1] participant   intervention happiness
## <0 rows> (or 0-length row.names)
```

- Show Intervention group 2 with happiness scores above 7

```
filter(happinessSample, happiness > 7 & intervention == 2)
```

```
##   participant intervention happiness
## 1           1            2  7.622706
## 2           3            2 12.669884
## 3           4            2 12.247820
## 4          15            2 11.656510
## 5          17            2 10.722183
## 6          19            2  8.757775
## 7          20            2 10.281363
```

- Group by intervention and show the mean happiness score

```
happinessSample %>% group_by(intervention) %>% summarise(mean = mean(happiness))
```

```
## # A tibble: 2 x 2
```

```
##    intervention  mean
##    <fct>        <dbl>
## 1 1              9.75
## 2 2             10.6
```

4.9   Create new variables from data

To create new variables from data, we can use the **mutate()** function.

For example, let's say we wanted to calculate the difference between each person's happiness score and the mean happiness score.

We could do the following:

```
happinessSample %>% mutate(difference = happiness - mean(happiness))
```

```
##    participant intervention happiness   difference
## 1            1            2  7.622706 -2.41286859
## 2            2            1 11.298334  1.26275885
## 3            3            2 12.669884  2.63430964
## 4            4            2 12.247820  2.21224556
## 5            5            1  6.681176 -3.35439857
## 6            6            1 12.089051  2.05347615
## 7            7            1 10.655260  0.61968566
## 8            8            1  7.895135 -2.14043925
## 9            9            1 11.948733  1.91315805
## 10          10            1 10.215708  0.18013335
## 11          11            1  6.703790 -3.33178497
## 12          12            1  9.523169 -0.51240558
## 13          13            1 10.076470  0.04089565
## 14          14            1 11.584766  1.54919162
## 15          15            2 11.656510  1.62093501
## 16          16            1  9.741287 -0.29428744
## 17          17            2 10.722183  0.68660864
## 18          18            1  8.340372 -1.69520242
## 19          19            2  8.757775 -1.27779981
## 20          20            2 10.281363  0.24578845
```

Chapter 5

Exploratory and descriptive analysis with R

5.1 Working example - record sales data

Let's import the data

```
Album_Sales <- read_csv("Datasets/album_sales.csv")
```

```
##
## -- Column specification -----------------------------------------------------------
## cols(
##   Adverts = col_double(),
##   Sales = col_double(),
##   Airplay = col_double(),
##   Attract = col_double(),
##   Genre = col_character()
## )
```

Let's look at the data

```
head(Album_Sales)
```

```
## # A tibble: 6 x 5
##    Adverts Sales Airplay Attract Genre
##      <dbl> <dbl>   <dbl>   <dbl> <chr>
## 1     10.3   330      43      10 Country
## 2    986.    120      28       7 Pop
## 3   1446.    360      35       7 HipHop
## 4   1188.    270      33       7 HipHop
## 5    575.    220      44       5 Metal
```

```
## 6   569.    170      19      5 Country
```

5.2 Let's make sure our data types are correct #1

- This variable is currently stored as charcters, not as a factor / category variable

```
str(Album_Sales$Genre)
```

```
##  chr [1:200] "Country" "Pop" "HipHop" "HipHop" "Metal" "Country" "Pop" "Pop" "Metal"
```

- We can save it as a factor

```
Album_Sales$Genre <- as.factor(Album_Sales$Genre)
```

```
str(Album_Sales$Genre)
```

```
##  Factor w/ 4 levels "Country","HipHop",..: 1 4 2 2 3 1 4 4 3 2 ...
```

5.3 Measures of central tendency

The main measures of central tendency are: - Mean - Median - Mode

5.3.1 Mean

"What is the mean of album sales?"

```
mean(Album_Sales$Sales)
```

```
## [1] 193.2
```

5.3.2 Trimmed mean

- The trimmed mean is used to reduce the influence of outliers on the summary

```
mean(Album_Sales$Sales, trim = 0.05)
```

```
## [1] 192.6667
```

5.3.3 Median

"What is the median amount of Airplay?"

```
median(Album_Sales$Airplay)
```

```
## [1] 28
```

5.3.4 Mode

"What is the most common attractiveness rating of bands?"

- The easiest way to get the mode in R is to generate a frequency table

```
table(Album_Sales$Attract)
```

```
##
##  1  2  3  4  5  6  7  8  9 10
##  3  1  1  4 17 44 73 44 12  1
```

- We can then look for the most frequently occuring response

5.4 Measures of dispresion or variance

5.4.1 Range

The range is the difference between the lowest and highest values

- You can calculate it using these values

```
max(Album_Sales$Airplay) - min(Album_Sales$Airplay)
```

```
## [1] 63
```

- Or you can use the range command to get the min and max values in one go

```
range(Album_Sales$Airplay)
```

```
## [1]  0 63
```

5.4.2 Interquartile range

- We know that the median is the "middle" of the data = 50th percentile
- The interquatile range is the difference between the values at the 25th and 75th percentiles

```
quantile( x = Album_Sales$Airplay, probs = c(.25,.75) )
```

```
##   25%   75%
## 19.75 36.00
```

- Interquartile range = 36 - 19.75 = 16.25

Sum of squares

- The difference between each value and the mean value, squared, and then summed together

```
sum( (Album_Sales$Adverts - mean(Album_Sales$Adverts))^2 )
```

```
## [1] 46936335
```

5.4.3 Variance

- Variance: Sum of sqaures divided by n-1

```r
# variance calculation
varianceAdverts <- sum( (Album_Sales$Adverts - mean(Album_Sales$Adverts))^2 ) / 199
```

5.4.4 Standard deviation

- Standard deviation is square root of the variance

```r
# sd calculation


sqrt(varianceAdverts)
```

```
## [1] 485.6552
```

- Can be calculated using the sd() command

```r
sd(Album_Sales$Adverts)
```

```
## [1] 485.6552
```

5.5 Skewness and Kurtosis

5.5.1 Assessing skewness of distribution #1

- It is possible to use graphs to view the distribution
- We will focus on graphic presentation of data next week

```r
hist(Album_Sales$Sales)
```

Histogram of Album_Sales$Sales
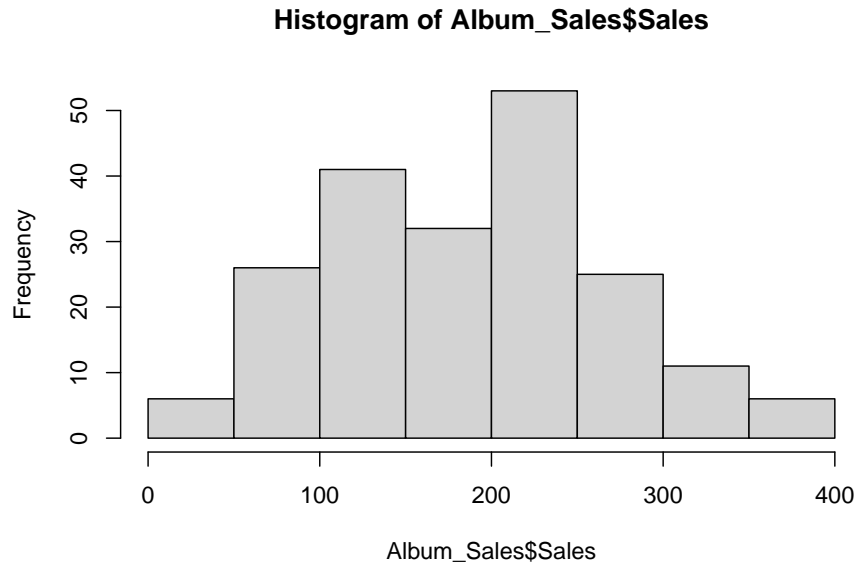
5.5.2 Assessing skewness of distribution #2

- We can check raw skewness value using the *skew()* command in the **psych** package

```
library(psych)
skew(Album_Sales$Sales)
```

```
## [1] 0.0432729
```

5.5.3 Kurtosis

informal term	technical name	kurtosis value
"too flat"	platykurtic	negative
"just pointy enough"	mesokurtic	zero
"too pointy"	leptokurtic	positive

```
kurtosi(Album_Sales$Sales)
```

```
## [1] -0.7157339
```

5.5.4 Assessing normality of distribution

- We can use the shapiro-wilk test of normality

- This is part of "base" r (no package needed)

```
shapiro.test(Album_Sales$Sales)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Album_Sales$Sales
## W = 0.98479, p-value = 0.02965
```

5.6 Getting and overall summary

5.6.1 summary() - in "base R"

```
summary(Album_Sales)
```

```
##     Adverts             Sales          Airplay          Attract           Genre
##  Min.    :   9.104   Min.   : 10.0   Min.   : 0.00   Min.    : 1.00   Country:46
##  1st Qu.: 215.918   1st Qu.:137.5   1st Qu.:19.75   1st Qu.: 6.00   HipHop :53
##  Median : 531.916   Median :200.0   Median :28.00   Median : 7.00   Metal  :48
##  Mean   : 614.412   Mean   :193.2   Mean   :27.50   Mean    : 6.77   Pop    :53
##  3rd Qu.: 911.226   3rd Qu.:250.0   3rd Qu.:36.00   3rd Qu.: 8.00
##  Max.   :2271.860   Max.   :360.0   Max.   :63.00   Max.    :10.00
```

5.6.2 describe() - in the "psych" package #1

```
describe(Album_Sales)
```

```
##           vars   n   mean     sd median trimmed    mad  min     max   range  skew kurt
## Adverts      1 200 614.41 485.66 531.92  560.81 489.09  9.1 2271.86 2262.76  0.84
## Sales        2 200 193.20  80.70 200.00  192.69  88.96 10.0  360.00  350.00  0.04
## Airplay      3 200  27.50  12.27  28.00   27.46  11.86  0.0   63.00   63.00  0.06
## Attract      4 200   6.77   1.40   7.00    6.88   1.48  1.0   10.00    9.00 -1.27
## Genre*       5 200   2.54   1.12   3.00    2.55   1.48  1.0    4.00    3.00 -0.02
```

5.6.3 describe() - in the "psych" package #2

- We can describe by factor variables

```
describeBy(Album_Sales, group = Album_Sales$Genre)
```

```
##
##  Descriptive statistics by group
## group: Country
##           vars  n   mean     sd median trimmed   mad  min     max   range  skew kurto
## Adverts      1 46 656.22 507.96 574.14  620.40 581.96  9.1 1985.12 1976.01  0.51     -0
## Sales        2 46 201.74  73.64 210.00  200.79  66.72 60.0  360.00  300.00  0.03     -0
```

```
## Airplay     3 46   29.07  10.53  28.00    28.50  11.12   9.0    54.00    45.00  0.44     -0.10  1.55
## Attract     4 46    6.52   1.63   7.00     6.71   1.48   1.0    10.00     9.00 -1.49      3.54  0.24
## Genre*      5 46    1.00   0.00   1.00     1.00   0.00   1.0     1.00     0.00  0.00       NaN   NaN  0.00
## --------------------------------------------------------------------------------
## group: HipHop
##         vars  n   mean     sd median trimmed    mad   min  max   range  skew kurtosis    se
## Adverts    1 53 606.32 452.84 601.43  568.33 501.36 10.65 2000 1989.35  0.70     0.05 62.20
## Sales      2 53 199.62  92.71 200.00  200.70 103.78 10.00  360  350.00 -0.10    -0.91 12.74
## Airplay    3 53  28.09  13.86  30.00   28.33  14.83  0.00   55   55.00 -0.14    -0.83  1.90
## Attract    4 53   6.96   1.13   7.00    7.00   1.48  3.00    9    6.00 -0.80     2.03  0.15
## Genre*     5 53   2.00   0.00   2.00    2.00   0.00  2.00    2    0.00   NaN      NaN  0.00
## --------------------------------------------------------------------------------
## group: Metal
##         vars  n   mean     sd median trimmed    mad  min     max   range  skew kurtosis    se
## Adverts    1 48 693.45 534.06  593.0  640.19 521.34 45.3 2271.86 2226.56  0.92     0.21 77.08
## Sales      2 48 197.71  75.18  200.0  198.25  88.96 40.0  340.00  300.00 -0.07    -0.94 10.85
## Airplay    3 48  27.96  11.37   27.5   28.00  11.12  2.0   57.00   55.00  0.02    -0.26  1.64
## Attract    4 48   6.85   1.34    7.0    6.90   1.48  2.0    9.00    7.00 -0.84     1.74  0.19
## Genre*     5 48   3.00   0.00    3.0    3.00   0.00  3.0    3.00    0.00   NaN      NaN  0.00
## --------------------------------------------------------------------------------
## group: Pop
##         vars  n   mean     sd median trimmed    mad   min     max   range  skew kurtosis    se
## Adverts    1 53 514.63 446.04  429.5  453.85 438.01 15.31 1789.66 1774.35  1.01     0.27 61.27
## Sales      2 53 175.28  77.92  160.0  171.86  88.96 40.00  360.00  320.00  0.34    -0.67 10.70
## Airplay    3 53  25.13  12.75   26.0   25.02  11.86  1.00   63.00   62.00  0.25     0.46  1.75
## Attract    4 53   6.72   1.47    7.0    6.81   1.48  1.00    9.00    8.00 -1.11     2.51  0.20
## Genre*     5 53   4.00   0.00    4.0    4.00   0.00  4.00    4.00    0.00   NaN      NaN  0.00
```
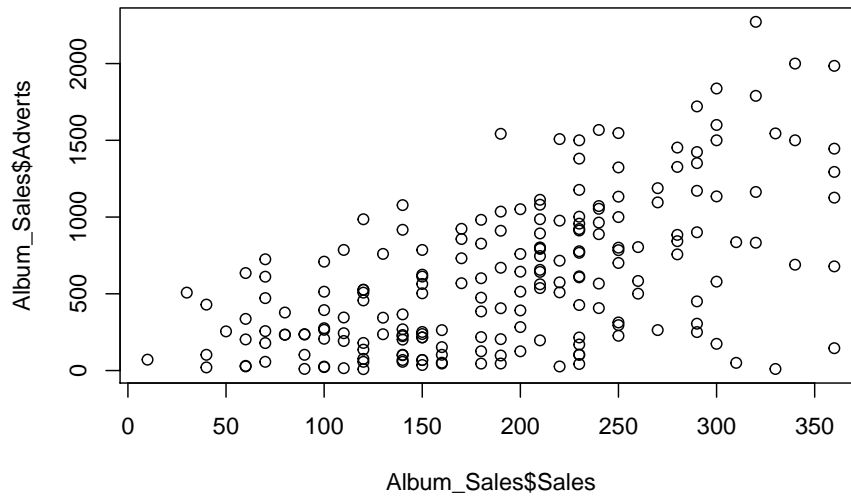
5.7 Basic statistical tests (more detail in later sections)

5.7.1 Corrleation

"Is there a relationship between advert spend and sales?"

- We would use an correlational analysis to answer this question

```
plot(Album_Sales$Sales,Album_Sales$Adverts)
```

"Is there a relationship between advert spend and sales?"

- We would use an correlational analysis to answer this question

```
cor.test(Album_Sales$Sales,Album_Sales$Adverts)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Album_Sales$Sales and Album_Sales$Adverts
## t = 9.9793, df = 198, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4781207 0.6639409
## sample estimates:
##       cor
## 0.5784877
```

5.7.2   Tests of difference - t-test

"Is there a significant difference in sales between the Country and Hip-hop musical genres?"

- We would use a t-test to answer this question

```
myTtestData <- Album_Sales %>% filter(Genre == c("Country", "HipHop"))
```

```
t.test(myTtestData$Sales ~ myTtestData$Genre)
```

```
##
##   Welch Two Sample t-test
##
## data:  myTtestData$Sales by myTtestData$Genre
## t = 0.80489, df = 40.62, p-value = 0.4256
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -27.80146  64.62904
## sample estimates:
## mean in group Country  mean in group HipHop
##               216.0000              197.5862
```

5.7.3   Tests of difference - ANOVA

"Is there a significant difference in sales between all musical genres?"

- We would use an ANOVA to answer this question

```
myAnova <- aov(Album_Sales$Sales ~ Album_Sales$Genre)
summary(myAnova)
```

```
##                     Df  Sum Sq Mean Sq F value Pr(>F)
## Album_Sales$Genre    3   23530    7843   1.208  0.308
## Residuals          196 1272422    6492
```

Chapter 6

Graphing and data visualisation with R

6.1 Presenting data visually

6.2 By the end of this section, you will be able to:

- Describe the ggplot "grammar of visualisation": coordinates and geoms
- Write a graph function to display multiple variables on a plot
- Amend the titles and legends of a plot
- Save plots in PDF or image formats

6.3 The "grammar of visualisation"

- Graphs are made up of 3 components:
    - A dataset
    - A coordinate system
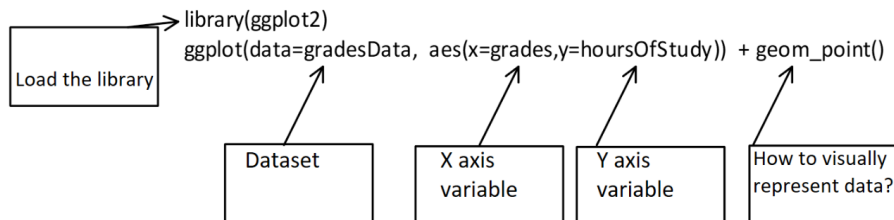   - Visual marks to represent data **(geoms)**

The "grammar of visualisation" #2
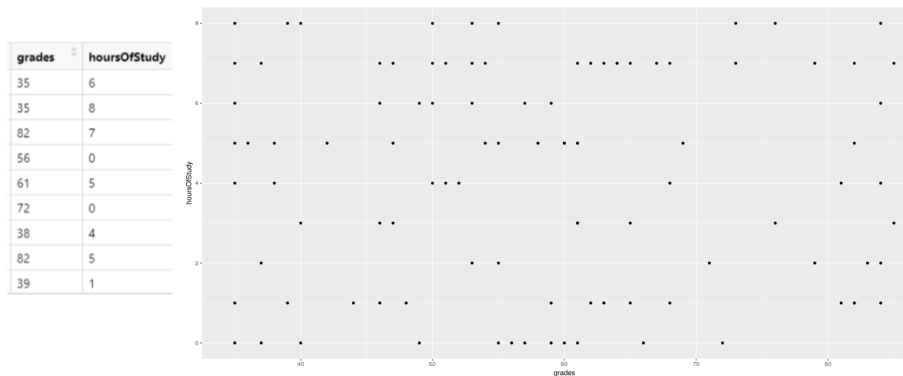
- In the above example, the dataset is the *studentData* that we used previously.
- The *grades* variable is mapped to the X axis
- The *hoursOfStudy* variable is mapped to the Y axis

6.4   How to code a graph

- The graph is created using the following code:

- In this code, we specify the dataset, the variables for the X and Y axes and the **geom** that will represent the data points visually (in this case, each datum is a point)

6.5   The graph output

```
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_point()
```
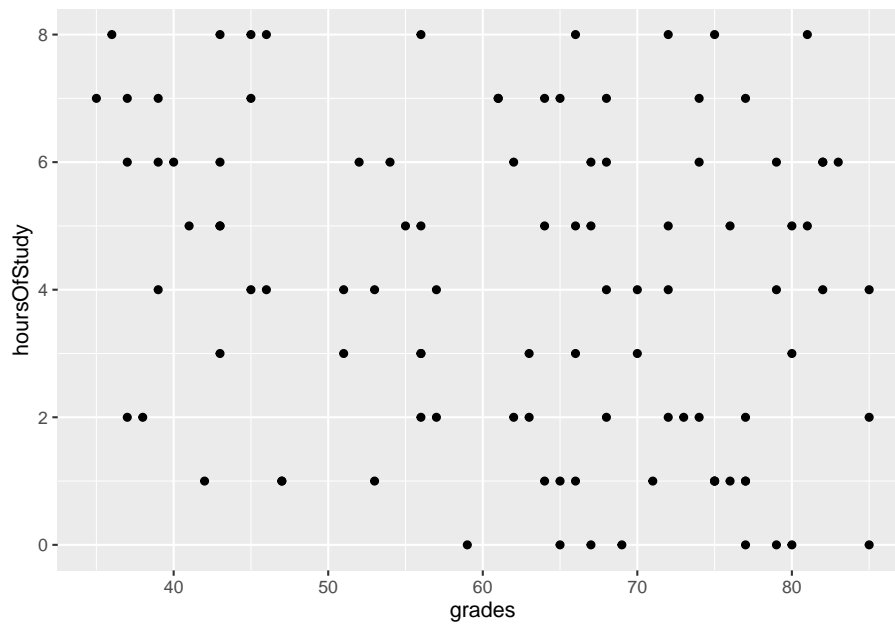
6.6 Changing the geoms leads to different visualisations

- If we change from points to lines, for example we get a different plot:

```
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_line()
```
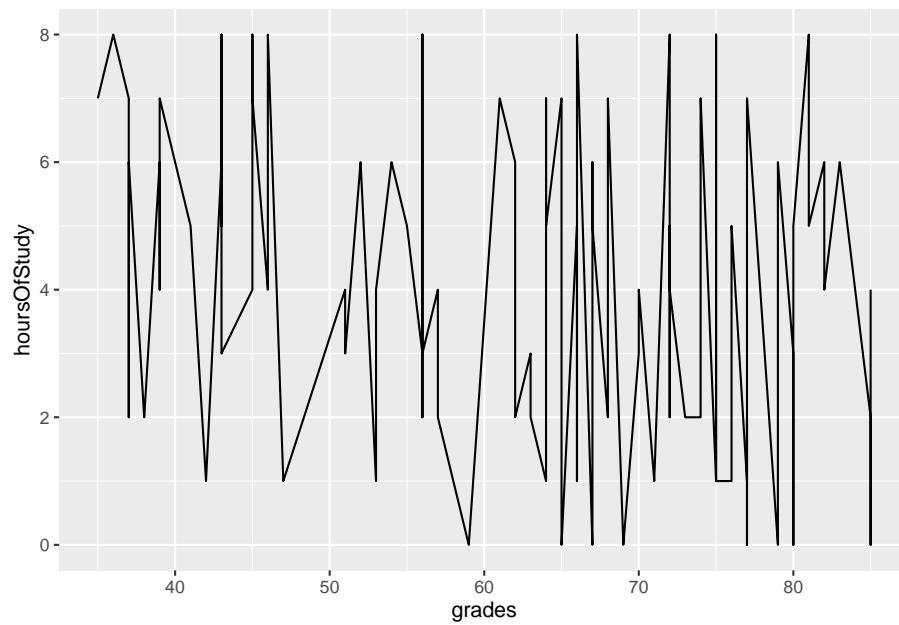
6.7 It is possible to represent more variables on the plot

- By specifying that colours of our points should be attached to the **route** variable, the data is now colour-coded

```
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_point(aes(color = route)
```

6.8 It is possible to represent more variables on the plot #2

- By specifying that size of our points should be attached to the **satisfactionLevel** variable, the size of the points adjusts

```
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_point(aes(color = route, size=satis
```

```
## Warning: Using size for a discrete variable is not advised.
```
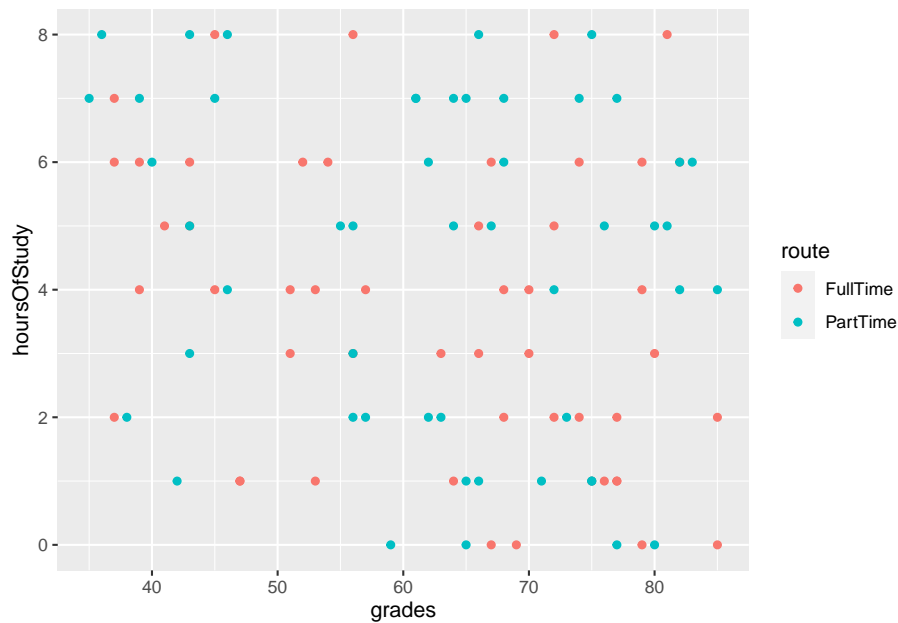
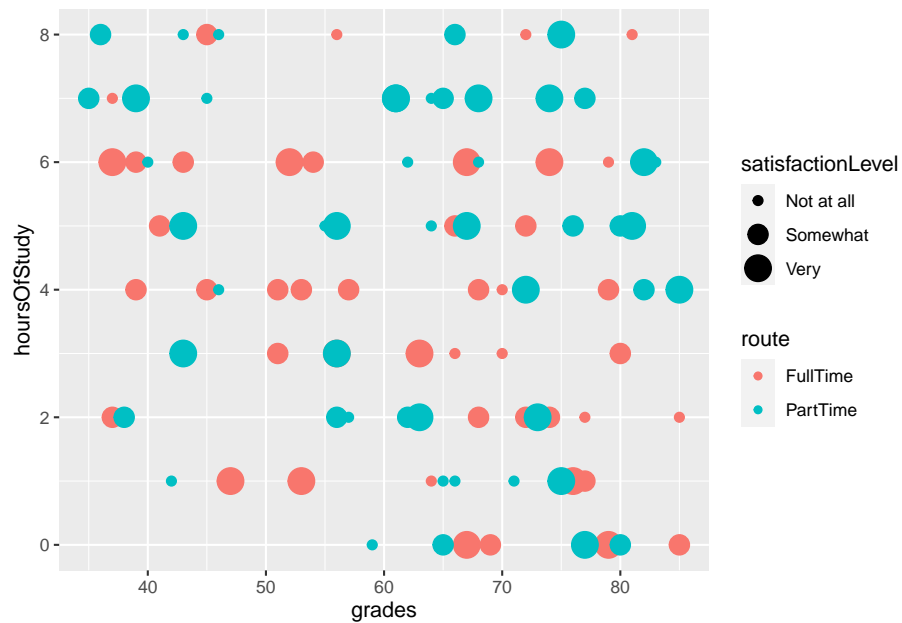## 6.9   It is possible to represent more variables on the plot #3

- By specifying that shape of our points should be attached to the **hasDe-pendents** variable, the shape of the points changes accordingly

```
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_point(aes(color = route,
```

```
## Warning: Using size for a discrete variable is not advised.
```
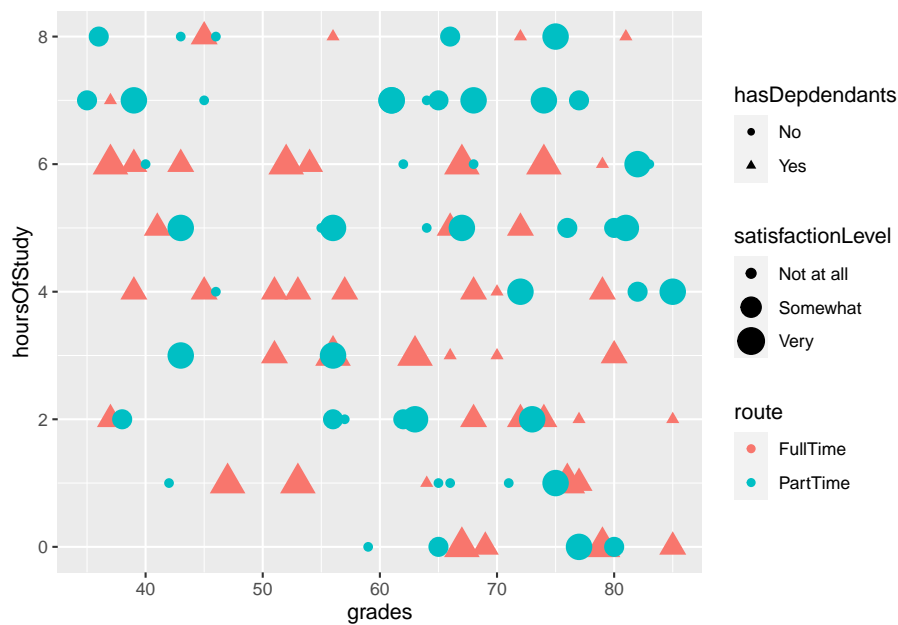
6.10   Plotting summaries of data

- We can summarise the data (e.g. get the mean or sd) using the *stat_summary()* function
- Below we are making a bar chart with the mean grade for each route

```
ggplot(data=studentData, aes(x=route, y= grades, fill=route)) + stat_summary(fun.y = "mean", geom
```

```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```

6.11   Changing the axis labels and title on a plot

We can change the axis labels and title using the **labs()** command:

```
labs(x="Student Grade", y="Hours of Study", title = "Scattterplot of student data")
library(ggplot2)

ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) + geom_point(aes(color = route,
```

```
## Warning: Using size for a discrete variable is not advised.
```

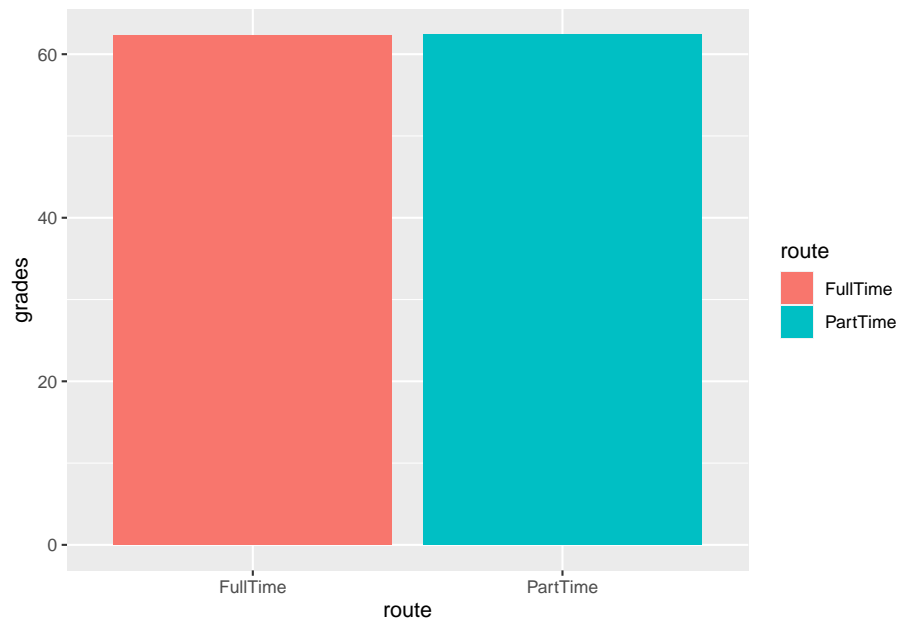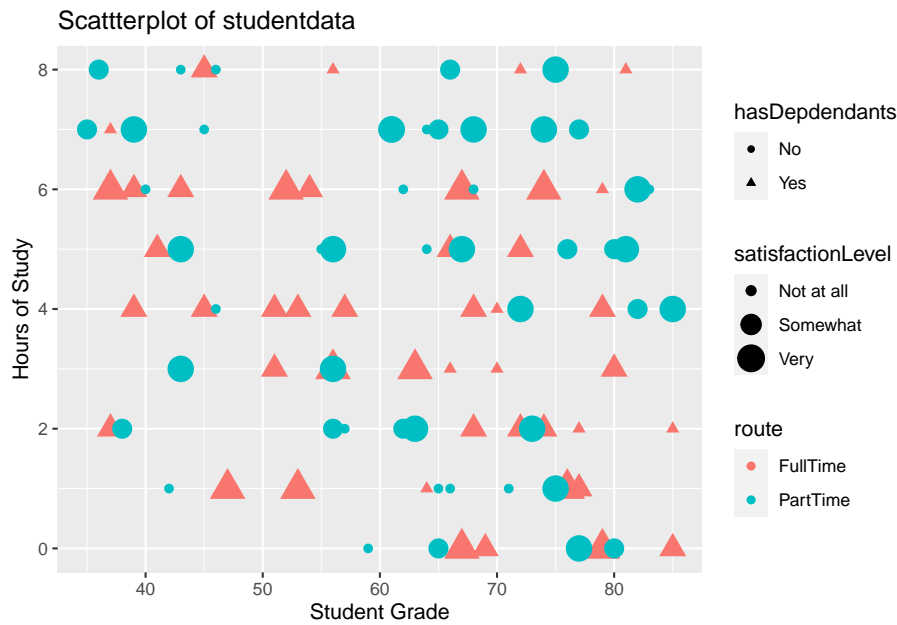## 6.12 Changing the legend on a plot

To change the legend, we use the **labs()** command too, and reference the relevant property (e.g. size, shape, colour)

```
labs(x="Student Grade", y="Hours of Study", title = "Scattterplot of student data", color="Route
library(ggplot2)

  ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) +
    geom_point(aes(color = route, size=satisfactionLevel, shape=hasDepdendants)) +
  labs(x="Student Grade", y="Hours of Study", title = "Scattterplot of studentdata", color="Route
```

```
## Warning: Using size for a discrete variable is not advised.
```
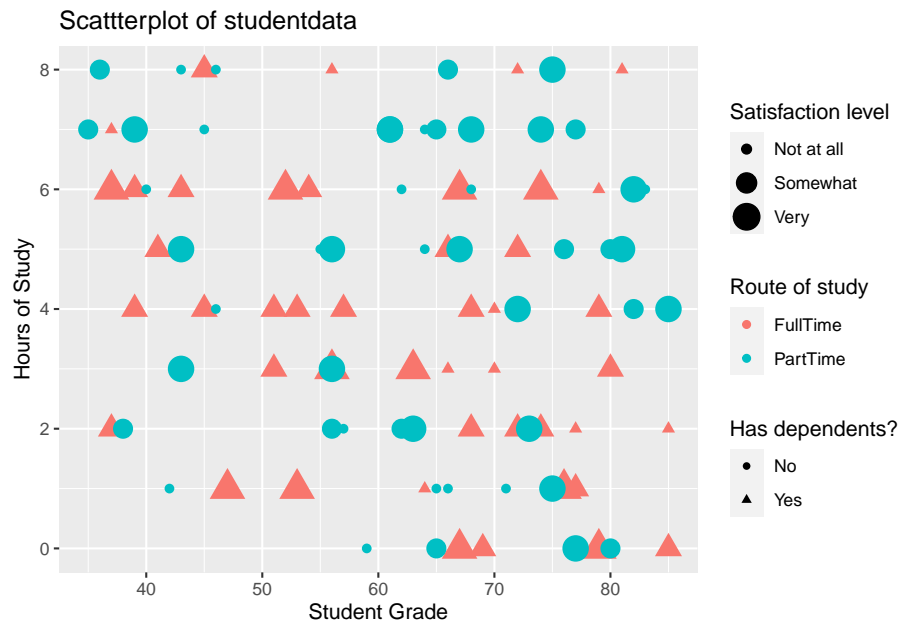
6.13   Storing plots to be recalled later

- Plots can be assigned to objects in R and recalled later, just like any other piece of data

```
library(ggplot2)

## Create plot and store it as "myPlot" object

myPlot <- ggplot(data=studentData, aes(x=grades,y=hoursOfStudy)) +
  geom_point(aes(color = route, size=satisfactionLevel, shape=hasDepdendants)) +
  labs(x="Student Grade", y="Hours of Study", title = "Scattterplot of studentdata", co
```

6.14   Recalling a stored plot

```
#Recall myPlot
myPlot
```

```
## Warning: Using size for a discrete variable is not advised.
```

Scattterplot of studentdata
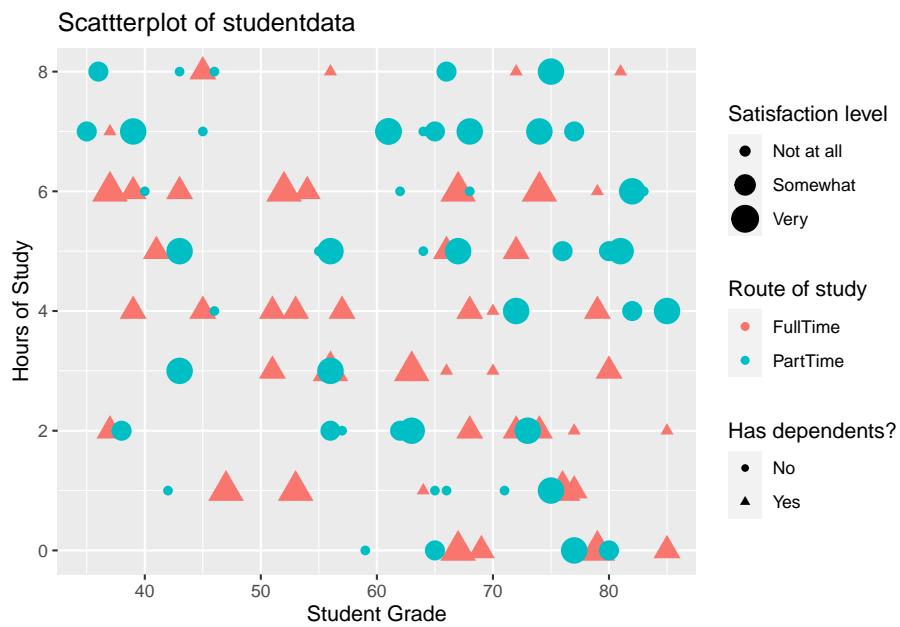
6.15 Saving plots # 1

- Plots can be save using the **export** button in the plots tab
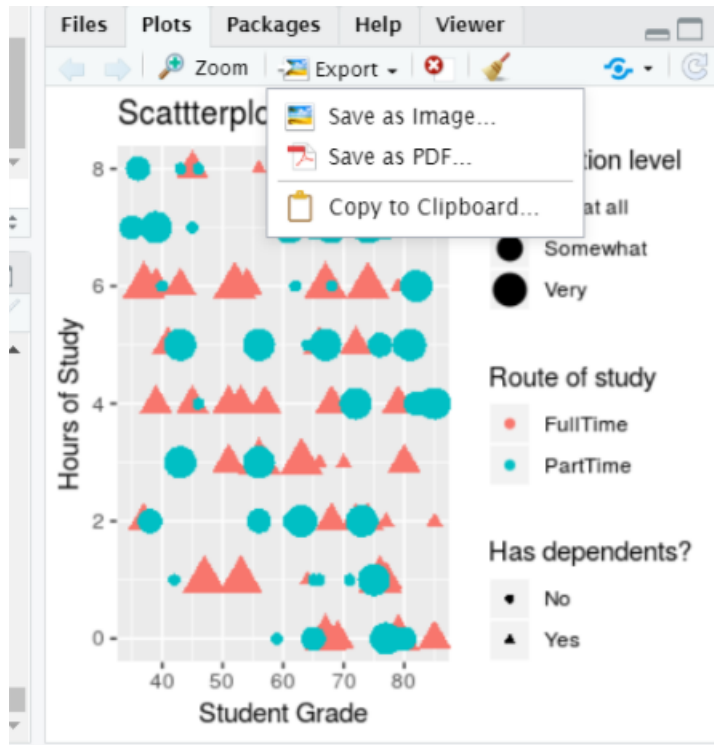
6.16   Plots can also be saved using code

- You might want to include code to save your plot in a script, for example
- This can allow greater control over the output file and plot dimensions:

```
ggsave(plot= myPlot, file="myPlot.pdf", width = 4, height = 4)
```

```
## Warning: Using size for a discrete variable is not advised.
```

```
ggsave(plot= myPlot, file="myPlot.png", width = 4, height = 4, units="cm", dpi=320)
```

```
## Warning: Using size for a discrete variable is not advised.
```

Chapter 7

Correlation

7.1 What is Correlation?

- The relationship between 2 variables
- Question: Is treatment duration related to aggression levels?

7.2 How is correlation calculated?

- Think of this as covariance divided by individual variance
- If the changes are consistent with both variables, the final value will be higher

$$r_{xy} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

② *Subtract Mean*   **③** *C*

Temp °C	Sales	"a"	"b"	a×l
14.2	$215	-4.5	-$187	84
16.4	$325	-2.3	-$77	17
11.9	$185	-6.8	-$217	1,470
15.2	$332	-3.5	-$70	245
18.5	$406	-0.2	$4	-
22.1	$522	3.4	$120	408
19.4	$412	0.7	$10	
25.1	$614	6.4	$212	1,357
23.4	$544	4.7	$142	667
18.1	$421	-0.6	$19	-1
22.6	$445	3.9	$43	168
17.2	$408	-1.5	$6	-9
18.7	**$402**			**5,325**

① *Calculate Means*

④

⑤ $$\frac{5{,}325}{\sqrt{177.0 \times 174{,}757}} = 0$$

7.3 Running correlation in R

- Step 1: Check assumptions
  - Data,distribution,linearity
- Step 2: Run correlation
- Step 3: Check R value
- Step 4: Check significance

7.3.1 Check assumptions: data

- Parametric tests require interval or ratio data
- If the data are ordinal then a non-parametric correlation is used

What type of data are treatment duration and aggression level?

7.3.2 Check assumptions: distribution

- Parametric tests require normally distributed data

Histogram of regression_data$treatment_duration

regression_data$treatment_duration

Histogram of regression_data$aggression_level

regression_data$aggression_level
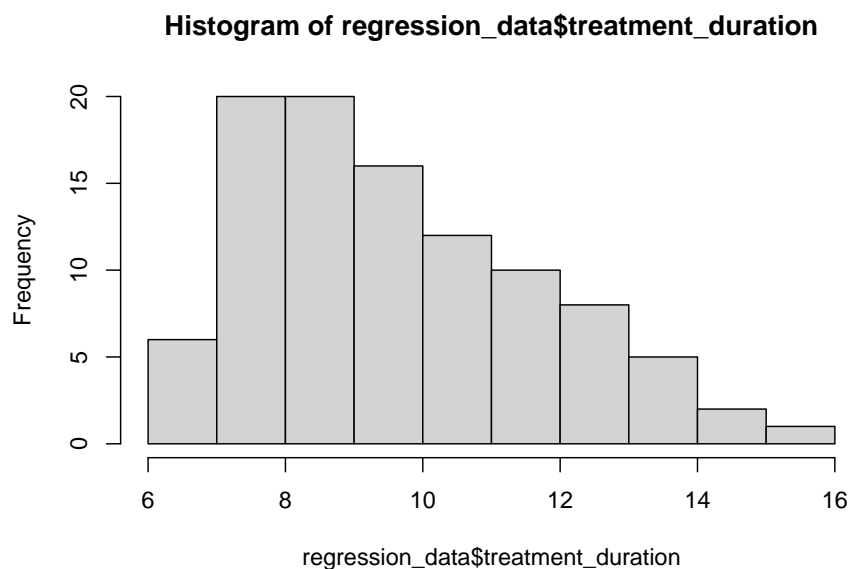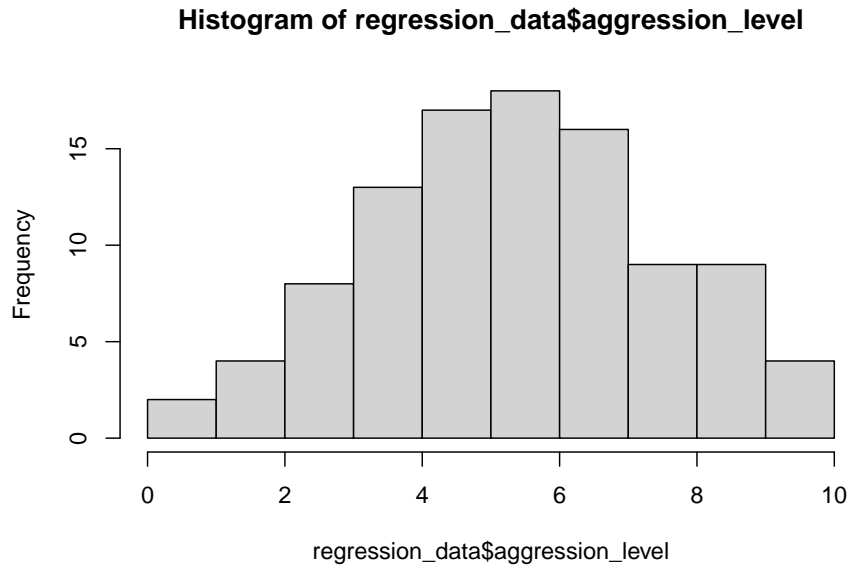
7.3.3   Check assumptions: distribution #2

- Parametric tests require normally distributed data

```
shapiro.test(regression_data$treatment_duration)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  regression_data$treatment_duration
## W = 0.94971, p-value = 0.0007939
```

```
shapiro.test(regression_data$aggression_level)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  regression_data$aggression_level
## W = 0.9928, p-value = 0.8756
```

- The normality assumption is less of an issue when sample size is > 30

7.3.4   Checking assumptions: linearity

(a) Positive linear
$r = +.82$

(b) Negative lin...
$r = -.70$

(c) Independent
$r = 0.00$

(d) Curvilin...
$r = 0.00$
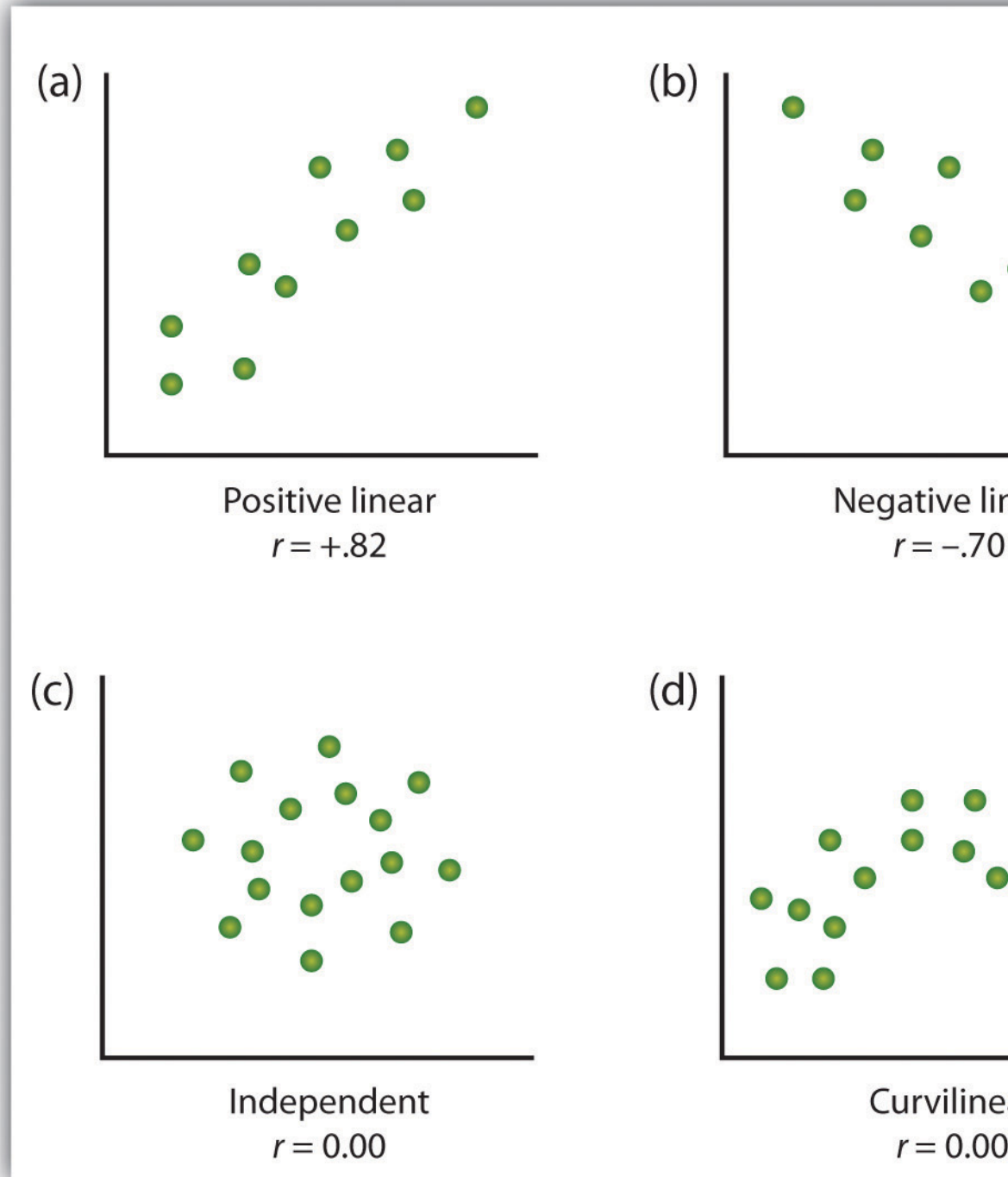
```
regression_data %>% ggplot(aes(x=treatment_duration,y=aggression_level)) +
  geom_point()
```
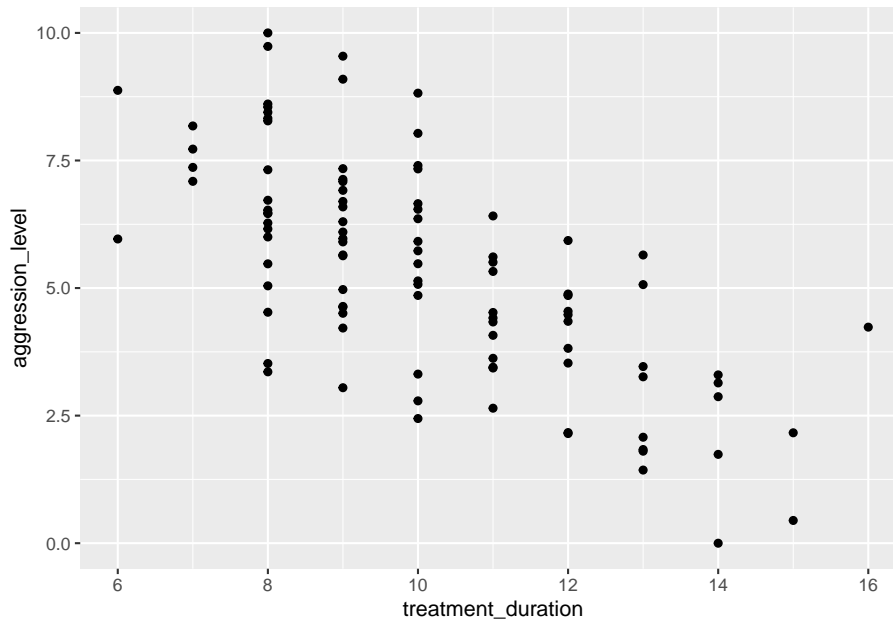

- Here we are looking to see if the relationship is linear

7.3.5 Run correlation

- R can run correlations using the *cor.test()* command

```
cor.test(regression_data$treatment_duration,regression_data$aggression_level)
```

```
##
##  Pearson's product-moment correlation
##
## data:  regression_data$treatment_duration and regression_data$aggression_level
## t = -9.5503, df = 98, p-value = 1.146e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7838251 -0.5765006
## sample estimates:
##        cor
## -0.6942996
```

7.3.6 Check r Value (correlation value)

- The r value tells us the strength and direction of the relationship

- In the output it is labelled as "cor" (short for correlation)

```
cor.test(regression_data$treatment_duration,regression_data$aggression_level)
```

```
##
##  Pearson's product-moment correlation
##
## data:  regression_data$treatment_duration and regression_data$aggression_level
## t = -9.5503, df = 98, p-value = 1.146e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7838251 -0.5765006
## sample estimates:
##        cor
## -0.6942996
```

7.3.7   Check the significance of the correlation

- We can see that the significance by looking at the p value
  - The significance is 1.146^-15
  - This means: 0.000000000000001146
- Therefore p value < 0.05

```
cor.test(regression_data$treatment_duration,regression_data$aggression_level)
```

```
##
##  Pearson's product-moment correlation
##
## data:  regression_data$treatment_duration and regression_data$aggression_level
## t = -9.5503, df = 98, p-value = 1.146e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7838251 -0.5765006
## sample estimates:
##        cor
## -0.6942996
```

Chapter 8

Simple Regression

8.1 What is regression?

- Testing to see if we can make predictions based on data that are correlated

  We found a strong correlation between treatment duration and agression levels. Can we use this data to predict aggression levels of other clients, based on their treatment duration?

- When we carry out regression, we get a information about:
  - How much variance in the **outcome** is explained by the **predictor**
  - How confident we can be about these results generalising (i.e. **significance**)
  - How much error we can expect from anu predictions that we make (i.e. **standard error of the estimate**)
  - The figures we need to calculate a predicted outcome value (i.e. **coefficient values**)

8.2   How is regression calculated?

The Best Fitting Regression Line

(a)
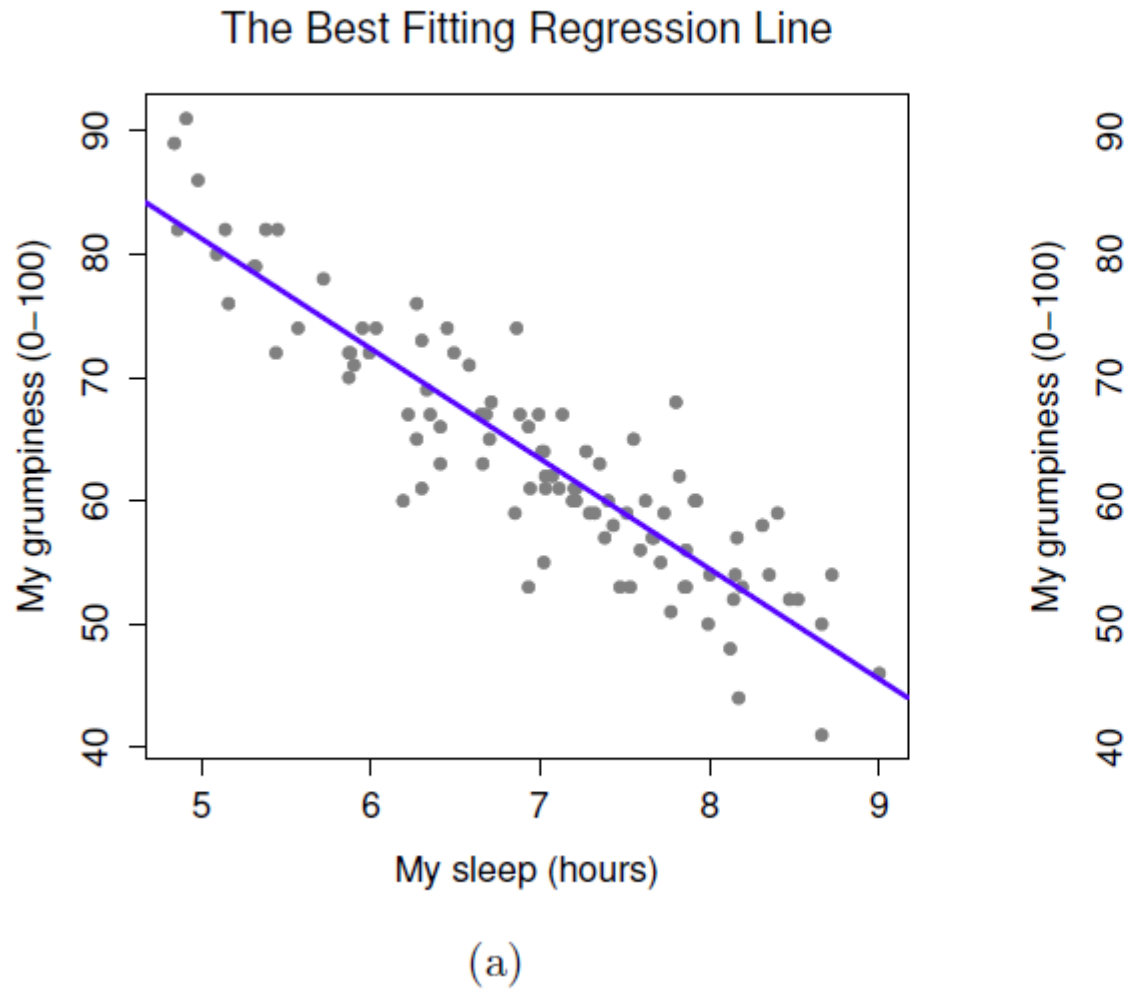
- When we run a regression analysis, a calculation is done to select the "line of best fit"
- This is a "prediction line" that minimises the overall amount of error
  - Error = difference between the data points and the line

8.3 The regression equation
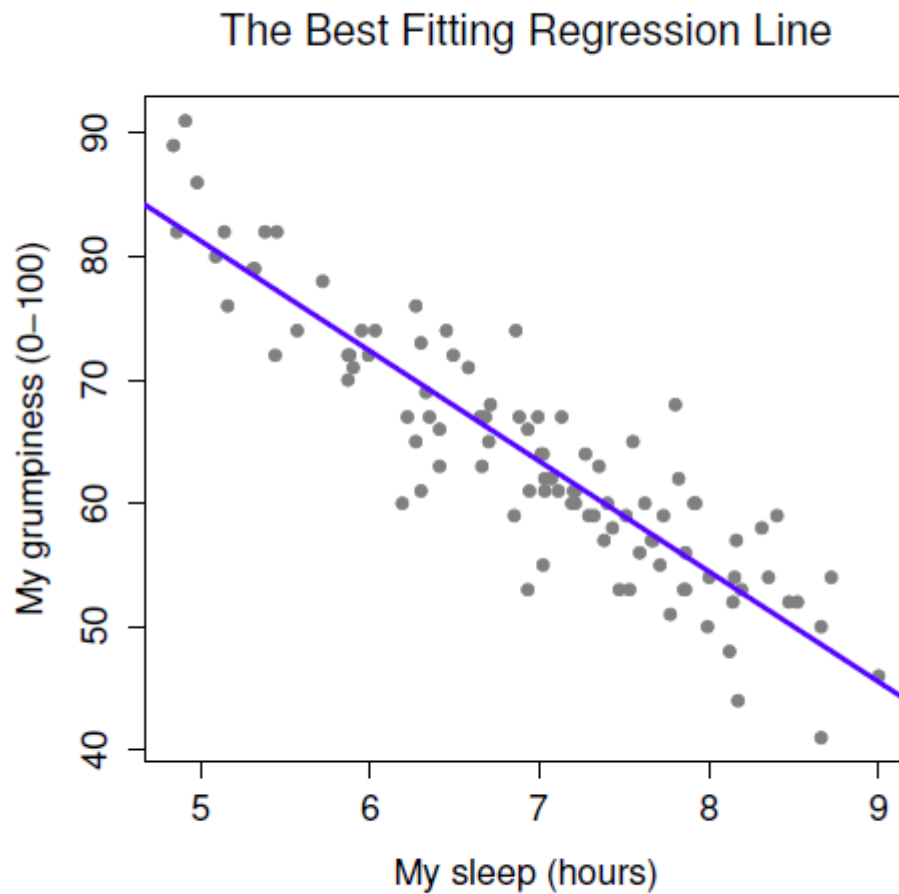
(a)
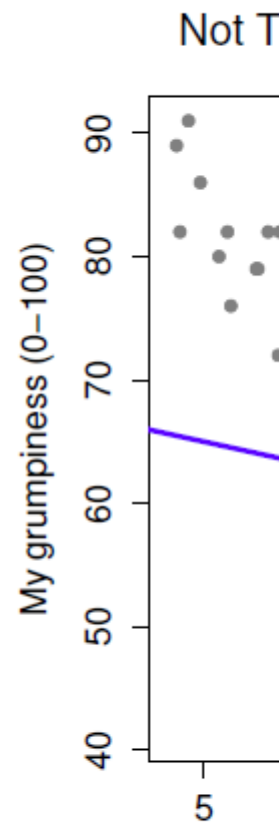
- Once the line of best fit is calculated, predictions are based on this line
- To make predictions we need the **intercept** and **slope** of the line
  - **Intercept** or **constant**= where the line crosses the y axis
  - **Slope** or **beta** = the angle of the line
- Predictions are made using the calculation for a line: $\mathbf{Y = bX + c}$

- You can think of the equation like this:

predicted outcome value = beta coefficient * value of predictor + constant

8.4   Running regression in R

- Step 1: Run regression
- Step 2: Check assumptions
    - Data
    - Distribution
    - Linearity
    - Homogeneity of variance
    - Uncorrelated predictors
    - Indpendence of residuals
    - No influental cases / outliers
- Step 3: Check R^2 value
- Step 4: Check model significance
- Step 5: Check coefficient values

8.5   Run regression

- We use the *lm()* command to run regression while saving the results
- We then use the *summary()* function to check the results

```
model1 <- lm(formula= aggression_level ~ treatment_duration ,data=regression_data)
summary(model1)
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration, data = regression_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4251 -1.1493 -0.0593  0.8814  3.4542
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          12.3300     0.7509   16.42  < 2e-16 ***
## treatment_duration   -0.6933     0.0726   -9.55 1.15e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.551 on 98 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4768
## F-statistic: 91.21 on 1 and 98 DF,  p-value: 1.146e-15
```

8.6   What are residuals?

- In regression, the assumptions apply to the residuals, not the data themselves
- Residual just means the difference between the data point and the regression line
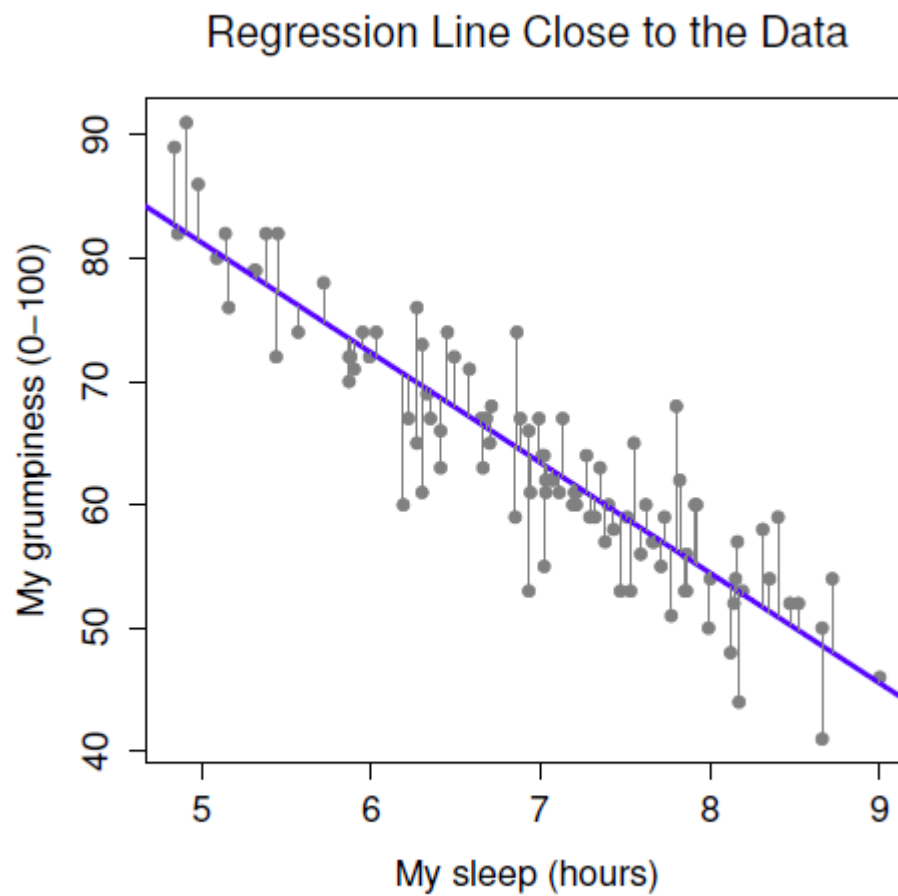
(a)
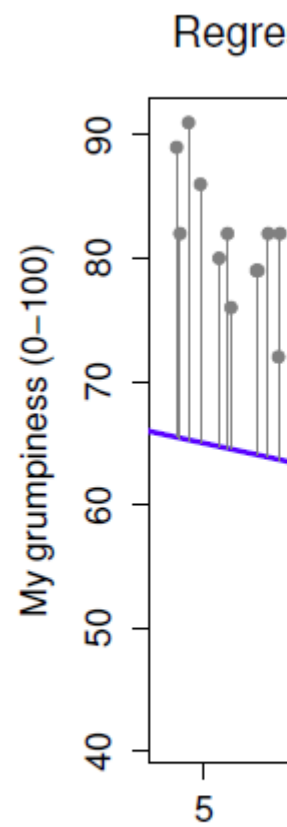
8.7   Check assumptions: distribution

- Using the *plot()* command on our regression model will give us some useful diagnostic plots
- The second plot that it outputs shows the normality

```
plot(model1, which=2)
```


Normal Q–Q

lm(aggression_level ~ treatment_duration)
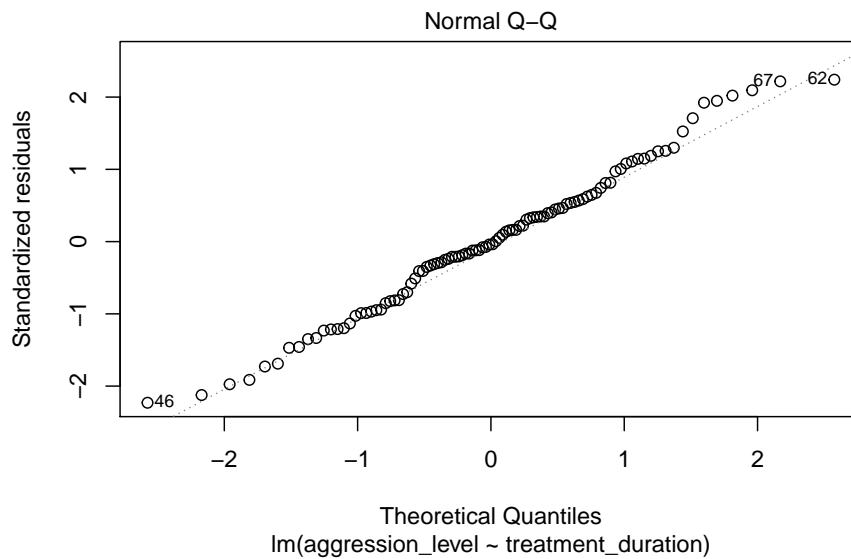
- We could also use a histogram to check the distribution
- Notice how we can use the $ sign to get the residuals from the model

```
hist(model1$residuals)
```

Histogram of model1$residuals

Frequency
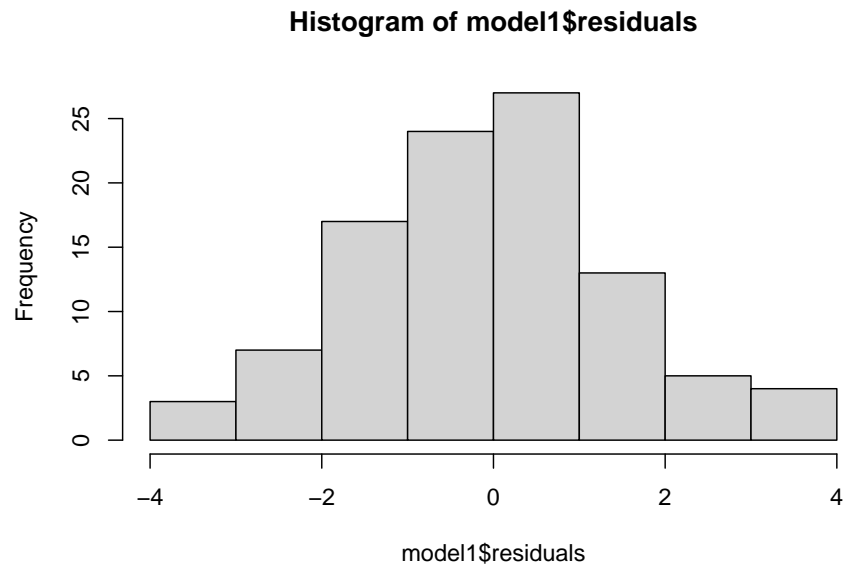
model1$residuals

8.8 Check assumptions: linearity

- Using the *plot()* command on our regression model will give us some useful diagnostic plots
- The first plot that it outputs shows the residuals vs the fitted values
- Here, we want to see them spread out, with the line being horizontal and straight

```
plot(model1, which=1)
```
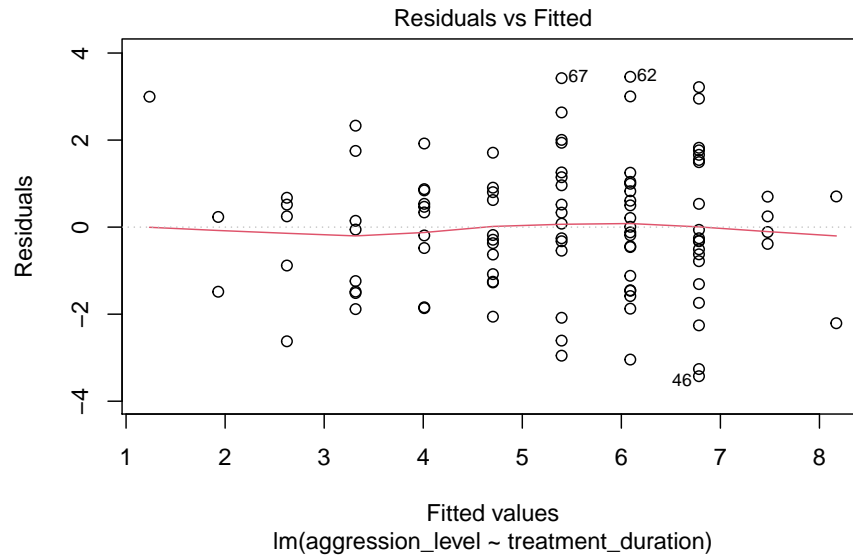
- There is a slight amount of curvilinearity here but nothing to be worried about

8.9   Check assumptions: Homogeneity of Variance #1

- We can use the sample plot to check Homogeneity of Variance
- We want the variance to be constant across the data set. We do not want the variance to change at different points in the data

```
plot(model1, which=1)
```

Residuals vs Fitted

Fitted values
lm(aggression_level ~ treatment_duration)
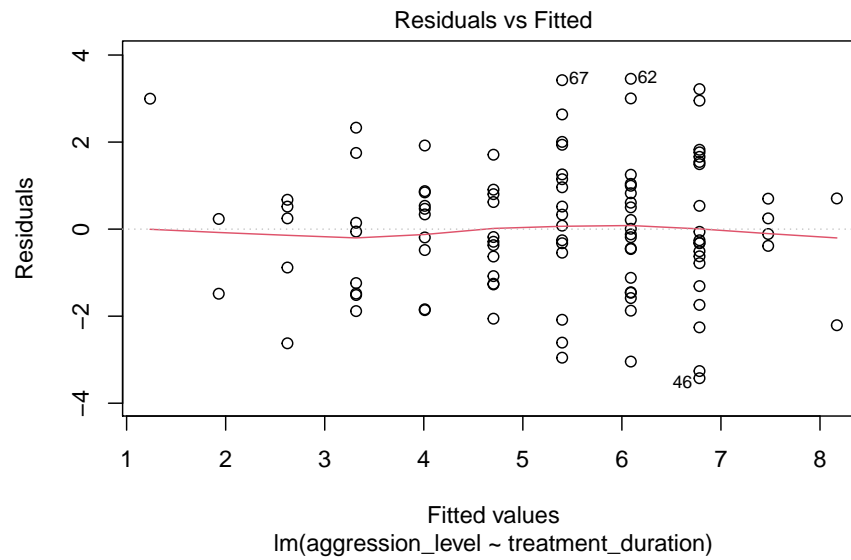
- A violation of Homogeneity of Variance would usually look like a funnel, with the data narrowing
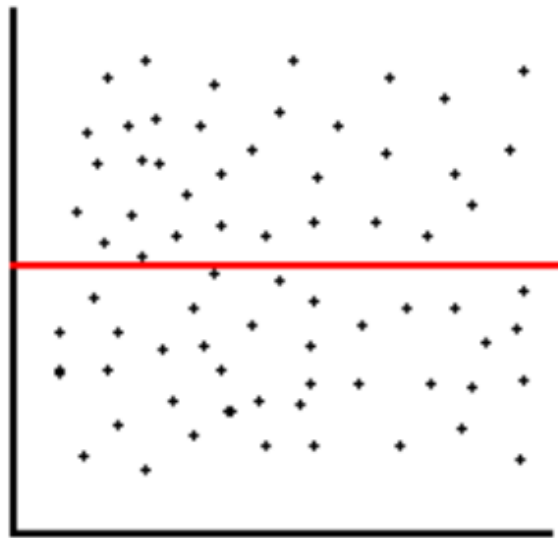
(a) Unbiased and
Homoscedastic
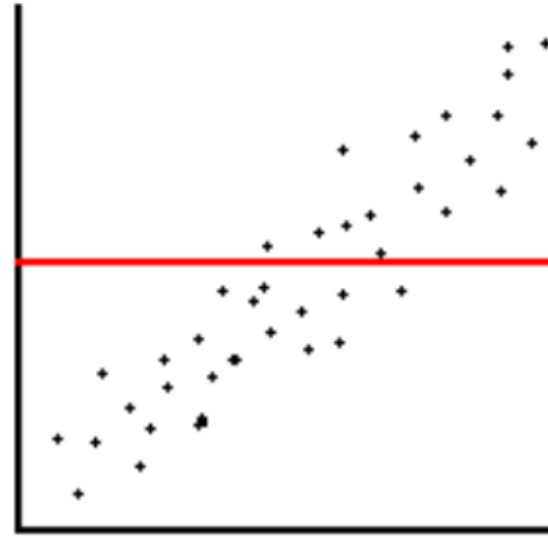
(b) Biased and
Homoscedas
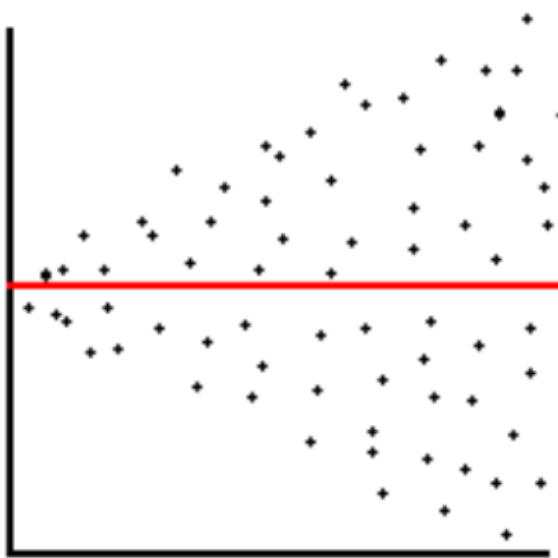
(d) Unbiased and
Heteroscedastic
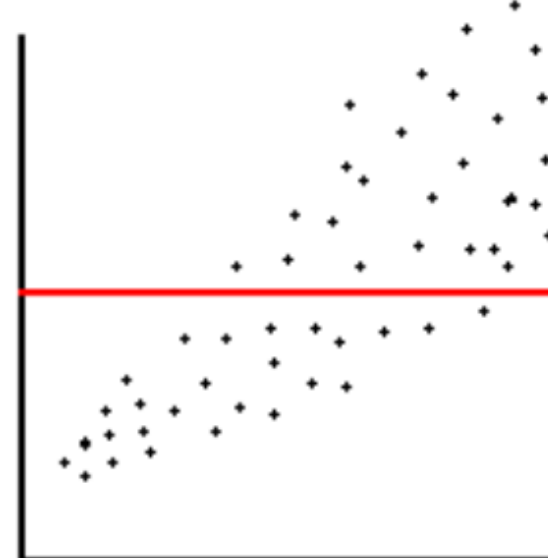
(e) Biased and
Heteroscedas

8.10 Check assumptions: Influential cases

- We need to check that there are no extreme outliers - they could throw off our predictions
- We are looking for participants that have high rediduals + high leverage
    - Some guidance suggests anything higher than 1 is an influential case
    - Others suggest 4/n is the cut off point (4 divided by number of participants)

```
plot(model1, which=4)
```
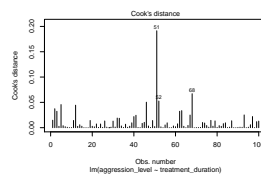

- We are looking for participants that have high rediduals + high leverage
    - No cases over 1
    - Many are over 0.04 (4/n = 0.04)

```
plot(model1, which=5)
```
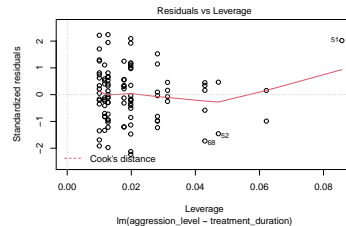

8.11 Check the r squared value

- r^2 = the amount of variance in the **outcome** that is explained by the **predictor(s)**
- The closer this value is to 1, the more useful our regression model is for predicting the outcome

```
modelSummary <- summary(model1)
modelSummary
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration, data = regression_data)
##
## Residuals:
```

```
##      Min      1Q  Median      3Q      Max
## -3.4251 -1.1493 -0.0593  0.8814  3.4542
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          12.3300     0.7509   16.42  < 2e-16 ***
## treatment_duration   -0.6933     0.0726   -9.55 1.15e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.551 on 98 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4768
## F-statistic: 91.21 on 1 and 98 DF,  p-value: 1.146e-15
```

- The r^2 of 0.482052 means that 48% of the variance in **aggression level** is explained by **treatment duration**

8.12   Check model significance

- The model significance is displayed at the very end of the output
  - *p-value: 1.146e-15*
  - As p < 0.05, the model is significant

```
modelSummary
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration, data = regression_data)
##
## Residuals:
##      Min      1Q  Median      3Q      Max
## -3.4251 -1.1493 -0.0593  0.8814  3.4542
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          12.3300     0.7509   16.42  < 2e-16 ***
## treatment_duration   -0.6933     0.0726   -9.55 1.15e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.551 on 98 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4768
## F-statistic: 91.21 on 1 and 98 DF,  p-value: 1.146e-15
```

8.13 Check coefficient values

- The coefficient values are displayed in the coefficients table
- If we have more than one predictor, they are all listed here

```
modelSummary$coefficients
```

```
##                      Estimate Std. Error    t value      Pr(>|t|)
## (Intercept)        12.3300211 0.75087601  16.420848 6.840516e-30
## treatment_duration -0.6933201 0.07259671  -9.550297 1.145898e-15
```

- The **beta coefficient** for treatment duration is in the *Estimate* column
- For every unit increase in treatment duration, aggression level decreases by 0.69

8.14 The regression equation

- The regression equation is:

Outcome = predictor value * beta coefficient + constant

- For this model, that is:

Aggression level = treatment duration * -0.69 + 12.33

```
modelSummary$coefficients
```

```
##                      Estimate Std. Error    t value      Pr(>|t|)
## (Intercept)        12.3300211 0.75087601  16.420848 6.840516e-30
## treatment_duration -0.6933201 0.07259671  -9.550297 1.145898e-15
```

8.15 Accounting for error in predictions

- We also know that the accuracy of predictions will be within a certain margin of error
- This is known as **standard error of the estimate** or **residual standard error**

```
modelSummary
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration, data = regression_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4251 -1.1493 -0.0593  0.8814  3.4542
##
## Coefficients:
```

```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       12.3300     0.7509   16.42  < 2e-16 ***
## treatment_duration -0.6933     0.0726   -9.55 1.15e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.551 on 98 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4768
## F-statistic: 91.21 on 1 and 98 DF,  p-value: 1.146e-15
```

Chapter 9

Multiple Regression

9.1 By the end of this session, you will be able to:

- Compare multiple regression to simple regression
- Describe the assumptions of multiple regression
- Consider sample size in regression
- Use categorical predictors in regression in R
- Conduct different types of multiple regression
- Interpret the output of Multiple regression

9.2 What is multiple regression?

- An extension of simple regression
- Same format as simple regression but adding each predictor:

$$Y = b_1 X_1 + b_2 X_2 + b_0$$

(The constant can be referred to in the equation as **c** or **b0**)

9.3 What are the assumptions of Multiple Regression?

- They are primarily the same as simple regression
- The additional assumption of no **multicollinearity** (due to having multiple predictors)
    - i.e. predictors should not be highly correlated

9.4   What is multicollinearity?

- Multicollinearity = predictors correlated highly with each other.
- This is not good because:
  - It makes it difficult to determine the role of individual predictors
  - Increases the error of the model (higher standard errors)
  - Difficult to identify significant predictors - wider confidence interval

9.5   Testing multicollinearity

```
## use the mctest package
# install.packages('mctest')o
library(mctest)

m1 <- lm(aggression_level ~ treatment_group + treatment_duration + trust_score, data=re

mctest(m1)


##
## Call:
## omcdiag(mod = mod, Inter = TRUE, detr = detr, red = red, conf = conf,
##      theil = theil, cn = cn)
##
##
## Overall Multicollinearity Diagnostics
##
##                     MC Results detection
## Determinant |X'X|:       0.9229        0
## Farrar Chi-Square:       7.7960        0
## Red Indicator:           0.1547        0
## Sum of Lambda Inverse:   3.1728        0
## Theil's Method:         -0.8800        0
## Condition Number:       13.6549        0
##
## 1 --> COLLINEARITY is detected by the test
## 0 --> COLLINEARITY is not detected by the test
```

- The format of *mctest()* is:

  mctest(predictors, outcome)

- In the above example we used the *cbind()* function to bind 3 columns of data together (the predictors)

9.6 Sample size for multiple regression

- Is based on the number of predictors
- More predictors = more participants needed
- **Do a power analysis**
- Loose "rule of thumb" = 10-15 participants per predictor

9.7 Approaches to multiple regression: All predictors at once

Research question: Do a client's treatment duration and treatment group predict aggression level?

```
model1 <- lm(data = regression_data, aggression_level ~ treatment_duration + treatment_group)
```

- Here we are including all of the predictors at the same time
- Note that we are using a plus sign + between each predictor
  - This means that no interactions will be tested

9.7.1 Using categorical predictors in R

- Treatment group is a categorical (also called "nominal" or "factor") variable
- No special "dummy coding" is required in R to use categorical predictors in regression
- R will use the first group as the reference category and test whether being in another group shows a significant difference
- R chooses the reference group based on numerical value or alphabetical order
- If you want you can change the reference category or "force" it using the relevel function:

```
regression_data$treatment_group <- relevel(regression_data$treatment_group, ref = "therapy1")
```

More information in categorical predictors in section 9.8

9.7.2 Reviewing the output

```
summary(model1)
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration + treatment_group,
##     data = regression_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -2.9468 -1.1104  0.0205  0.9621  3.4481
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              11.58713    0.77331  14.984  < 2e-16 ***
## treatment_duration       -0.66024    0.07119  -9.274 4.96e-15 ***
## treatment_grouptherapy2   0.85032    0.30449   2.793   0.0063 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.5 on 97 degrees of freedom
## Multiple R-squared:  0.5206, Adjusted R-squared:  0.5107
## F-statistic: 52.67 on 2 and 97 DF,  p-value: 3.267e-16
```

- Multiple R^2 = Total variance in outcome that is explained by the model
- p-value = Statistical significance of the model
- Coefficients = Contribution of each predictor to the model
  - Pr = Significance of the individual predictor
  - Estimate = Change in the outcome level that occurs when the predictor increases by 1 unit of measurement

9.7.3   All predictors at once (testing interactions)

Research questions: - Do a client's treatment duration and treatment group predict aggression level - Do the predictors interact?

```
model2 <- lm(data = regression_data, aggression_level ~ treatment_duration * treatment_
```

- Here we are including all of the predictors at the same time
- Note that we are using an asterisk * between each predictor
  - This means that interactions will be tested

Reviewing the output

```
summary(model2) %>% coefficients
```

```
##                                               Estimate Std. Error    t value      Pr(>
## (Intercept)                                 12.3529190  1.1006127 11.2236751 3.599000
## treatment_duration                          -0.7334435  0.1033086 -7.0995381 2.166226
## treatment_grouptherapy2                     -0.5615517  1.4753596 -0.3806202 7.043260
## treatment_duration:treatment_grouptherapy2   0.1394649  0.1425977  0.9780305 3.305175
```

- We get additional information in the coefficients table about the interaction between variables
  - e.g. does the interaction between level of trust and treatment duration predict the outcome (aggression level)?
- We can see from the output that none of the interactions are significant

9.7.4 Hierarchical multiple regression: Theory driven "blocks" of variables

- It might be the case that we have previous research or theory to guide how we run the analysis
- For example, we might know that treatment duration and therapy group are likely to predict the outcome
- We might want to check whether client's level of trust in the clinician has any **additional** impact on our ability to predict the outcome (aggression level)

    - To do this, we run three regression models
        - Model 0: the constant (baseline)
        - Model 1: treatment duration and therapy group
        - Model 2: treatment duration and therapy group and trust score

- We then compare the two regression models to see if:
    - Model 1 is better than Model 0 (the constant)
    - Model 2 is better than Model 1

Hierarchical multiple regression: Running and comparing 2 models

```
## run regression using the same method as above
model0 <- lm(data = regression_data, aggression_level ~ 1)
model1 <- lm(data = regression_data, aggression_level ~ treatment_duration + treatment_group)
model2 <- lm(data = regression_data, aggression_level ~ treatment_duration + treatment_group + tr

## use the aov() command to compare the models
anova(model0,model1,model2)
```

```
## Analysis of Variance Table
##
## Model 1: aggression_level ~ 1
## Model 2: aggression_level ~ treatment_duration + treatment_group
## Model 3: aggression_level ~ treatment_duration + treatment_group + trust_score
##   Res.Df    RSS Df Sum of Sq        F    Pr(>F)
## 1     99 455.27
## 2     97 218.26  2   237.013 52.2195 4.507e-16 ***
## 3     96 217.86  1     0.399  0.1757     0.676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- We can see that:
    - Model 1 (treatment duration and treatment group) is significant relative to the constant (Model 0)
    - Model 2 (treatment duration, treatment group and trust score) shows no significant change compared to Model 1

9.7.5   Stepwise multiple regression: computational selection of predictors

- Stepwise multiple regression is controversial because:
  - The computer selects which predictors to include based on Akaike information criterion (AIC)
    * This is a calculation of the quality of statistical models when they are compared to each other

9.7.6   What's the problem?

  - This selection is not based on any underlying theory or understanding of the real-life relationship between the variables

9.7.7   Stepwise multiple regression: loading the MASS package and run the full model

1. **install and load the MASS package**
2. **run a regression model with all of the variables**
3. use the *stepAIC()* command on the full model to run stepwise regression
4. View the best model

```
library(MASS)

# Run the full model
full.model <- lm(data = regression_data, aggression_level ~ treatment_duration + treatr
```

9.7.8   Stepwise multiple regression: Use stepAIC( ) with options

- **Trace** *(TRUE or FALSE)*: do we want to see the steps that were involved in selecting the best model ?
- **Direction** *("forward," "backward" or "both")*:
  - start with no variables and add them *(forward)*
  - start with all variables and subtract them *(backward)*
  - use both approaches *(both)*

```
# Run stepwise
step.model <- stepAIC(full.model, direction = "both", trace = TRUE)

## Start:  AIC=85.87
## aggression_level ~ treatment_duration + treatment_group + trust_score
##
##                     Df Sum of Sq     RSS     AIC
## - trust_score        1     0.399  218.26  84.052
## <none>                                217.86  85.869
## - treatment_group    1    17.877  235.74  91.755
```

```
## - treatment_duration  1    188.709 406.57 146.259
##
## Step:  AIC=84.05
## aggression_level ~ treatment_duration + treatment_group
##
##                     Df Sum of Sq    RSS     AIC
## <none>                            218.26  84.052
## + trust_score         1     0.399 217.86  85.869
## - treatment_group     1    17.547 235.81  89.785
## - treatment_duration  1   193.515 411.78 145.531
```

9.7.9 Stepwise multiple regression: Display the best model

1. install and load the MASS package
2. run a regression model with all of the variables
3. use the ***stepAIC()* command on the full model to run stepwise regression**
4. **View best model**

```
#view the stepwise output
summary(step.model)
```

```
##
## Call:
## lm(formula = aggression_level ~ treatment_duration + treatment_group,
##     data = regression_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9468 -1.1104  0.0205  0.9621  3.4481
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         11.58713    0.77331  14.984  < 2e-16 ***
## treatment_duration  -0.66024    0.07119  -9.274 4.96e-15 ***
## treatment_grouptherapy2  0.85032    0.30449   2.793   0.0063 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.5 on 97 degrees of freedom
## Multiple R-squared:  0.5206, Adjusted R-squared:  0.5107
## F-statistic: 52.67 on 2 and 97 DF,  p-value: 3.267e-16
```

9.8   Using regression with categorical predictors (more information)

People are often taught to use ANOVA to compare groups (i.e. if you have a categorical IV) and regression if you have continuous IVs. However, ANOVA and regression are the same thing, so it is possible to use regression to do analysis instead of ANOVA or ANCOVA.

However, it might be difficult to understand how this is, so let's look at an example. The dataset **Baumann** compares 3 different methods of teaching reading comprehension. For this example, we will just look at the variable post.test.1 as the DV.

9.8.1   ANOVA Approach

ANOVA asks the question in the following way:

> Is there a difference in reading comprehension scores between teaching groups?

The analysis takes the following approach:

- What are the means of groups 1,2 and 3?
- Are the means of groups 1,2 and 3 different?
- Is the difference in means of groups 1,2 and 3 statistically significant?

If we were to summarise the data, we might present it in the following way:

group

mean

sd

Basal

6.681818

2.766920

DRTA

9.772727

2.724349

Strat

7.772727

3.927095

In the table above we can see that the mean scores are different and highest in the DRTA group.

If we were to run an ANOVA on the data, we might present it in the following way:

term

df

sumsq

meansq

statistic

p.value

group

2

108.1212

54.06061

5.317437

0.0073468

Residuals

63

640.5000

10.16667

NA

NA

Notice that the ANOVA output tells us that the difference between groups is significant ($p < 0.05$) but we cannot tell yet which of the 3 groups are significantly different from each other.

9.8.2 Regression approach

Regression asks the question the following way:

> Does teaching group predict reading comprehension score?

The analysis takes the following approach:

- Let's use the mean of group 1 as a reference point (i.e. the intercept).
- What's the difference between the intercept and the mean scores of the other groups (i.e. the coefficients)?
- Are any of the coefficients statistically significant?

If we run a regression analysis, we might present the results like this:

R2

0.1444271

term

df

sumsq

meansq

statistic

p.value

group

2

108.1212

54.06061

5.317437

0.0073468

Residuals

63

640.5000

10.16667

NA

NA

term

estimate

std.error

statistic

p.value

(Intercept)

6.681818

0.6797950

9.829167

0.0000000

groupDRTA

3.090909

0.9613753

3.215091

0.0020583

groupStrat

1.090909

0.9613753

1.134738

0.2607841

9.8.3 Interpreting regression output

If we look at the coefficient (estimate) for the intercept (see regression output above), we can see that the value is the same as the mean of the Basal group in the previous section (See table of mean and sd, above).

Furthermore, if we look at the estimates of DRTA and Strat, we can see that the values are the difference between their mean score, and the score for of the intercept (BASAL) group. So we can see whether DTRA and STRAT groups are significantly different from the BASAL group.

If we wanted to compare the groups differently (e.g. using Strat as the reference point), we can use the relevel function and run the regression analysis again (See Using categorical predictors in R)
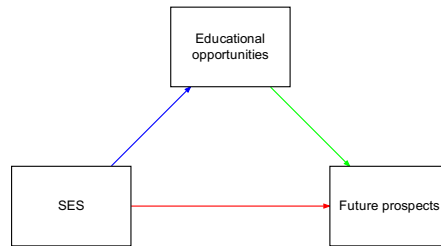
Chapter 10

Mediation analysis

10.1 Overview

- What are mediation and moderation?
- Mediation analysis example
- Packages needed
- Baron and Kenny approach in R
- Mediation package approach in R

10.2 What is mediation?

Where the relationship between a predictor (X) and an outcome (Y) is mediated
by another variable (M).

In the above model, we theorise that socio-economic status predicts education level, which predicts future prospects.

10.3   What is moderation?

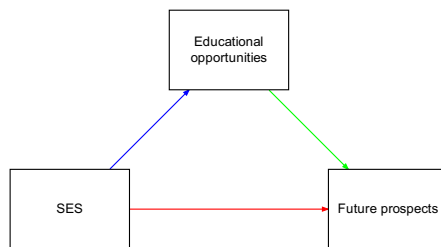There is a direct relationship between X and Y but it is affected by a moderator (M)
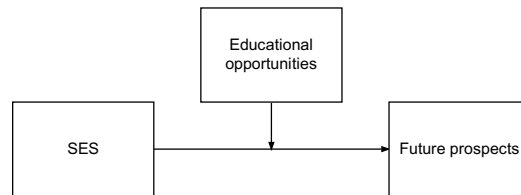
In the above model, we theorise that socio-economic status predicts future prospects but the strength of the relationship is changed by education level

10.4   Why different models?

This might be more appropriate if higher education costs money

This might be more appopriate if access to higher education is free

10.5   Mediation analysis

10.5.1   What is a mediation design?

Whether a mediation analysis is appropriate is determined as much by the design as by statistical criteria.
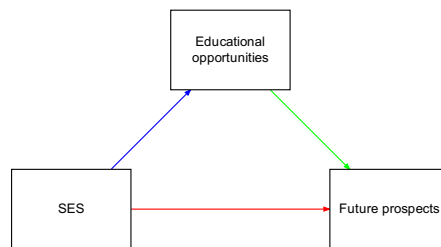
We must consider whether it makes sense to predict this relationship between variables

10.5.2   What is mediation analysis?

- Based on regression

A summary of the logic of mediation:

- The direct relationship between X and Y should be significant
- The relationship between X and M should be significant
- The relationship between M and Y (controlling for X) should be significant
- When controlling for M, the strength of the relationship between X and Y decreases and is **not** significant
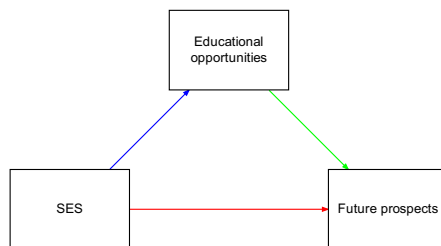
- The direct relationship between X and Y should be significant
- The relationship between X and M should be significant
- The relationship between M and Y (controlling for X) should be significant
- When controlling for M, the strength of the relationship between X and Y decreases and is **not** significant

  Baron & Kenny (1986) originally used a 4-step regression model to test each of these relationships.

10.5.3   What packages do we need?

```
library(mediation) #Mediation package

library(multilevel) #Sobel Test

library(bda) #Another Sobel Test option

library(gvlma) #Testing Model Assumptions

library(stargazer) #Handy regression tables
```

10.6   Mediation analysis (the Baron and Kenny Approach)

10.6.1   Conducting mediation analysis (the Baron and Kenny Approach)

- Baron & Kenny (1986) originally used a 4-step regression model to test each of these relationships.
- The sobel test is then used to test the significance of mediation

10.6.2   Step 1: Total Effect

```
#1. Total Effect
fit <- lm(Y ~ X, data=Meddata)
summary(fit)
```

```
##
## Call:
## lm(formula = Y ~ X, data = Meddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.917  -3.738  -0.259   2.910  12.540
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 19.88368   14.26371   1.394   0.1665
## X            0.16899    0.08116   2.082   0.0399 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.16 on 98 degrees of freedom
## Multiple R-squared:  0.04237,    Adjusted R-squared:  0.0326
```

```
## F-statistic: 4.336 on 1 and 98 DF,  p-value: 0.03993
```

10.6.3   Step 2: Path A (X on M)

```
#2. Path A (X on M)
fita <- lm(M ~ X, data=Meddata)
summary(fita)
```

```
##
## Call:
## lm(formula = M ~ X, data = Meddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5367 -3.4175 -0.4375  2.9032 16.4520
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.04494   13.41692   0.451    0.653
## X            0.66252    0.07634   8.678 8.87e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.854 on 98 degrees of freedom
## Multiple R-squared:  0.4346, Adjusted R-squared:  0.4288
## F-statistic: 75.31 on 1 and 98 DF,  p-value: 8.872e-14
```

10.6.4   Step 3: Path B (M on Y, controlling for X)

```
#3. Path B (M on Y, controlling for X)
fitb <- lm(Y ~ M + X, data=Meddata)
summary(fitb)
```

```
##
## Call:
## lm(formula = Y ~ M + X, data = Meddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3651 -3.3037 -0.6222  3.1068 10.3991
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.32177   13.16216   1.316    0.191
## M            0.42381    0.09899   4.281 4.37e-05 ***
```

```
## X              -0.11179    0.09949  -1.124     0.264
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.756 on 97 degrees of freedom
## Multiple R-squared:  0.1946, Adjusted R-squared:  0.1779
## F-statistic: 11.72 on 2 and 97 DF,  p-value: 2.771e-05
```

10.6.5   Step 4: Reversed Path C (Y on X, controlling for M)

```
#4. Reversed Path C (Y on X, controlling for M)
fitc <- lm(X ~ Y + M, data=Meddata)
summary(fitc)
```

```
##
## Call:
## lm(formula = X ~ Y + M, data = Meddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.438  -2.573  -0.030   3.010  11.779
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 96.11234    9.27663  10.361  < 2e-16 ***
## Y           -0.11493    0.10229  -1.124    0.264
## M            0.69619    0.08356   8.332 5.27e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.823 on 97 degrees of freedom
## Multiple R-squared:  0.4418, Adjusted R-squared:  0.4303
## F-statistic: 38.39 on 2 and 97 DF,  p-value: 5.233e-13
```

10.6.6   Viewing output

```
Summary Table
stargazer(fit, fita, fitb, fitc, type = "text", title = "Baron and Kenny Method")
```

```
Baron and Kenny Method
===============================================

                              ------------------------
                                         Y
                                        (1)
              ------------------------------------
Y


M


X                                    0.169**                    (
                                    (0.081)

Constant                             19.884
                                    (14.264)

              ------------------------------------
Observations                          100
R2                                   0.042
Adjusted R2                          0.033
Residual Std. Error    5.160 (df = 98)          4.85
F Statistic            4.336** (df = 1; 98) 75.313*
===============================================
Note:
```

10.6.7   Interpreting Baron and Kenny approach

A reminder of the logic of mediation:

- The direct relationship between X and Y should be significant
- The relationship between X and M should be significant
- The relationship between M and Y (controlling for X) should be significant
- When controlling for M, the strength of the relationship between X and Y decreases and is **not** significant

10.6.8   Running the Sobel test

- The Sobel test checks the singificance of indirect effects

```
#Sobel Test
library(multilevel)
sobel(Meddata$X, Meddata$M, Meddata$Y)
```

```
## $`Mod1: Y~X`
##               Estimate Std. Error  t value   Pr(>|t|)
## (Intercept) 19.8836805 14.2637142 1.394004 0.16646905
## pred          0.1689931  0.0811601 2.082220 0.03992761
##
## $`Mod2: Y~X+M`
##               Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept) 17.3217682 13.16215851  1.316028 1.912663e-01
## pred        -0.1117904  0.09949262 -1.123605 2.639537e-01
## med          0.4238113  0.09899469  4.281152 4.371472e-05
##
## $`Mod3: M~X`
##               Estimate  Std. Error  t value     Pr(>|t|)
## (Intercept) 6.0449365 13.41692114 0.4505457 6.533122e-01
## pred        0.6625203  0.07634187 8.6783345 8.871741e-14
##
## $Indirect.Effect
## [1] 0.2807836
##
## $SE
## [1] 0.07313234
##
## $z.value
## [1] 3.83939
##
## $N
## [1] 100
```

However, the above code only gives us to information we need to test the significance of the indirect effect, not the significance itself. Thereore, we can

use the following, to get the actual significance of the indirect effect:

```
library(bda)
mediation.test(Meddata$M, Meddata$X, Meddata$Y)
```

```
##                 Sobel      Aroian     Goodman
## z.value 3.8393902040 3.8190525305 3.8600562907
## p.value 0.0001233403 0.0001339652 0.0001133609
```

10.7 Mediation analysis (the Mediation package)

10.7.1 Preacher & Hayes (2004) mediation approach

- Mediation package in R uses the Preacher & Hayes (2004) bootstrapping approach
- They argue that few people test the significance of the indirect effect

    "Baron and Kenny simply state that perfect mediation has occurred if c' becomes nonsignificant after controlling for M, so researchers have focused on that requirement." (Preacher & Hayes, 2004, p. 719)

- Sobel test has low power (requires larger sample sizes)
- Sobel test assumes normality (often violated)

10.7.2 What is bootstrapping?

    "Bootstrapping is a nonparametric approach to effect-size estimation and hypothesis testing that makes no assumptions about the shape of the distributions of the variables or the sampling distribution of the statistic" (Preacer & Hayes, 2004, p. 722)

- Bootstrapping takes a large number of samples from our data and runs the analysis on each of these samples

- The sampling is done randomly with replacement, and each sample in the bootstrap is the same size as our dataset

- Using this method, we can create estimates with that fall within a narrower confidence interval (since we have now run the analysis on 100's of samples)

- Bootstrapping overcomes concerns about the distribution of our original dataset

10.7.3   Mediation example

Is the relationship between *No of hours awake* and *wakefulness* mediated by *caffiene consumption*?

> This example is from Demos & Salas (2019). *A Language, not a Letter: Learning Statistics in R* (Chapter 14)
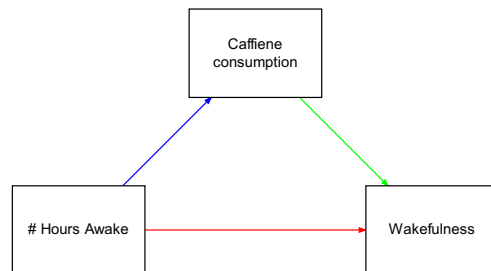
10.7.4   Step 1: Run the models

```
#Mediate package
library(mediation)


fitM <- lm(M ~ X,     data=Meddata) #IV on M; Hours since waking predicting coffee con
fitY <- lm(Y ~ X + M, data=Meddata) #IV and M on DV; Hours since dawn and coffee predi
```

10.7.5   Step 2: Check assumptions

```
gvlma(fitM)
```

```
##
## Call:
## lm(formula = M ~ X, data = Meddata)
##
## Coefficients:
## (Intercept)              X
##      6.0449         0.6625
##
```

```
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = fitM)
##
##                    Value p-value                  Decision
## Global Stat        8.833 0.06542    Assumptions acceptable.
## Skewness           6.314 0.01198 Assumptions NOT satisfied!
## Kurtosis           1.219 0.26949    Assumptions acceptable.
## Link Function      1.076 0.29959    Assumptions acceptable.
## Heteroscedasticity 0.223 0.63674    Assumptions acceptable.
```

```
# We can see that the data is positively skewed. We might need to transform the data (we will dis
```

```
gvlma(fitY)
```

```
##
## Call:
## lm(formula = Y ~ X + M, data = Meddata)
##
## Coefficients:
## (Intercept)             X             M
##     17.3218       -0.1118        0.4238
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = fitY)
##
##                     Value p-value                 Decision
## Global Stat        3.41844  0.4904 Assumptions acceptable.
## Skewness           1.85648  0.1730 Assumptions acceptable.
## Kurtosis           0.77788  0.3778 Assumptions acceptable.
## Link Function      0.71512  0.3977 Assumptions acceptable.
## Heteroscedasticity 0.06896  0.7929 Assumptions acceptable.
```

10.7.6   Step 3.1: Run the mediation analysis on the models

The mediate function gives us:

- Average Causal Mediation Effects (ACME)

- Average Direct Effects (ADE)
- combined indirect and direct effects (Total Effect)
- the ratio of these estimates (Prop. Mediated).

The ACME here is the indirect effect of M (total effect - direct effect) and thus this value tells us if our mediation effect is significant.

```
fitMed <- mediate(fitM, fitY, treat="X", mediator="M")
summary(fitMed)
```

```
##
## Causal Mediation Analysis
##
## Quasi-Bayesian Confidence Intervals
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME              0.2835       0.1517         0.43  <2e-16 ***
## ADE              -0.1167      -0.3013         0.07    0.22
## Total Effect      0.1668       0.0113         0.33    0.04 *
## Prop. Mediated    1.6739       0.5929         9.22    0.04 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 100
##
##
## Simulations: 1000
```

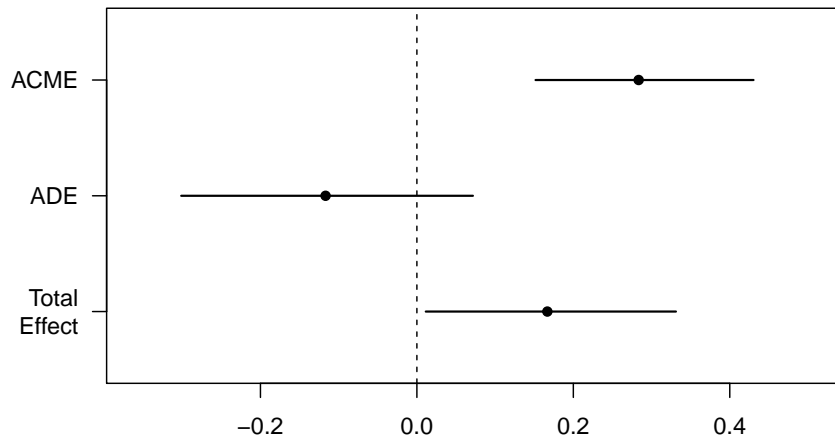### 10.7.7   Step 3.2: Plot the mediation analysis of the models

The plot below reiterates what was on the previous slide:

- The confidence intervals of Total Effect and ACME are significant
- The confidence interval of ADE is not significant

Translation:

- Total effect is signficant: there is a relationship between X and Y (direct and indirect)
- ADE is not significant: the relationship between X and Y is not direct
- ACME is significant: the relationship between X and Y is mediated by M

```
plot(fitMed)
```

```
       ACME  ─                                    ●────────
                                    ├────────────┤

        ADE  ─        ├──────────────●─────────────┤

      Total  ─                      ├─────●────────────┤
      Effect

              ─0.2      0.0      0.2      0.4
```

10.7.8   Step 4: Bootstrap the mediation model

The plot below changes our interpretation slightly:

- The confidence interval ACME is significant
- The confidence interval of Total Effect and ADE are not significant

Translation:

- Total effect is not signficant: the relationship between X and Y is not significant when we combine direct and indirect effects

- ADE is not significant: the relationship between X and Y is not direct

- ACME is significant: the relationship between X and Y is mediated by M

```
fitMedBoot <- mediate(fitM, fitY, boot=TRUE, sims=999, treat="X", mediator="M")
```

```
## Running nonparametric bootstrap
```

```
summary(fitMedBoot)
```

```
##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
```

```
## 
##                Estimate 95% CI Lower 95% CI Upper p-value
## ACME             0.2808       0.1409        0.42  <2e-16 ***
## ADE             -0.1118      -0.3080        0.12    0.31
## Total Effect     0.1690      -0.0123        0.34    0.07 .
## Prop. Mediated   1.6615      -3.7235       11.33    0.07 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Sample Size Used: 100
## 
## 
## Simulations: 999
```

```
plot(fitMedBoot) ##
```
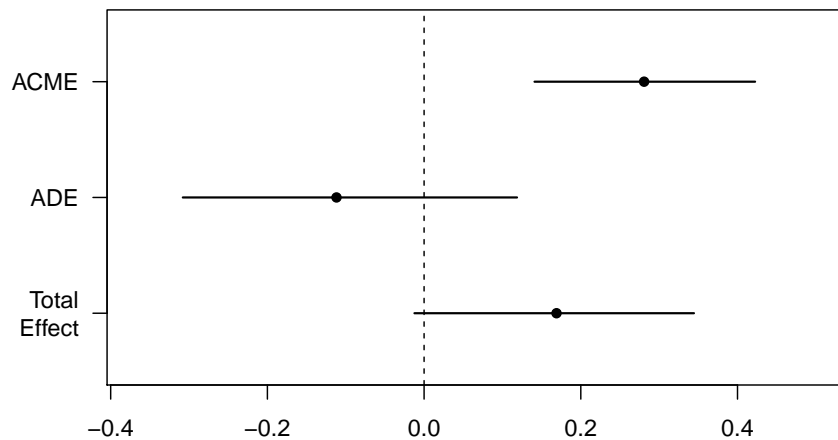

10.8   References

Demos & Salas (2019). *A Language, not a Letter: Learning Statistics in R* (Chapter 14). https://ademos.people.uic.edu/ Accessed Jan 2020.

Preacher, K. J., & Hayes, A. F. (2004). SPSS and SAS procedures for estimating indirect effects in simple mediation models. Behavior research methods, instruments, & computers, 36(4), 717-731.
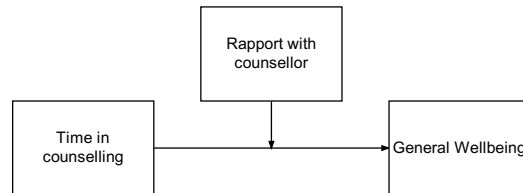
Chapter 11

Moderation analysis

Additional moderation example:

11.1 Overview

- What is moderation?
- Moderation analysis in more detail
- Grand Mean Centering
- Checking Assumptions
- Interpreting Moderation
- Bootstrapping Moderation

11.2 What is moderation?

There is a direct relationship between X and Y but it is affected by a moderator (M)

In the above model, we theorise that Time in counselling predicts General Wellbeing but the strength of the relationship is affected by the level of Rapport with counsellor

11.3   What packages do we need?

- **gvlma** (for checking assumptions)
- **interactions** (for generating interaction plot)
- **Rockchalk** (for testing simple slopes)
- **car** (includes a **Boot()** function to bootstrap regression models)

11.4   What is moderation?

- The relationship between a predictor (X) and outcome (Y) is affected by another variable (M)
- This is referred to as an interaction (similar to interaction in standard regression)
- A moderator can effect the direction and/or strength of a relationship between X and Y
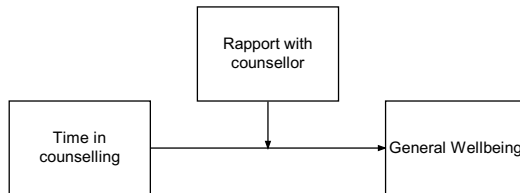
Here we might find that the relationship between Time in counselling and General Wellbeing is strong for those who have a strong rapport with their counselling psychologist and weak for those who do not have good rapport with their counselling psychologist.

- Very similar to multiple regression

  lm(Y ~ X + M + X*M)

- Moderation analysis includes X, Z and the interaction between X and Z

- If we find a moderation effect it becomes the focus of our analysis (the independent role of X and Z becomes less important)
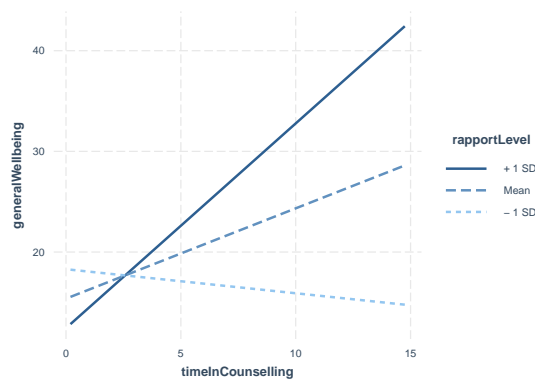
In the plot above:

- The blue line is the "standard" regression line
- The black line is when the moderator is "low" (-1sd)

- The dotted line is when the moderator is "high" (+1sd)

11.5 Moderation: step-by-step

11.5.1 Step 1: Grand Mean Centering

- Regression coefficients (b values) are based on predicting Y when X = 0
- Not all measures actually have a zero value
- To make results easier to interpret, we can centre our data around the grand mean of the data (making the mean 0)
    - The mean of the full sample is subtracted from the value
- This is similar to z-score (i.e. a standardised score)

To do this in R, we can use the **scale()** function:

```
Xc    <- scale(X, center=TRUE, scale=FALSE) #Centering X;
Mc    <- scale(M,  center=TRUE, scale=FALSE) #Centering M;
```

We then use the centred data in our analysis

We can see that the difference between the original data is the mean of the data.

```
Xc    <- scale(X, center=TRUE, scale=FALSE) #Centering X;

X
```

```
##   [1] 171.0767 173.3888 185.9110 175.4936 175.9050 187.0055 178.2264 166.1446 170.19
##  [13] 177.8054 175.7748 171.1091 187.5084 178.4850 161.2337 179.9095 171.6905 167.52
##  [25] 170.6247 163.1931 180.8645 176.0736 167.0330 183.7767 177.9852 172.9345 181.20
##  [37] 178.8774 174.5666 172.8583 172.3367 170.1371 173.5446 166.1422 190.1827 183.45
##  [49] 180.4598 174.4164 176.7732 174.8002 174.6999 184.5802 173.4196 185.6153 164.15
##  [61] 177.6575 171.4837 172.6675 167.8700 167.4975 177.1247 178.1375 175.3710 181.45
##  [73] 182.0402 170.0356 170.1839 182.1790 173.0066 166.4550 176.2691 174.0278 175.04
##  [85] 173.4566 177.3225 182.6779 178.0463 172.7185 183.0417 181.9545 178.8388 176.67
##  [97] 190.3113 185.7283 173.3501 167.8151
```

```
head(Xc)
```

```
##              [,1]
## [1,] -4.5561709
## [2,] -2.2440838
## [3,] 10.2781168
## [4,] -0.1392826
## [5,]  0.2721728
## [6,] 11.3726135
```

```
mean(X)
```

```
## [1] 175.6328
```

```
  X[1]-Xc[1]
```

```
## [1] 175.6328
```

```
#Centering Data
Moddata$timeInCounselling_centred    <- c(scale(timeInCounselling, center=TRUE, scale=FALSE))

#Centering IV;
Moddata$rapportLevel_centred    <- c(scale(rapportLevel,  center=TRUE, scale=FALSE)) #Centering m

#Moderation "By Hand" with centred data
library(gvlma)
fitMod <- lm(generalWellbeing ~ timeInCounselling_centred *rapportLevel_centred  , data = Moddata

library(interactions)
 ip <- interact_plot(fitMod, pred = timeInCounselling_centred, modx = rapportLevel_centred)
 ip
```
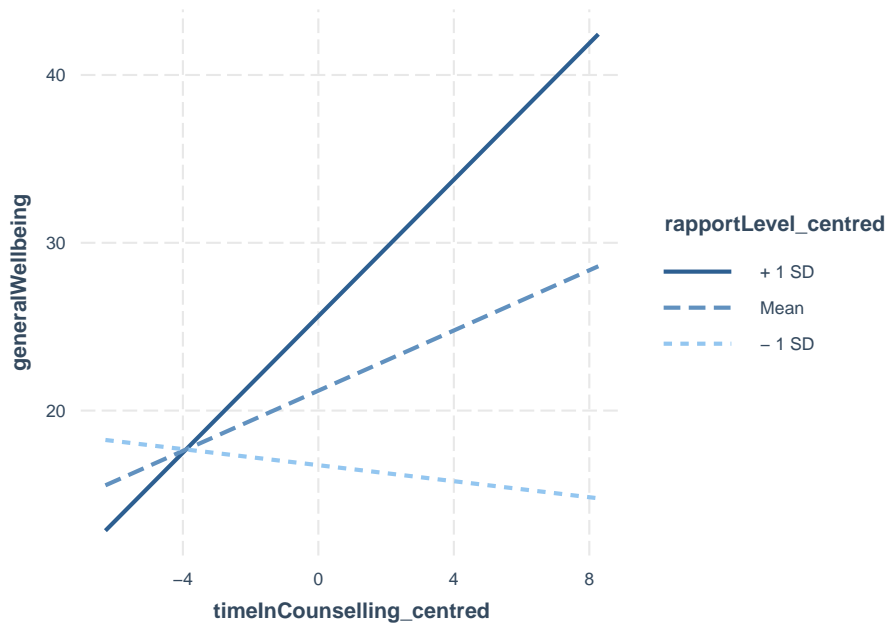

11.5.1.1   Do I need to mean centre my data?

It is worth noting:

- It does not change the results of your interaction (coefficient, standard error or significance tests).
- It will change the results of the direct effects (the individual predictors in your model).

- It is a step that tries to ensure that the coefficients of the predictor and moderator are meaningful in relation to each other.
- In some cases, it might not be necessary to mean centre at all. However, there is no harm in doing so, and it could potentially be helpful.

Hayes (2013) discusses mean centering, pp. 282-290.

McClelland, G. H., Irwin, J. R., Disatnik, D., & Sivan, L. (2017). Multicollinearity is a red herring in the search for moderator variables: A guide to interpreting moderated multiple regression models and a critique of Iacobucci, Schneider, Popovich, and Bakamitsos (2016). Behavior research methods, 49(1), 394-402.

11.5.2   Step 2: Check assumptions

We can use the gvlma function to check regression assumptions

```
library(gvlma)
gvlma(fitMod)
```

```
## 
## Call:
## lm(formula = generalWellbeing ~ timeInCounselling_centred * rapportLevel_centred,
##     data = Moddata)
## 
## Coefficients:
##                                (Intercept)                         timeInCounselli
##                                    21.1851
##                       rapportLevel_centred  timeInCounselling_centred:rapportLev
##                                     0.5842
## 
## 
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
## 
## Call:
##  gvlma(x = fitMod)
## 
##                    Value p-value                   Decision
## Global Stat        9.6949 0.04589 Assumptions NOT satisfied!
## Skewness           7.7571 0.00535 Assumptions NOT satisfied!
## Kurtosis           1.2182 0.26972    Assumptions acceptable.
## Link Function      0.5287 0.46716    Assumptions acceptable.
## Heteroscedasticity 0.1910 0.66207    Assumptions acceptable.
```

The "global stat" is an attempt to check multiple assumptions of linear model: Pena, E. A., & Slate, E. H. (2006). Global validation of linear model assumptions. Journal of the American Statistical Association, 101(473), 341-354.

Since one of the underlying assumptions is violated, the overall stat is also not acceptable.

The data looks skewed, we should transform it or perhaps use bootstrapping

11.5.3   Step 3: Moderation Analysis

```
fitMod <- lm(generalWellbeing ~ timeInCounselling_centred *rapportLevel_centred  , data = Moddata
 #Model interacts IV & moderator
summary(fitMod)
```
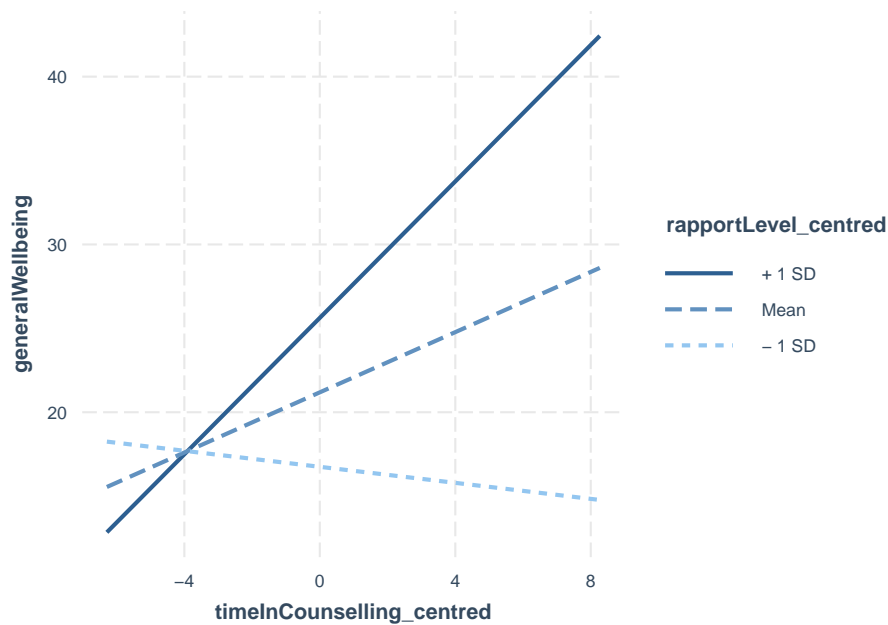
```
##
## Call:
## lm(formula = generalWellbeing ~ timeInCounselling_centred * rapportLevel_centred,
##     data = Moddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.121  -8.938  -0.670   5.840  37.396
##
## Coefficients:
##                                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                  21.18508    1.14115  18.565  < 2e-16 ***
## timeInCounselling_centred                     0.89707    0.33927   2.644 0.009569 **
## rapportLevel_centred                          0.58416    0.15117   3.864 0.000203 ***
## timeInCounselling_centred:rapportLevel_centred 0.14948    0.04022   3.716 0.000340 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.33 on 96 degrees of freedom
## Multiple R-squared:  0.2737, Adjusted R-squared:  0.251
## F-statistic: 12.06 on 3 and 96 DF,  p-value: 9.12e-07
```

The results above show that there is a moderated effect

11.5.3.1   Visualising the moderation effect

We use an approach called **simple slopes** to visualise the moderation effect

```
interact_plot(fitMod, pred = timeInCounselling_centred, modx = rapportLevel_centred)
```

The **rockchalk** package includes useful functions for visualising simple slopes

```
library(rockchalk)

fitMod <- lm(generalWellbeing ~ timeInCounselling *rapportLevel  , data = Moddata)
summary(fitMod)
```

```
##
## Call:
## lm(formula = generalWellbeing ~ timeInCounselling * rapportLevel,
##     data = Moddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.121  -8.938  -0.670   5.840  37.396
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   17.28006    3.17944   5.435 4.15e-07 ***
## timeInCounselling              0.15510    0.42033   0.369  0.71296
## rapportLevel                  -0.38484    0.29916  -1.286  0.20140
## timeInCounselling:rapportLevel  0.14948    0.04022   3.716  0.00034 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.33 on 96 degrees of freedom
```

```
## Multiple R-squared:  0.2737, Adjusted R-squared:  0.251
## F-statistic: 12.06 on 3 and 96 DF,  p-value: 9.12e-07
```

```
slopes <- plotSlopes(fitMod, modx = "rapportLevel", plotx = "timeInCounselling")
```
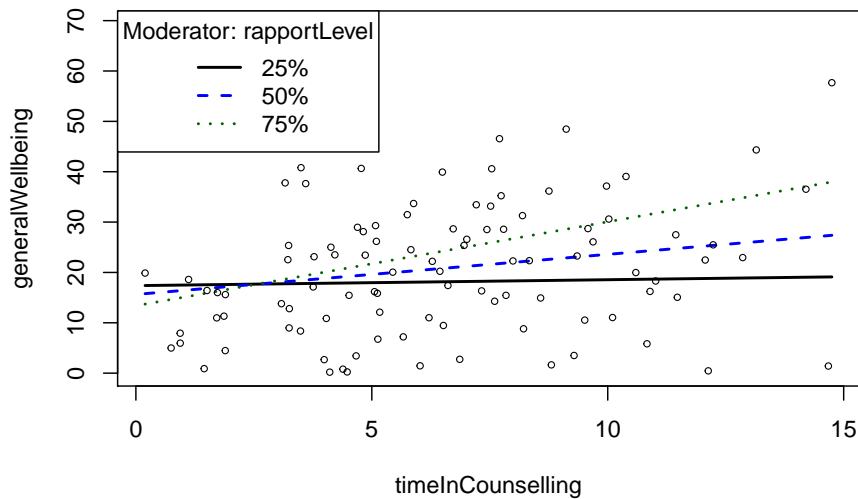


```
testSlopes <- testSlopes(slopes)
```

```
## Values of rapportLevel OUTSIDE this interval:
##          lo          hi
## -11.580166    3.634439
## cause the slope of (b1 + b2*rapportLevel)timeInCounselling to be statistically significant
```

```
plot(testSlopes)
```
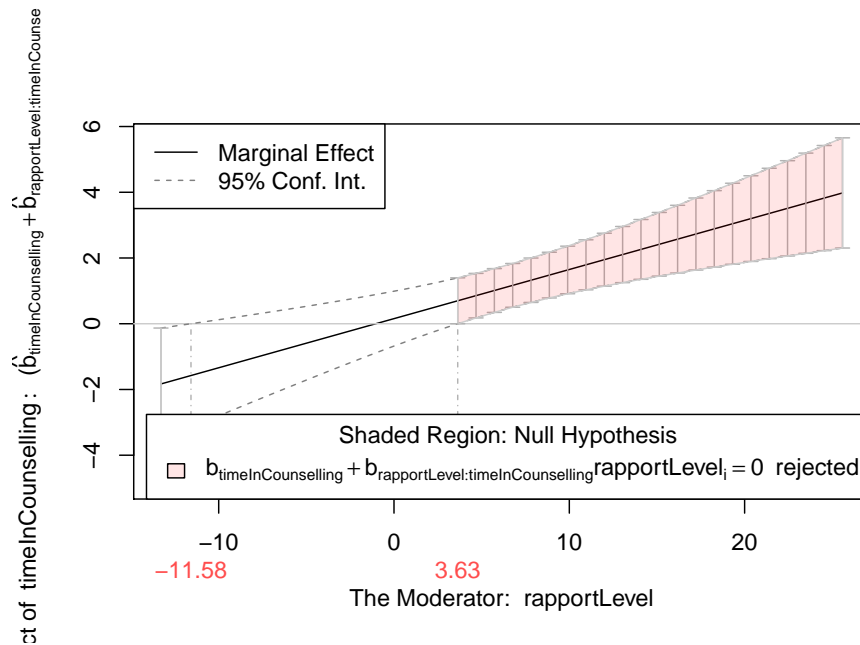
11.5.4 Step 4: Bootstrapping

The **car** package includes a function to bootstrap regression

```
library(car)

bootstrapModel <- Boot(fitMod, R=999)

confint(fitMod)
```

```
##                                    2.5 %      97.5 %
## (Intercept)                    10.96891826 23.5912086
## timeInCounselling              -0.67926290  0.9894532
## rapportLevel                   -0.97866229  0.2089882
## timeInCounselling:rapportLevel  0.06963667  0.2293205
```

```
confint(bootstrapModel)
```

```
## Bootstrap bca confidence intervals
##
##                                    2.5 %      97.5 %
## (Intercept)                    11.57230420 23.7222700
## timeInCounselling              -0.61780918  1.0397199
## rapportLevel                   -0.90786799  0.2558502
## timeInCounselling:rapportLevel  0.05806412  0.2146814
```

```
summary(bootstrapModel)
```

```
##
## Number of bootstrap replications R = 999
##                                  original     bootBias    bootSE     bootMed
## (Intercept)                      17.28006  -0.13667103  3.165301  17.05431
## timeInCounselling                 0.15510   0.01637117  0.399550   0.15929
## rapportLevel                     -0.38484   0.00716631  0.294061  -0.38218
## timeInCounselling:rapportLevel    0.14948  -0.00052838  0.038516   0.14974
```

```
hist(bootstrapModel)
```
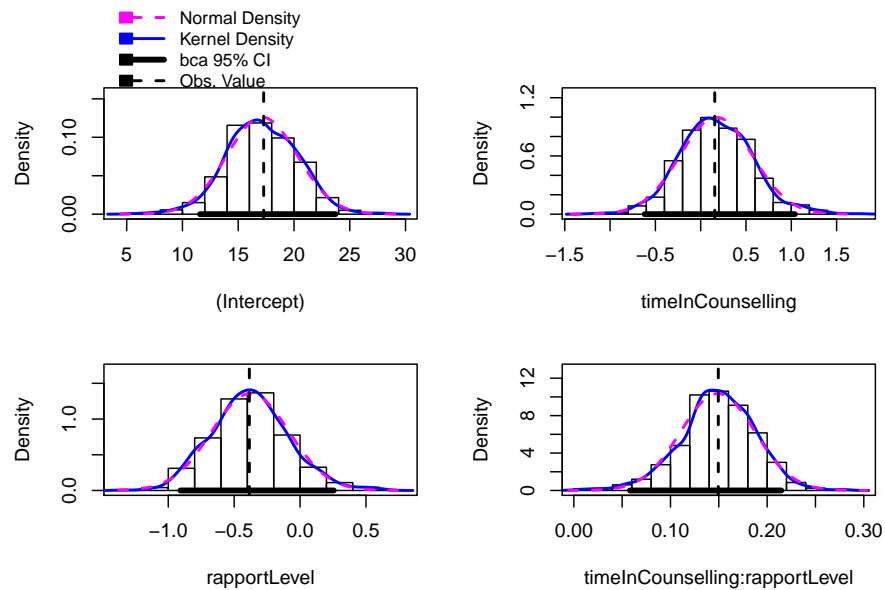
Chapter 12

Factor Analysis

```
##
## -- Column specification -----------------------------------------------------------------
## cols(
##   .default = col_double()
## )
## i Use `spec()` for the full column specifications.
```

12.1 Overview

- What is factor analysis
- CFA versus PCA
- Variance in factor analysis
- Considertations for factor analysis
- Identifying / extracting factors
- Rotation
- Cronbach's alpha

12.2 Exploratory Factor analysis

- Identify the relational structure between a set of variables in order to reduce them to a smaller set of factors
  - The process of **dimension reduction** (identify new variables) or **data summarisation** (summarise what is already there)

12.2.1 Dimension reducton

- **Latent Variables**: Not directly observable. Rather they are inferred from other responses

– Many psychological constructs (e.g. anxiety) are latent variables that
  we cannot directly measure.
– Rather, we can measure behaviours, cognitions and other variables
  that are related to the construct.

We might concptualise this as: "Responses to the questions are
indicative of levels of underlying anxiety"
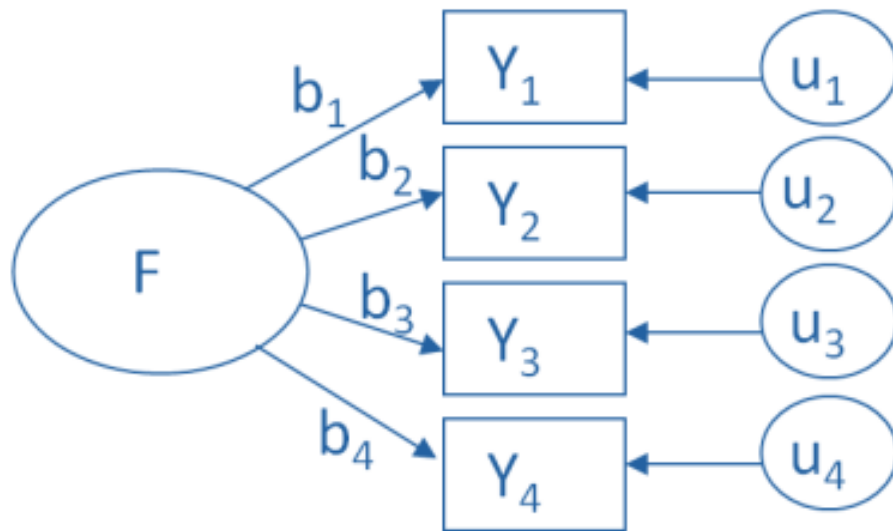
12.2.2   Data summarisation

- **Index Variables** or **Components**: A weighted summary of measured
  variables that contribute to the component variable

- "Principal components are variables of maximal variance constructed from
  linear combinations of the input features"

  We might conceptualise this as: "We can reduce these mea-
  sures/questions to a smaller set of higher order, independent,
  composite variables"
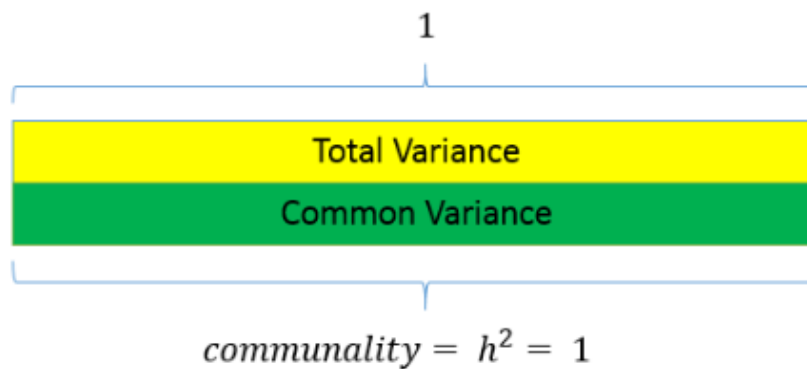
12.3   Variance in exploratory factor analysis

There are two common methods of exploratory factor analysis: **Common
Factor analysis** and **Principal Component Analysis**

- CFA assumes that there are two types of variance: common and unique
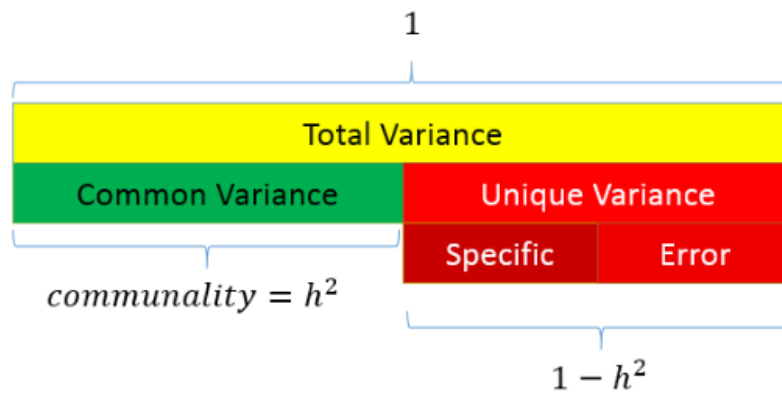
12.3.1 Variance in PCA

- PCA only assumes common variance

12.3.2 Variance in CFA

- Due to these different approaches, PCA is considered to be reflective of the current sample but not generalisable to the wider population
- Whereas, CFA is considered appropriate for hypothesis testing and making inferences to the population

12.4 What is factor analysis?

- If we measure several variables (or questions), we can examine the correlation between sets of these variables
  - Such a correlation matrix is known as an **R Matrix** (r because correlation)

- If there are clusters of correlations between a number of the variables (or questions), this indicates that they might be linked to the same underlying dimension (or latent variable)
- The researcher should use informed judgement when assessing the appropriateness of variables for inclusion

Correlations

	1	2	3	4
1	1			
2	-.099**	1		
3	-.337**	.318**	1	
4	.436**	-.112**	-.380**	1
5	.402**	-.119**	-.310**	.401**
6	.217**	-.074**	-.227**	.278**
7	.305**	-.159**	**-.382****	.409**
8	.331**	-.050*	-.259**	.349**

**. Correlation is significant at the

*. Correlation is significant at the

An r matrix example

12.5   Considerations with factor analysis

- Sample size:
  - Must be more data points than variables being measured
  - A common rule of thumb is at least 10 per variable
  - There are tests to assess sample size adequacy (e.g. Kaiser-Meyer test should be greater than 0.5)
- Inter-correlation:
  - There must be sufficient correlation between the variables being measured
  - A high number of correlations over 0.3
  - Can be tested using Bartlett test of sphericity (sig. result means factor analysis can be used)

Other things to check (see Field, 2018)

- The quality of analysis depends upon the quality of the data (GI/GO).
- Avoid multicollinearity:
  - several variables highly correlated, r > .80.
  - Determinent: should be greater than 0.00001
- Avoid singularity:
  - some variables perfectly correlated, r = 1.
- Screen the correlation matrix, eliminate any variables that obviously cause concern.

12.6   Representing factor analysis

We can represent factors visually based on the strength of their inter-correlations - Here, the axis of the graph represents a factor or latent variable
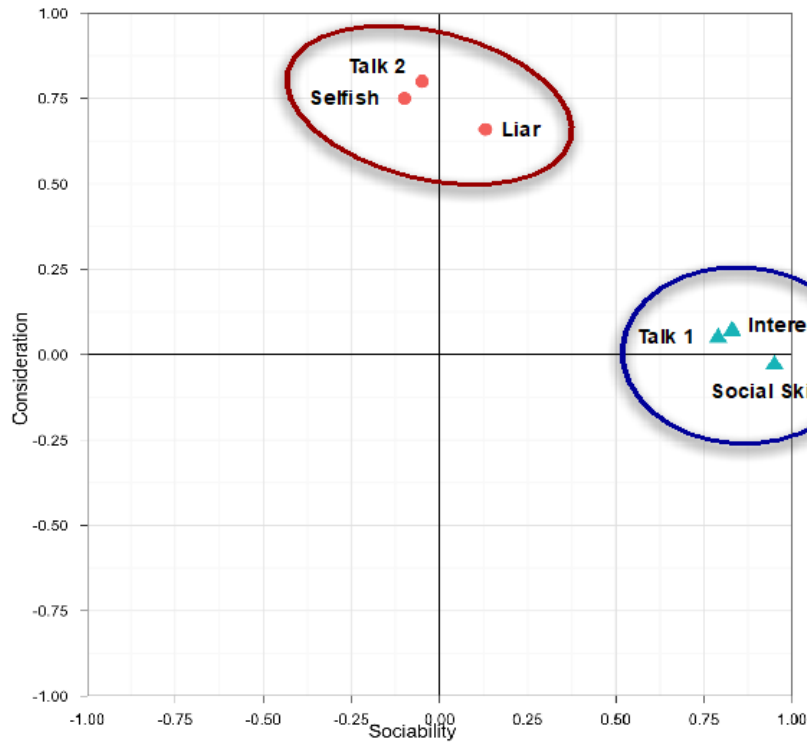
We can also represent factor analysis using a regression equation - Here the beta values represent the extent to which the variable "loads onto" a particular factor

$$Y = b_1 X_1 + b_2 X_2 + \ldots + b_n X_n$$

$$\text{Factor}_i = b_1 \text{Variable}_1 + b_2 \text{Variable}_2 + \ldots + b_n \text{Variable}_n$$

$$Y = b_1 X_1 + b_2 X_2 + \ldots + b_n X_n$$

$$\text{Sociability} = b_1 \text{Talk1} + b_2 \text{Social Skills} + b_3 \text{Interest}$$
$$+ b_4 \text{Talk2} + b_5 \text{Selfish} + b_6 \text{Liar}$$

$$\text{Consideration} = b_1 \text{Talk1} + b_2 \text{Social Skills} + b_3 \text{Interest}$$
$$+ b_4 \text{Talk2} + b_5 \text{Selfish} + b_6 \text{Liar}$$

Example: Statistics anxiety

- Many people get anxious about statistics

- We can ask them about their experience in a number of ways (e.g. questions compiled by students in a stats class)

- Their responses might indicate that stats anxiety has a number of dimensions

    - i.e. it is a multi-dimensional construct, as opposed to a unitary construct

SD = Strongly Disagree, D = Disagree, N = Neither, A = Agree, SA = Strongly Agree

		SD	D	N	A	SA
1	Statistics make me cry	O	O	O	O	O
2	My friends will think I'm stupid for not being able to cope with R	O	O	O	O	O
3	Standard deviations excite me	O	O	O	O	O
4	I dream that Pearson is attacking me with correlation coefficients	O	O	O	O	O
5	I don't understand statistics	O	O	O	O	O
6	I have little experience of computers	O	O	O	O	O
7	All computers hate me	O	O	O	O	O
8	I have never been good at mathematics	O	O	O	O	O
9	My friends are better at statistics than me	O	O	O	O	O
10	Computers are useful only for playing games	O	O	O	O	O
11	I did badly at mathematics at school	O	O	O	O	O
12	People try to tell you that R makes statistics easier to understand but it doesn't	O	O	O	O	O
13	I worry that I will cause irreparable damage because of my incompetence with computers	O	O	O	O	O
14	Computers have minds of their own and deliberately go wrong whenever I use them	O	O	O	O	O
15	Computers are out to get me	O	O	O	O	O
16	I weep openly at the mention of central tendency	O	O	O	O	O
17	I slip into a coma whenever I see an equation	O	O	O	O	O
18	R always crashes when I try to use it	O	O	O	O	O
19	Everybody looks at me when I use R	O	O	O	O	O
20	I can't sleep for thoughts of eigenvectors	O	O	O	O	O
21	I wake up under my duvet thinking that I am trapped under a normal distribution	O	O	O	O	O
22	My friends are better at R than I am	O	O	O	O	O

12.7   Step 1: Create a correlation matrix

```
raq.matrix <- cor(raq)

raq.matrix
```

```
##                  Q01          Q02          Q03          Q04          Q05          Q06
## Q01   1.000000000 -0.09872403  -0.3366489   0.43586018   0.40243992   0.21673399   0.3053(
## Q02  -0.098724032  1.00000000   0.3183902  -0.11185965  -0.11934658  -0.07420968  -0.1591
## Q03  -0.336648879  0.31839020   1.0000000  -0.38046016  -0.31030879  -0.22674048  -0.3819
## Q04   0.435860179 -0.11185965  -0.3804602   1.00000000   0.40067225   0.27820154   0.4086
## Q05   0.402439917 -0.11934658  -0.3103088   0.40067225   1.00000000   0.25746014   0.3393
## Q06   0.216733985 -0.07420968  -0.2267405   0.27820154   0.25746014   1.00000000   0.51358
## Q07   0.305365139 -0.15917448  -0.3819533   0.40861502   0.33939179   0.51358048   1.00000
## Q08   0.330737608 -0.04962257  -0.2586342   0.34942939   0.26862697   0.22283175   0.29749
## Q09  -0.092339458  0.31464054   0.2998036  -0.12454637  -0.09570151  -0.11264384  -0.12829
## Q10   0.213681706 -0.08400316  -0.1933887   0.21581010   0.25820925   0.32223023   0.2837:
## Q11   0.356786290 -0.14382984  -0.3506397   0.36865655   0.29782882   0.32807072   0.34474
## Q12   0.345381133 -0.19486946  -0.4099513   0.44164706   0.34674325   0.31250937   0.42298
## Q13   0.354646283 -0.14274026  -0.3179193   0.34429168   0.30182159   0.46640487   0.44211
## Q14   0.337879655 -0.16469991  -0.3707551   0.35080964   0.31533810   0.40224407   0.44070
## Q15   0.245752635 -0.16499581  -0.3123968   0.33423089   0.26137190   0.35989309   0.39136
## Q16   0.498618057 -0.16755228  -0.4186478   0.41586725   0.39491795   0.24433888   0.38854
## Q17   0.370550512 -0.08699527  -0.3273715   0.38273945   0.31041722   0.28226121   0.39074
## Q18   0.347118037 -0.16389415  -0.3752329   0.38200149   0.32209148   0.51332164   0.50086
## Q19  -0.189011027  0.20329748   0.3415737  -0.18597751  -0.16532210  -0.16675017  -0.26912
## Q20   0.213897945 -0.20159437  -0.3248338   0.24291796   0.19966945   0.10092489   0.22095
## Q21   0.329153138 -0.20461730  -0.4171878   0.41029317   0.33461494   0.27233273   0.48300
## Q22  -0.104408664  0.23087487   0.2036569  -0.09838349  -0.13253593  -0.16513541  -0.16820
## Q23  -0.004480593  0.09967828   0.1502065  -0.03381815  -0.04165684  -0.06868743  -0.07029
##                  Q10          Q11          Q12          Q13          Q14          Q15
## Q01   0.21368171   0.35678629   0.34538113   0.35464628   0.33787966   0.24575263   0.4986:
## Q02  -0.08400316  -0.14382984  -0.19486946  -0.14274026  -0.16469991  -0.16499581  -0.16755
## Q03  -0.19338871  -0.35063969  -0.40995127  -0.31791928  -0.37075510  -0.31239678  -0.41864
## Q04   0.21581010   0.36865655   0.44164706   0.34429168   0.35080964   0.33423089   0.41586
## Q05   0.25820925   0.29782882   0.34674325   0.30182159   0.31533810   0.26137190   0.39491
## Q06   0.32223023   0.32807072   0.31250937   0.46640487   0.40224407   0.35989309   0.24433
## Q07   0.28372299   0.34474770   0.42298591   0.44211926   0.44070276   0.39136675   0.38854
## Q08   0.15860850   0.62929768   0.25198582   0.31424716   0.28058958   0.29968600   0.32149
## Q09  -0.13418658  -0.11552479  -0.16739436  -0.16743882  -0.12150197  -0.18657099  -0.18886
## Q10   1.00000000   0.27143657   0.24582591   0.30196707   0.25468730   0.29523438   0.29058
## Q11   0.27143657   1.00000000   0.33529466   0.42316548   0.32532025   0.36482687   0.36907
## Q12   0.24582591   0.33529466   1.00000000   0.48871303   0.43270398   0.33179910   0.40805
## Q13   0.30196707   0.42316548   0.48871303   1.00000000   0.44978632   0.34219704   0.3583:
## Q14   0.25468730   0.32532025   0.43270398   0.44978632   1.00000000   0.38011484   0.41841
```

```
## Q15   0.29523438   0.36482687   0.33179910   0.34219704   0.38011484   1.00000000   0.45427861   0.373
## Q16   0.29058576   0.36907763   0.40805908   0.35837775   0.41841820   0.45427861   1.00000000   0.409
## Q17   0.21832214   0.58683495   0.33269383   0.40837657   0.35374183   0.37310235   0.40976309   1.000
## Q18   0.29250304   0.37341373   0.49296482   0.53293713   0.49830615   0.34287045   0.42197911   0.375
## Q19  -0.12723487  -0.19965203  -0.26665953  -0.22697105  -0.25405813  -0.20980230  -0.26704702  -0.162
## Q20   0.08406520   0.25533736   0.29802585   0.20396327   0.22592173   0.20625622   0.26514025   0.205
## Q21   0.19313633   0.34643407   0.44063832   0.37443078   0.39938896   0.29971557   0.42054273   0.363
## Q22  -0.13090831  -0.16198921  -0.16728557  -0.19535632  -0.16983754  -0.16790617  -0.15579385  -0.126
## Q23  -0.06191796  -0.08637256  -0.04642506  -0.05298304  -0.04847418  -0.06200665  -0.08152195  -0.091
##              Q19          Q20          Q21          Q22          Q23
## Q01 -0.1890110   0.21389794   0.32915314  -0.10440866  -0.004480593
## Q02  0.2032975  -0.20159437  -0.20461730   0.23087487   0.099678285
## Q03  0.3415737  -0.32483385  -0.41718781   0.20365686   0.150206522
## Q04 -0.1859775   0.24291796   0.41029317  -0.09838349  -0.033818152
## Q05 -0.1653221   0.19966945   0.33461494  -0.13253593  -0.041656841
## Q06 -0.1667502   0.10092489   0.27233273  -0.16513541  -0.068687430
## Q07 -0.2691203   0.22095420   0.48300388  -0.16820488  -0.070290157
## Q08 -0.1594767   0.17515089   0.29571756  -0.07917265  -0.050238392
## Q09  0.2493117  -0.15864747  -0.13594310   0.25684622   0.170774410
## Q10 -0.1272349   0.08406520   0.19313633  -0.13090831  -0.061917956
## Q11 -0.1996520   0.25533736   0.34643407  -0.16198921  -0.086372565
## Q12 -0.2666595   0.29802585   0.44063832  -0.16728557  -0.046425059
## Q13 -0.2269710   0.20396327   0.37443078  -0.19535632  -0.052983042
## Q14 -0.2540581   0.22592173   0.39938896  -0.16983754  -0.048474181
## Q15 -0.2098023   0.20625622   0.29971557  -0.16790617  -0.062006650
## Q16 -0.2670470   0.26514025   0.42054273  -0.15579385  -0.081521950
## Q17 -0.1628810   0.20523013   0.36349147  -0.12629066  -0.091672426
## Q18 -0.2566318   0.23518040   0.43010427  -0.15982631  -0.080416984
## Q19  1.0000000  -0.24859386  -0.27489793   0.23392259   0.122434401
## Q20 -0.2485939   1.00000000   0.46770448  -0.09970186  -0.034665293
## Q21 -0.2748979   0.46770448   1.00000000  -0.12902148  -0.067664367
## Q22  0.2339226  -0.09970186  -0.12902148   1.00000000   0.230369402
## Q23  0.1224344  -0.03466529  -0.06766437   0.23036940   1.000000000
```

```
library(corrplot)
corrplot(raq.matrix, method = "number")
```
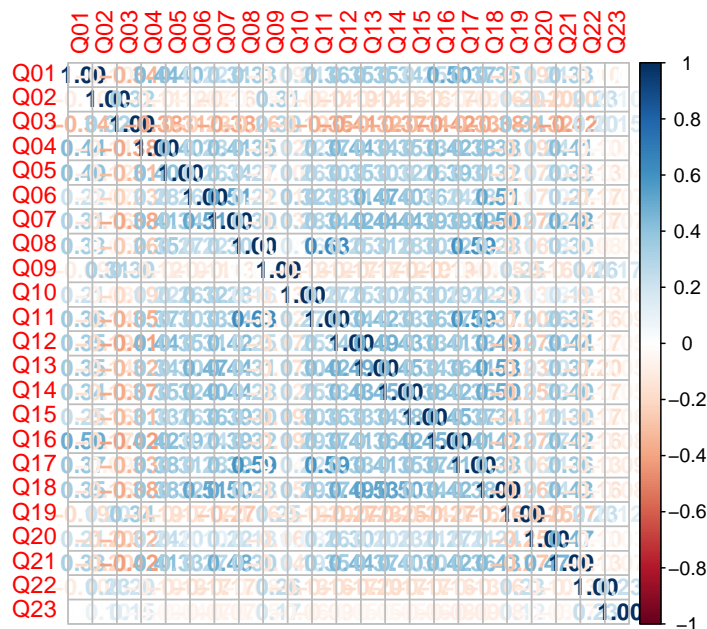
- We can use bartlett's test from the psych package

```r
library(psych)

cortest.bartlett(raq.matrix, n=2571)
```

```
## $chisq
## [1] 19334.49
##
## $p.value
## [1] 0
##
## $df
## [1] 253
```

12.9   Step 3: Check sampling adequacy

- Overall should be > 0.5

```r
KMO(raq)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = raq)
## Overall MSA =  0.93
## MSA for each item =
##  Q01  Q02  Q03  Q04  Q05  Q06  Q07  Q08  Q09  Q10  Q11  Q12  Q13  Q14  Q15  Q16  Q1
```

0.93 0.87 0.95 0.96 0.96 0.89 0.94 0.87 0.83 0.95 0.91 0.95 0.95 0.97 0.94 0.93 0.93 0.95 0.94

12.10 Step 4: Identify number of factors

- Based on Eigenvalues:
  - Kaiser (1960) – retain factors with eigen values > 1.
  - Joliffe (1972) – retain factors with eigen values > .70.
- Use a scree plot: Cattell (1966): use 'point of inflexion.'

12.10.1 Which rule?

- Use Kaiser's extraction when
  - Less than 30 variables, communalities after extraction > 0.7
  - Sample size > 250 and mean communality > 0.6
- Scree plot is good if sample size is > 200

12.10.2 Scree plot

```
scree(raq)
```


Scree plot
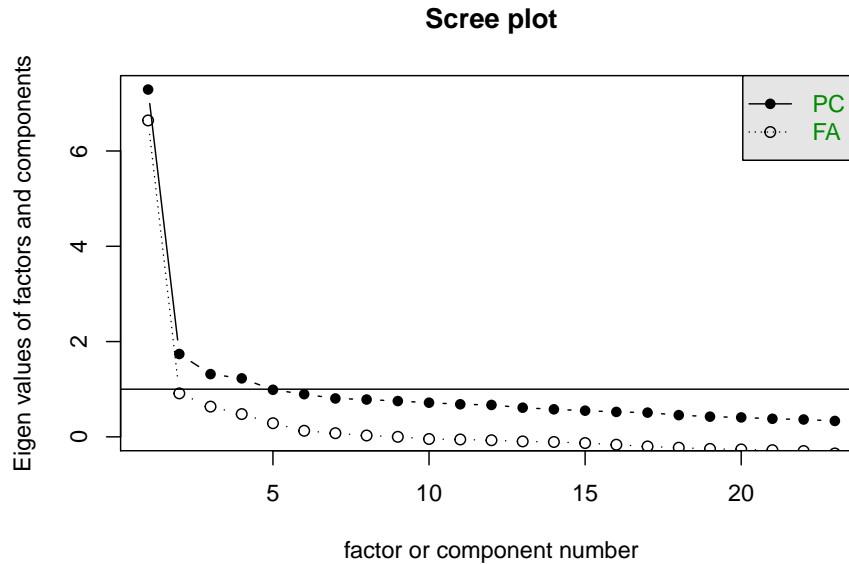
- We are looking for the point of inflection
- Where there is a drop-off

One approach: See how many factors we can draw a line through

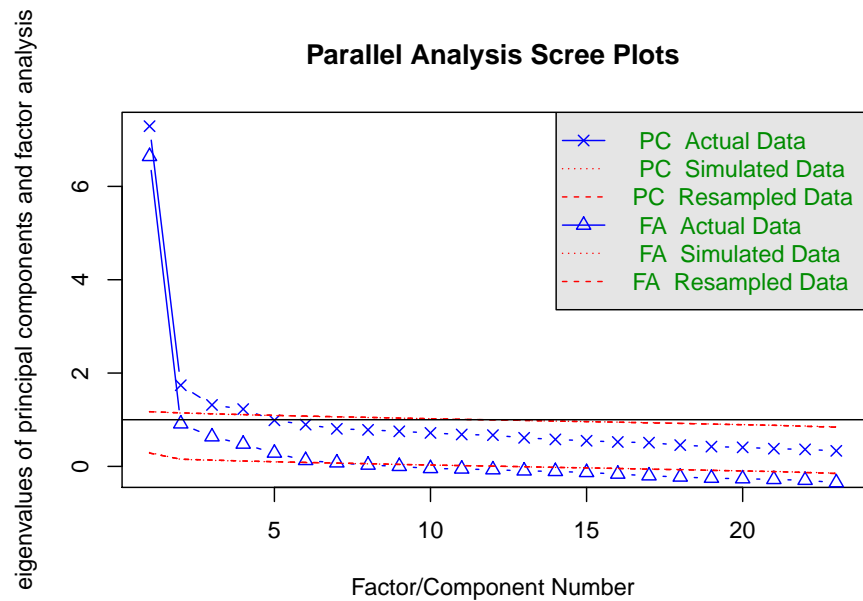### 12.10.3   Parallel analysis

How many dimensions of stats anxiety are captured in the question-
naire?

- We can run a **parallel analysis** to get an indication of the number of
  factors contained within the data
- Parallel Analysis:
  - Simulates data within the same range of values as our data set
  - Suggests that we retain, at maximum, the factors with eigenvalues
    larger than those extracted from simulated data.

SD = Strongly Disagree, D = Disagree, N = Neither, A = Agree, SA = Strongly Agree

		SD	D	N	A	SA
1	Statistics make me cry	○	○	○	○	○
2	My friends will think I'm stupid for not being able to cope with R	○	○	○	○	○
3	Standard deviations excite me	○	○	○	○	○
4	I dream that Pearson is attacking me with correlation coefficients	○	○	○	○	○
5	I don't understand statistics	○	○	○	○	○
6	I have little experience of computers	○	○	○	○	○
7	All computers hate me	○	○	○	○	○
8	I have never been good at mathematics	○	○	○	○	○
9	My friends are better at statistics than me	○	○	○	○	○
10	Computers are useful only for playing games	○	○	○	○	○
11	I did badly at mathematics at school	○	○	○	○	○
12	People try to tell you that R makes statistics easier to understand but it doesn't	○	○	○	○	○
13	I worry that I will cause irreparable damage because of my incompetence with computers	○	○	○	○	○
14	Computers have minds of their own and deliberately go wrong whenever I use them	○	○	○	○	○
15	Computers are out to get me	○	○	○	○	○
16	I weep openly at the mention of central tendency	○	○	○	○	○
17	I slip into a coma whenever I see an equation	○	○	○	○	○
18	R always crashes when I try to use it	○	○	○	○	○
19	Everybody looks at me when I use R	○	○	○	○	○
20	I can't sleep for thoughts of eigenvectors	○	○	○	○	○
21	I wake up under my duvet thinking that I am trapped under a normal distribution	○	○	○	○	○
22	My friends are better at R than I am	○	○	○	○	○

```r
library(psych)

parallel_analysis <- fa.parallel(raq)
```


Parallel Analysis Scree Plots

```
## Parallel analysis suggests that the number of factors =  6  and the number of compor
parallel_analysis
```

```
## Call: fa.parallel(x = raq)
## Parallel analysis suggests that the number of factors =  6  and the number of compor
##
##  Eigen Values of
##   Original factors Resampled data Simulated data Original components Resampled compo
## 1             6.64           0.29           0.30               7.29
## 2             0.91           0.15           0.15               1.74
## 3             0.63           0.14           0.13               1.32
## 4             0.48           0.12           0.11               1.23
## 5             0.29           0.10           0.10               0.99
## 6             0.13           0.09           0.08               0.90
```

12.11   Step 5: Perform factor analysis (with initial recommended # factors)

```
paf <- fa(raq,
nfactors = 6,
fm="pa",
max.iter = 100,
rotate = "none")
```

```
paf
```

```
## Factor Analysis using method =  pa
## Call: fa(r = raq, nfactors = 6, rotate = "none", max.iter = 100, fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##       PA1   PA2   PA3   PA4   PA5   PA6   h2   u2 com
## Q01  0.57  0.13 -0.12  0.23 -0.28 -0.19 0.52 0.48 2.3
## Q02 -0.28  0.37  0.17  0.12 -0.03  0.01 0.26 0.74 2.6
## Q03 -0.60  0.25  0.20 -0.02 -0.01  0.03 0.46 0.54 1.6
## Q04  0.61  0.08 -0.06  0.18 -0.09 -0.03 0.42 0.58 1.3
## Q05  0.52  0.04 -0.02  0.15 -0.17 -0.08 0.33 0.67 1.5
## Q06  0.55  0.02  0.49 -0.17  0.07 -0.01 0.57 0.43 2.2
## Q07  0.66 -0.03  0.22  0.03  0.11  0.06 0.50 0.50 1.3
## Q08  0.55  0.49 -0.27 -0.21  0.10 -0.02 0.66 0.34 2.9
## Q09 -0.27  0.46  0.12  0.21  0.10  0.03 0.35 0.65 2.4
## Q10  0.40 -0.01  0.17 -0.09 -0.15  0.02 0.22 0.78 1.8
## Q11  0.64  0.31 -0.20 -0.27  0.08 -0.04 0.63 0.37 2.1
## Q12  0.64 -0.10  0.06  0.15  0.05 -0.07 0.45 0.55 1.2
## Q13  0.65  0.02  0.22 -0.06  0.06 -0.13 0.50 0.50 1.4
## Q14  0.63 -0.04  0.16  0.06  0.01  0.01 0.42 0.58 1.2
## Q15  0.58 -0.01  0.07 -0.15 -0.19  0.44 0.59 0.41 2.3
## Q16  0.66 -0.02 -0.11  0.14 -0.28  0.09 0.56 0.44 1.6
## Q17  0.63  0.36 -0.15 -0.15  0.04  0.01 0.57 0.43 1.9
## Q18  0.68 -0.04  0.28  0.04  0.09 -0.10 0.57 0.43 1.4
## Q19 -0.40  0.27  0.11  0.06 -0.05  0.02 0.25 0.75 2.0
## Q20  0.41 -0.17 -0.25  0.19  0.24  0.11 0.37 0.63 3.5
## Q21  0.64 -0.10 -0.11  0.27  0.28  0.10 0.60 0.40 2.0
## Q22 -0.28  0.29  0.05  0.28  0.05  0.11 0.26 0.74 3.4
## Q23 -0.13  0.18  0.08  0.23  0.01  0.08 0.12 0.88 3.1
##
##                         PA1  PA2  PA3  PA4  PA5  PA6
## SS loadings            6.79 1.14 0.83 0.67 0.45 0.32
## Proportion Var         0.30 0.05 0.04 0.03 0.02 0.01
## Cumulative Var         0.30 0.34 0.38 0.41 0.43 0.44
## Proportion Explained   0.67 0.11 0.08 0.07 0.04 0.03
## Cumulative Proportion  0.67 0.78 0.86 0.92 0.97 1.00
```

```
##
## Mean item complexity =  2
## Test of the hypothesis that 6 factors are sufficient.
##
## The degrees of freedom for the null model are  253  and the objective function was
## The degrees of freedom for the model are 130  and the objective function was  0.23
##
## The root mean square of the residuals (RMSR) is  0.02
## The df corrected root mean square of the residuals is  0.02
##
## The harmonic number of observations is  2571 with the empirical chi square  364.66
## The total number of observations was  2571  with Likelihood Chi Square =  578.65  wi
##
## Tucker Lewis Index of factoring reliability =  0.954
## RMSEA index =  0.037  and the 90 % confidence intervals are  0.034 0.04
## BIC =  -442.12
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                    PA1  PA2  PA3  PA4   PA5   PA6
## Correlation of (regression) scores with factors   0.97 0.83 0.80 0.75  0.70  0.65
## Multiple R square of scores with factors          0.93 0.68 0.64 0.56  0.48  0.42
## Minimum correlation of possible factor scores     0.87 0.37 0.27 0.12 -0.03 -0.17
```

12.11.1   Check the factor matrix

- We are looking high levels of variance explained with SS loadings > 1

```
print(paf$loadings, cutoff=0, digits=3)
```

```
##
## Loadings:
##       PA1    PA2    PA3    PA4    PA5    PA6
## Q01   0.567  0.129 -0.120  0.229 -0.275 -0.188
## Q02 -0.280  0.369  0.172  0.115 -0.029  0.009
## Q03 -0.603  0.245  0.199 -0.022 -0.006  0.030
## Q04   0.606  0.082 -0.056  0.184 -0.090 -0.033
## Q05   0.523  0.043 -0.020  0.154 -0.167 -0.083
## Q06   0.548  0.024  0.488 -0.166  0.073 -0.006
## Q07   0.662 -0.026  0.223  0.030  0.107  0.057
## Q08   0.545  0.488 -0.272 -0.214  0.096 -0.020
## Q09 -0.266  0.462  0.124  0.210  0.097  0.032
## Q10   0.405 -0.005  0.172 -0.090 -0.148  0.024
## Q11   0.644  0.312 -0.199 -0.270  0.085 -0.037
## Q12   0.641 -0.099  0.063  0.154  0.047 -0.067
## Q13   0.650  0.024  0.223 -0.058  0.061 -0.134
## Q14   0.626 -0.036  0.161  0.056  0.011  0.013
```

```
## Q15  0.580 -0.007  0.072 -0.152 -0.188  0.436
## Q16  0.661 -0.016 -0.109  0.138 -0.283  0.094
## Q17  0.629  0.355 -0.155 -0.150  0.038  0.006
## Q18  0.683 -0.039  0.277  0.041  0.092 -0.099
## Q19 -0.395  0.267  0.110  0.060 -0.052  0.022
## Q20  0.412 -0.171 -0.250  0.190  0.241  0.114
## Q21  0.644 -0.099 -0.110  0.270  0.283  0.099
## Q22 -0.279  0.291  0.050  0.284  0.047  0.114
## Q23 -0.130  0.182  0.081  0.235  0.011  0.077
##
##                  PA1   PA2   PA3   PA4   PA5   PA6
## SS loadings    6.786 1.140 0.827 0.667 0.452 0.324
## Proportion Var 0.295 0.050 0.036 0.029 0.020 0.014
## Cumulative Var 0.295 0.345 0.381 0.410 0.429 0.443
```

12.11.2 Check the structure matrix

```
print(paf$Structure, cutoff=0, digits=3)
```

```
##
## Loadings:
##      PA1    PA2    PA3    PA4    PA5    PA6
## Q01  0.567  0.129 -0.120  0.229 -0.275 -0.188
## Q02 -0.280  0.369  0.172  0.115 -0.029  0.009
## Q03 -0.603  0.245  0.199 -0.022 -0.006  0.030
## Q04  0.606  0.082 -0.056  0.184 -0.090 -0.033
## Q05  0.523  0.043 -0.020  0.154 -0.167 -0.083
## Q06  0.548  0.024  0.488 -0.166  0.073 -0.006
## Q07  0.662 -0.026  0.223  0.030  0.107  0.057
## Q08  0.545  0.488 -0.272 -0.214  0.096 -0.020
## Q09 -0.266  0.462  0.124  0.210  0.097  0.032
## Q10  0.405 -0.005  0.172 -0.090 -0.148  0.024
## Q11  0.644  0.312 -0.199 -0.270  0.085 -0.037
## Q12  0.641 -0.099  0.063  0.154  0.047 -0.067
## Q13  0.650  0.024  0.223 -0.058  0.061 -0.134
## Q14  0.626 -0.036  0.161  0.056  0.011  0.013
## Q15  0.580 -0.007  0.072 -0.152 -0.188  0.436
## Q16  0.661 -0.016 -0.109  0.138 -0.283  0.094
## Q17  0.629  0.355 -0.155 -0.150  0.038  0.006
## Q18  0.683 -0.039  0.277  0.041  0.092 -0.099
## Q19 -0.395  0.267  0.110  0.060 -0.052  0.022
## Q20  0.412 -0.171 -0.250  0.190  0.241  0.114
## Q21  0.644 -0.099 -0.110  0.270  0.283  0.099
## Q22 -0.279  0.291  0.050  0.284  0.047  0.114
## Q23 -0.130  0.182  0.081  0.235  0.011  0.077
```

```
##
##                   PA1   PA2   PA3   PA4   PA5   PA6
## SS loadings     6.786 1.140 0.827 0.667 0.452 0.324
## Proportion Var  0.295 0.050 0.036 0.029 0.020 0.014
## Cumulative Var  0.295 0.345 0.381 0.410 0.429 0.443
```

12.11.3   Check eigenvalues

```
paf$e.values[1:6]
```

```
## [1] 7.2900471 1.7388287 1.3167515 1.2271982 0.9878779 0.8953304
```

12.11.4   Check communalities

- Communality for each variable: the percentage of variance that can be explained by the retained factors.
- Retained factors should explain more of the variance in each variable.

```
paf$communality
```

```
##       Q01       Q02       Q03       Q04       Q05       Q06       Q07       Q08
## 0.5170176 0.2585136 0.4643374 0.4196524 0.3341637 0.5720655 0.5042725 0.6649413 0.35
##       Q13       Q14       Q15       Q16       Q17       Q18       Q19       Q20
## 0.4973541 0.4223263 0.5902303 0.5571656 0.5700891 0.5655104 0.2467731 0.3686202 0.59
```

12.12   Step 6: Perform factor analysis (with re-duced number of factors)

```
paf1 <- fa(raq,
nfactors = 2,
fm="pa",
max.iter = 100,
rotate = "none")

paf1
```

```
## Factor Analysis using method =  pa
## Call: fa(r = raq, nfactors = 2, rotate = "none", max.iter = 100, fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1   PA2    h2   u2 com
## Q01   0.56  0.12 0.324 0.68 1.1
## Q02  -0.28  0.39 0.228 0.77 1.8
## Q03  -0.61  0.25 0.430 0.57 1.3
## Q04   0.61  0.09 0.377 0.62 1.0
## Q05   0.52  0.05 0.276 0.72 1.0
```

```
## Q06   0.53   0.04 0.282 0.72 1.0
## Q07   0.66 -0.01 0.437 0.56 1.0
## Q08   0.53   0.40 0.445 0.56 1.9
## Q09 -0.27   0.46 0.287 0.71 1.6
## Q10   0.40   0.00 0.163 0.84 1.0
## Q11   0.63   0.27 0.472 0.53 1.3
## Q12   0.64 -0.08 0.421 0.58 1.0
## Q13   0.65   0.04 0.421 0.58 1.0
## Q14   0.63 -0.02 0.396 0.60 1.0
## Q15   0.56   0.00 0.315 0.68 1.0
## Q16   0.65 -0.01 0.428 0.57 1.0
## Q17   0.63   0.34 0.511 0.49 1.5
## Q18   0.68 -0.02 0.461 0.54 1.0
## Q19 -0.40   0.28 0.238 0.76 1.8
## Q20   0.40 -0.15 0.187 0.81 1.3
## Q21   0.63 -0.07 0.403 0.60 1.0
## Q22 -0.28   0.29 0.161 0.84 2.0
## Q23 -0.13   0.19 0.053 0.95 1.8
##
##                             PA1  PA2
## SS loadings             6.67 1.04
## Proportion Var          0.29 0.05
## Cumulative Var          0.29 0.34
## Proportion Explained  0.86 0.14
## Cumulative Proportion 0.86 1.00
##
## Mean item complexity =  1.3
## Test of the hypothesis that 2 factors are sufficient.
##
## The degrees of freedom for the null model are  253  and the objective function was  7.55 with
## The degrees of freedom for the model are 208  and the objective function was  1.23
##
## The root mean square of the residuals (RMSR) is  0.05
## The df corrected root mean square of the residuals is  0.05
##
## The harmonic number of observations is  2571 with the empirical chi square  3114.53  with prob
## The total number of observations was  2571  with Likelihood Chi Square =  3155.34  with prob <
##
## Tucker Lewis Index of factoring reliability =  0.812
## RMSEA index =  0.074  and the 90 % confidence intervals are  0.072 0.077
## BIC =  1522.12
## Fit based upon off diagonal values = 0.97
## Measures of factor score adequacy
##                                                   PA1  PA2
## Correlation of (regression) scores with factors   0.96 0.78
## Multiple R square of scores with factors          0.92 0.61
```

```
## Minimum correlation of possible factor scores    0.83 0.23
plot(paf1)
```
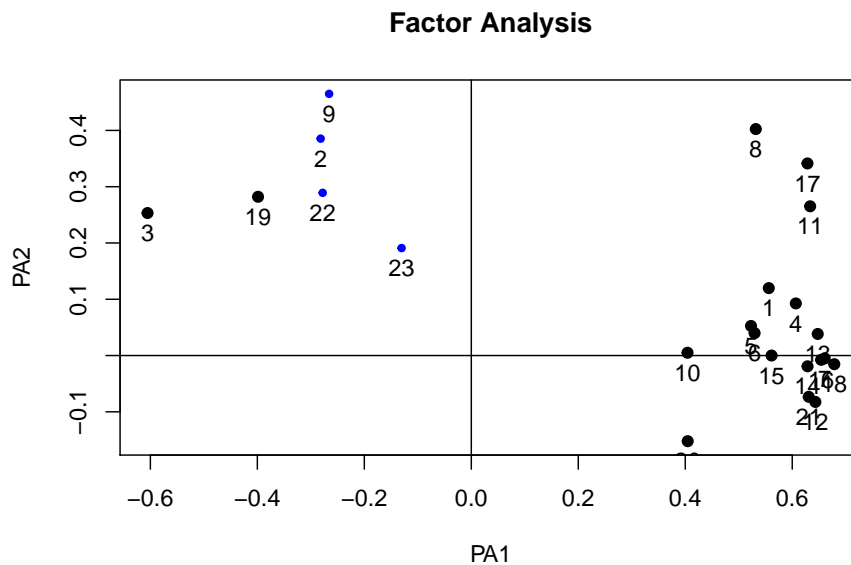
Factor Analysis

12.13   Factor analysis rotation

What is rotation?

- It is possible that variables load "highly" onto one factor and "medium" onto another
- By rotating the factor axes, the variables are aligned with the factors that they load onto most
- This helps us discriminate between factors

There are different methods of rotation

- **Orthogonal rotation:** Assumes that factors are unrelated and keeps them that way
- **Oblique rotation:** Assumes that factors might be related and allows them to be correlated after rotation

  Are factors related? -Theoretical: Do we have logical reason for thinking they could be connected? -Based on data: Does the factor plot suggest independence or relatedness?

12.14 Step 7: Rotation

- Perform factor analysis (with rotation)

```
paf2 <- fa(raq,
nfactors = 2,
fm="pa",
max.iter = 100,
rotate = "oblimin")

paf2
```

```
## Factor Analysis using method =  pa
## Call: fa(r = raq, nfactors = 2, rotate = "oblimin", max.iter = 100,
##     fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1   PA2   h2    u2 com
## Q01  0.57  0.03 0.324 0.68 1.0
## Q02 -0.12  0.44 0.228 0.77 1.2
## Q03 -0.48  0.36 0.430 0.57 1.8
## Q04  0.61 -0.01 0.377 0.62 1.0
## Q05  0.52 -0.03 0.276 0.72 1.0
## Q06  0.52 -0.05 0.282 0.72 1.0
## Q07  0.63 -0.11 0.437 0.56 1.1
## Q08  0.66  0.32 0.445 0.56 1.4
## Q09 -0.08  0.51 0.287 0.71 1.0
## Q10  0.39 -0.06 0.163 0.84 1.1
## Q11  0.70  0.16 0.472 0.53 1.1
## Q12  0.58 -0.19 0.421 0.58 1.2
## Q13  0.63 -0.07 0.421 0.58 1.0
## Q14  0.59 -0.12 0.396 0.60 1.1
## Q15  0.53 -0.09 0.315 0.68 1.1
## Q16  0.62 -0.12 0.428 0.57 1.1
## Q17  0.73  0.24 0.511 0.49 1.2
## Q18  0.64 -0.13 0.461 0.54 1.1
## Q19 -0.27  0.35 0.238 0.76 1.9
## Q20  0.33 -0.22 0.187 0.81 1.8
## Q21  0.57 -0.18 0.403 0.60 1.2
## Q22 -0.15  0.34 0.161 0.84 1.4
## Q23 -0.05  0.21 0.053 0.95 1.1
##
##                        PA1  PA2
## SS loadings           6.33 1.39
## Proportion Var        0.28 0.06
## Cumulative Var        0.28 0.34
## Proportion Explained  0.82 0.18
```

```
## Cumulative Proportion 0.82 1.00
##
##  With factor correlations of
##       PA1   PA2
## PA1  1.00 -0.22
## PA2 -0.22  1.00
##
## Mean item complexity =  1.2
## Test of the hypothesis that 2 factors are sufficient.
##
## The degrees of freedom for the null model are  253  and the objective function was
## The degrees of freedom for the model are 208  and the objective function was  1.23
##
## The root mean square of the residuals (RMSR) is  0.05
## The df corrected root mean square of the residuals is  0.05
##
## The harmonic number of observations is  2571 with the empirical chi square  3114.53
## The total number of observations was  2571  with Likelihood Chi Square =  3155.34
##
## Tucker Lewis Index of factoring reliability =  0.812
## RMSEA index =  0.074  and the 90 % confidence intervals are  0.072 0.077
## BIC =  1522.12
## Fit based upon off diagonal values = 0.97
## Measures of factor score adequacy
##                                                   PA1  PA2
## Correlation of (regression) scores with factors   0.96 0.81
## Multiple R square of scores with factors          0.91 0.65
## Minimum correlation of possible factor scores     0.82 0.30
```

```
plot(paf1)
```

Factor Analysis


```
plot(paf2)
```

Factor Analysis
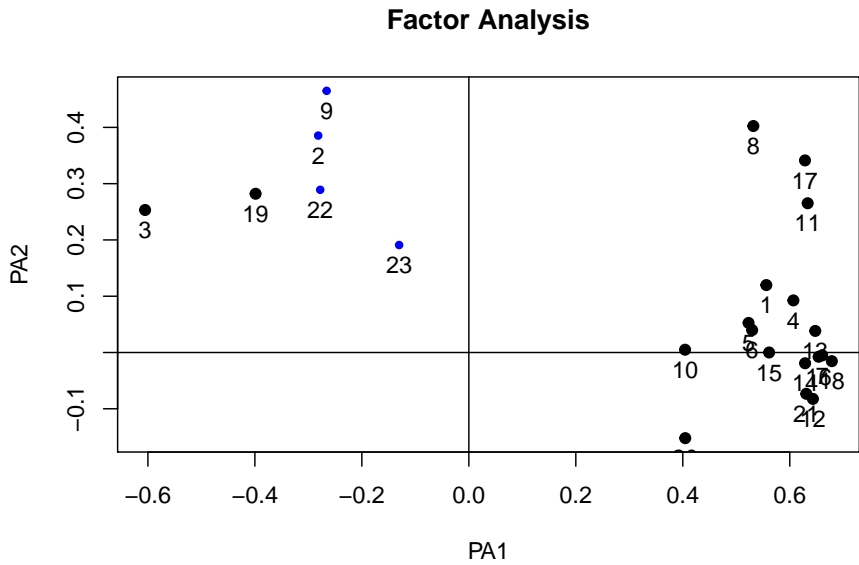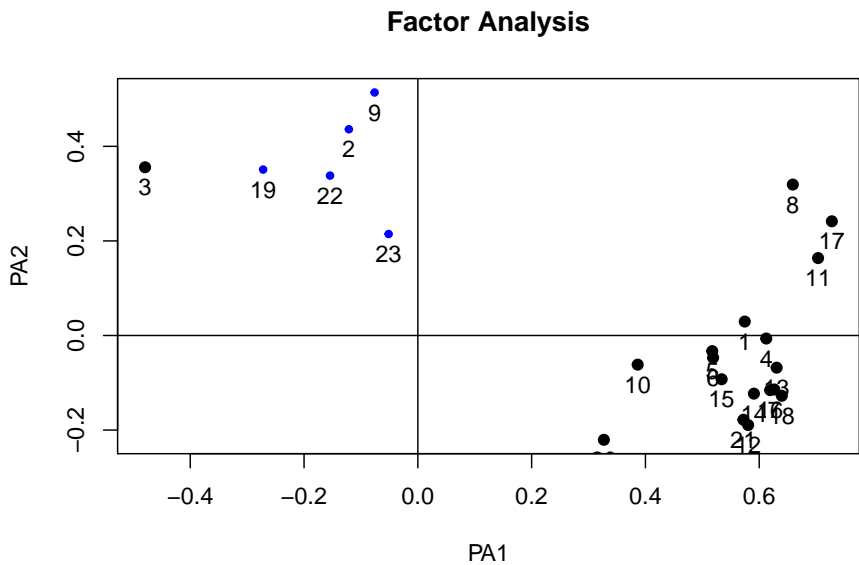
12.15   Reliability / internal consistency

12.15.1   Cronbach's Alpha

- An expansion of the split-half reliability concept
- Alpha takes all possible combination of items and assesses their relationship to each other
- High values above 0.7 suggest internal consistency among items

12.15.2   Chronbach's Alpha in R

- We can use the *alpha()* function in the psych package

```
library(psych)

alpha(raq)
```

```
## Warning in alpha(raq): Some items were negatively correlated with the total scale a
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option

## Some items ( Q02 Q03 Q09 Q19 Q22 Q23 ) were negatively correlated with the total sca
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option

##
## Reliability analysis
## Call: alpha(x = raq)
##
##   raw_alpha std.alpha G6(smc) average_r S/N    ase mean   sd median_r
##       0.75      0.77    0.83      0.13 3.4 0.0065  3.3 0.39     0.23
##
##  lower alpha upper     95% confidence boundaries
## 0.74 0.75 0.77
##
##  Reliability if an item is dropped:
##     raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## Q01      0.73      0.76    0.82      0.12 3.1   0.0071 0.071  0.23
## Q02      0.77      0.79    0.84      0.15 3.8   0.0061 0.071  0.25
## Q03      0.79      0.81    0.85      0.16 4.2   0.0055 0.059  0.25
## Q04      0.73      0.75    0.82      0.12 3.0   0.0072 0.070  0.22
## Q05      0.74      0.76    0.82      0.12 3.1   0.0071 0.072  0.22
## Q06      0.73      0.76    0.82      0.12 3.1   0.0072 0.072  0.23
## Q07      0.73      0.75    0.82      0.12 3.0   0.0074 0.069  0.22
## Q08      0.73      0.76    0.82      0.12 3.1   0.0071 0.072  0.23
## Q09      0.78      0.79    0.84      0.15 3.8   0.0058 0.071  0.25
## Q10      0.74      0.76    0.83      0.13 3.3   0.0068 0.074  0.23
## Q11      0.73      0.75    0.81      0.12 3.0   0.0072 0.069  0.22
```

```
## Q12       0.73        0.75        0.82       0.12 3.1   0.0072 0.069   0.22
## Q13       0.73        0.75        0.82       0.12 3.0   0.0073 0.069   0.22
## Q14       0.73        0.75        0.82       0.12 3.1   0.0072 0.070   0.22
## Q15       0.73        0.76        0.82       0.12 3.1   0.0071 0.071   0.22
## Q16       0.73        0.75        0.82       0.12 3.0   0.0072 0.069   0.22
## Q17       0.73        0.75        0.81       0.12 3.0   0.0072 0.070   0.22
## Q18       0.72        0.75        0.81       0.12 3.0   0.0074 0.068   0.22
## Q19       0.78        0.80        0.85       0.15 4.0   0.0057 0.067   0.26
## Q20       0.75        0.77        0.83       0.13 3.3   0.0067 0.073   0.25
## Q21       0.73        0.75        0.82       0.12 3.1   0.0072 0.069   0.22
## Q22       0.77        0.79        0.84       0.15 3.8   0.0059 0.071   0.26
## Q23       0.77        0.79        0.84       0.14 3.7   0.0061 0.074   0.26
##
##   Item statistics
##         n   raw.r  std.r  r.cor r.drop mean    sd
## Q01 2571  0.5598  0.581  0.564  0.492  3.6 0.83
## Q02 2571 -0.0116 -0.018 -0.114 -0.105  4.4 0.85
## Q03 2571 -0.3356 -0.361 -0.465 -0.435  3.4 1.08
## Q04 2571  0.6064  0.618  0.606  0.533  3.2 0.95
## Q05 2571  0.5365  0.546  0.516  0.454  3.3 0.96
## Q06 2571  0.5709  0.560  0.547  0.478  3.8 1.12
## Q07 2571  0.6409  0.636  0.635  0.560  3.1 1.10
## Q08 2571  0.5646  0.582  0.578  0.493  3.8 0.87
## Q09 2571  0.0587  0.020 -0.068 -0.081  3.2 1.26
## Q10 2571  0.4300  0.442  0.391  0.346  3.7 0.88
## Q11 2571  0.6078  0.628  0.633  0.540  3.7 0.88
## Q12 2571  0.5909  0.602  0.593  0.519  2.8 0.92
## Q13 2571  0.6288  0.637  0.634  0.559  3.6 0.95
## Q14 2571  0.6056  0.609  0.596  0.528  3.1 1.00
## Q15 2571  0.5433  0.550  0.526  0.457  3.2 1.01
## Q16 2571  0.5965  0.615  0.612  0.525  3.1 0.92
## Q17 2571  0.6329  0.650  0.653  0.568  3.5 0.88
## Q18 2571  0.6534  0.653  0.656  0.578  3.4 1.05
## Q19 2571 -0.1316 -0.157 -0.264 -0.248  3.7 1.10
## Q20 2571  0.3705  0.375  0.326  0.265  2.4 1.04
## Q21 2571  0.5922  0.598  0.591  0.514  2.8 0.98
## Q22 2571 -0.0063 -0.027 -0.127 -0.121  3.1 1.04
## Q23 2571  0.1030  0.084 -0.014 -0.013  2.6 1.04
##
## Non missing response frequency for each item
##         1    2    3    4    5 miss
## Q01 0.02 0.07 0.29 0.52 0.11    0
## Q02 0.01 0.04 0.08 0.31 0.56    0
## Q03 0.03 0.17 0.34 0.26 0.19    0
## Q04 0.05 0.17 0.36 0.37 0.05    0
## Q05 0.04 0.18 0.29 0.43 0.06    0
```

```
## Q06 0.06 0.10 0.13 0.44 0.27    0
## Q07 0.09 0.24 0.26 0.34 0.07    0
## Q08 0.03 0.06 0.19 0.58 0.15    0
## Q09 0.08 0.28 0.23 0.20 0.20    0
## Q10 0.02 0.10 0.18 0.57 0.14    0
## Q11 0.02 0.06 0.22 0.53 0.16    0
## Q12 0.09 0.23 0.46 0.20 0.02    0
## Q13 0.03 0.12 0.25 0.48 0.12    0
## Q14 0.07 0.18 0.38 0.31 0.06    0
## Q15 0.06 0.18 0.30 0.39 0.07    0
## Q16 0.06 0.16 0.42 0.33 0.04    0
## Q17 0.03 0.10 0.27 0.52 0.08    0
## Q18 0.06 0.12 0.31 0.37 0.14    0
## Q19 0.02 0.15 0.22 0.33 0.29    0
## Q20 0.22 0.37 0.25 0.15 0.02    0
## Q21 0.09 0.29 0.34 0.26 0.02    0
## Q22 0.05 0.26 0.34 0.26 0.10    0
## Q23 0.12 0.42 0.27 0.12 0.06    0
```

- Here we get a warning that some of the items are negatively correlated and we should probably reverse them.
- The decision to do so should be based on the logic of the questions themselves - check first
- However, since cronbach's alpha is designed to check internal consistency related to a single construct, we would expect that negative correlations would only result from:
  - Items that are designed to be reverse-scored
  - Questions that are related to another factor or construct
- Let's check the questionnaire
  - (Q02, Q03, Q09, Q19, Q22, Q23):

SD = Strongly Disagree, D = Disagree, N = Neither, A = Agree, SA = Strongly Agree

		SD	D	N	A	SA
1	Statistics make me cry	O	O	O	O	O
2	My friends will think I'm stupid for not being able to cope with R	O	O	O	O	O
3	Standard deviations excite me	O	O	O	O	O
4	I dream that Pearson is attacking me with correlation coefficients	O	O	O	O	O
5	I don't understand statistics	O	O	O	O	O
6	I have little experience of computers	O	O	O	O	O
7	All computers hate me	O	O	O	O	O
8	I have never been good at mathematics	O	O	O	O	O
9	My friends are better at statistics than me	O	O	O	O	O
10	Computers are useful only for playing games	O	O	O	O	O
11	I did badly at mathematics at school	O	O	O	O	O
12	People try to tell you that R makes statistics easier to understand but it doesn't	O	O	O	O	O
13	I worry that I will cause irreparable damage because of my incompetence with computers	O	O	O	O	O
14	Computers have minds of their own and deliberately go wrong whenever I use them	O	O	O	O	O
15	Computers are out to get me	O	O	O	O	O
16	I weep openly at the mention of central tendency	O	O	O	O	O
17	I slip into a coma whenever I see an equation	O	O	O	O	O
18	R always crashes when I try to use it	O	O	O	O	O
19	Everybody looks at me when I use R	O	O	O	O	O
20	I can't sleep for thoughts of eigenvectors	O	O	O	O	O
21	I wake up under my duvet thinking that I am trapped under a normal distribution	O	O	O	O	O
22	My friends are better at R than I am	O	O	O	O	O

- It is possible to run the analysis with automatic reversal of negatively-
  correlated items

```
alpha(raq, check.keys=TRUE)
```

```
## Warning in alpha(raq, check.keys = TRUE): Some items were negatively correlated with
##  This is indicated by a negative sign for the variable name.

##
## Reliability analysis
## Call: alpha(x = raq, check.keys = TRUE)
##
##   raw_alpha std.alpha G6(smc) average_r S/N    ase mean   sd median_r
##        0.89      0.89    0.91      0.27 8.3 0.0031  3.1 0.54     0.27
##
##  lower alpha upper     95% confidence boundaries
## 0.88 0.89 0.9
##
##  Reliability if an item is dropped:
##       raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## Q01        0.88      0.89    0.90      0.26 7.9   0.0032 0.016  0.27
## Q02-       0.89      0.89    0.91      0.28 8.4   0.0031 0.016  0.28
## Q03-       0.88      0.89    0.90      0.26 7.8   0.0033 0.017  0.26
## Q04        0.88      0.89    0.90      0.26 7.8   0.0033 0.016  0.26
## Q05        0.89      0.89    0.90      0.27 8.0   0.0032 0.017  0.27
## Q06        0.88      0.89    0.90      0.27 8.0   0.0032 0.016  0.27
## Q07        0.88      0.89    0.90      0.26 7.7   0.0034 0.016  0.26
## Q08        0.89      0.89    0.90      0.27 8.0   0.0032 0.016  0.27
## Q09-       0.89      0.89    0.91      0.28 8.4   0.0030 0.016  0.28
## Q10        0.89      0.89    0.90      0.27 8.2   0.0032 0.017  0.28
## Q11        0.88      0.89    0.90      0.26 7.8   0.0033 0.016  0.26
## Q12        0.88      0.89    0.90      0.26 7.7   0.0033 0.016  0.26
## Q13        0.88      0.89    0.90      0.26 7.7   0.0033 0.016  0.26
## Q14        0.88      0.89    0.90      0.26 7.8   0.0033 0.016  0.26
## Q15        0.88      0.89    0.90      0.26 7.9   0.0033 0.017  0.27
## Q16        0.88      0.89    0.90      0.26 7.7   0.0033 0.016  0.26
## Q17        0.88      0.89    0.90      0.26 7.8   0.0033 0.016  0.26
## Q18        0.88      0.88    0.90      0.26 7.7   0.0034 0.016  0.26
## Q19-       0.89      0.89    0.90      0.27 8.2   0.0032 0.017  0.29
## Q20        0.89      0.89    0.90      0.27 8.2   0.0032 0.017  0.28
## Q21        0.88      0.89    0.90      0.26 7.7   0.0033 0.016  0.26
## Q22-       0.89      0.89    0.91      0.28 8.4   0.0031 0.016  0.29
## Q23-       0.89      0.90    0.91      0.28 8.7   0.0030 0.014  0.29
##
##  Item statistics
##        n raw.r std.r r.cor r.drop mean   sd
## Q01 2571  0.55  0.57  0.54   0.50  3.6 0.83
```

```
## Q02- 2571  0.36  0.36  0.31   0.30  1.6 0.85
## Q03- 2571  0.65  0.64  0.62   0.59  2.6 1.08
## Q04  2571  0.61  0.61  0.59   0.55  3.2 0.95
## Q05  2571  0.54  0.55  0.52   0.48  3.3 0.96
## Q06  2571  0.56  0.55  0.53   0.49  3.8 1.12
## Q07  2571  0.67  0.67  0.65   0.62  3.1 1.10
## Q08  2571  0.51  0.53  0.51   0.46  3.8 0.87
## Q09- 2571  0.37  0.35  0.30   0.28  2.8 1.26
## Q10  2571  0.44  0.45  0.40   0.38  3.7 0.88
## Q11  2571  0.63  0.64  0.63   0.58  3.7 0.88
## Q12  2571  0.65  0.65  0.64   0.60  2.8 0.92
## Q13  2571  0.65  0.65  0.64   0.60  3.6 0.95
## Q14  2571  0.64  0.64  0.62   0.59  3.1 1.00
## Q15  2571  0.59  0.59  0.56   0.53  3.2 1.01
## Q16  2571  0.66  0.67  0.65   0.61  3.1 0.92
## Q17  2571  0.61  0.62  0.61   0.56  3.5 0.88
## Q18  2571  0.68  0.68  0.67   0.63  3.4 1.05
## Q19- 2571  0.47  0.46  0.42   0.40  2.3 1.10
## Q20  2571  0.45  0.45  0.41   0.38  2.4 1.04
## Q21  2571  0.64  0.64  0.63   0.59  2.8 0.98
## Q22- 2571  0.37  0.36  0.31   0.30  2.9 1.04
## Q23- 2571  0.23  0.22  0.15   0.15  3.4 1.04
##
## Non missing response frequency for each item
##         1    2    3    4    5 miss
## Q01 0.02 0.07 0.29 0.52 0.11    0
## Q02 0.01 0.04 0.08 0.31 0.56    0
## Q03 0.03 0.17 0.34 0.26 0.19    0
## Q04 0.05 0.17 0.36 0.37 0.05    0
## Q05 0.04 0.18 0.29 0.43 0.06    0
## Q06 0.06 0.10 0.13 0.44 0.27    0
## Q07 0.09 0.24 0.26 0.34 0.07    0
## Q08 0.03 0.06 0.19 0.58 0.15    0
## Q09 0.08 0.28 0.23 0.20 0.20    0
## Q10 0.02 0.10 0.18 0.57 0.14    0
## Q11 0.02 0.06 0.22 0.53 0.16    0
## Q12 0.09 0.23 0.46 0.20 0.02    0
## Q13 0.03 0.12 0.25 0.48 0.12    0
## Q14 0.07 0.18 0.38 0.31 0.06    0
## Q15 0.06 0.18 0.30 0.39 0.07    0
## Q16 0.06 0.16 0.42 0.33 0.04    0
## Q17 0.03 0.10 0.27 0.52 0.08    0
## Q18 0.06 0.12 0.31 0.37 0.14    0
## Q19 0.02 0.15 0.22 0.33 0.29    0
## Q20 0.22 0.37 0.25 0.15 0.02    0
## Q21 0.09 0.29 0.34 0.26 0.02    0
```

```
## Q22 0.05 0.26 0.34 0.26 0.10    0
## Q23 0.12 0.42 0.27 0.12 0.06    0
```

Chapter 13

Course videos

If you have trouble accessing the embedded videos, please try using this link:

https://teesside.hosted.panopto.com/Panopto/Pages/Sessions/List.aspx?folderID=efba294e-fa08-4156-8347-ac2f00b25bc9