Lecture 3

- Assignment 1/R tips
- Data manipulation
- Compact graphics
- Exploring your data

Presenter: Dr John Betts

Week-by-week outline

Week Starting	Seminar	Topic	App Ses	A1	A2	Q/P	A3	Due Date
2/3/2026	1	Introduction to Data Science, R, review of basic statistics	-					
9/3/2026	2	Data visualisation	S1					
16/3/2026	3	Data manipulation	S2					
23/3/2026	4	Regression modelling	S3					
30/3/2026	5	Clustering	S4					
6/4/2026	-	Mid-semester Break						
13/4/2026	6	Classification using decision trees	S5					17/4/2026
20/4/2026	7	Improving and evaluating classifiers. Naïve Bayes classification	S6					
27/4/2026	8	Ensemble methods, Artificial Neural Networks	S7					
4/5/2026	9	Network analysis	S8					
11/5/2026	10	Introduction to text analysis	S9					15/5/2026
18/5/2026	11	Text analysis applications	Quiz/Prac					22/5/2026
25/5/2026	12	Text Network Analysis, Review of the unit, Assignment 3	S10,11,12					
1/6/2026		SWOT VAC	-					
8/6/2026		EXAM PERIOD	-					12/6/2026

Assessment details

Assignment 1, Due 17th April, Weighting 25%

- Covers data manipulation, visualisation, and data analysis using a variety of techniques. Submission is a written report and short video explaining the key findings of your research.

Assignment 2, Due 15th May, Weighting 20%

- Covers machine learning/artificial intelligence models using R. Submission is a written report and short video.

Assignment 3, Due 12th June, Weighting 25%

- Covers text analysis, networks and clustering using R. Submission is a written report and short video.

Quiz + Practical Activity, Week 11 (Due 22nd May), Weighting 30%

- You will do practical activities and quiz style questions under supervision during your applied session. Content will cover topics from Weeks 1 – 9.

Consultations

Clayton students: see additional information and resources, under the “Learning” tile.

https://learning.monash.edu/course/view.php?id=41077&section=5

Assignment 1

Assignment 1

MONASH University

Faculty of Information Technology

FIT3152 Data analytics – 2026: Assignment 1

Your task	• Analyse the country level predictors of confidence in **social organisations and how these change over time** using data from the World Values Survey. • This is an individual assignment.
Value	• This assignment is worth **25%** of your total marks for the unit. • It has **40** marks in total.
Suggested Length	• 8 – 10 A4 pages, approximately 2,000 words (for your report) + extra pages as appendix for your R script and report on how Generative AI used, if required. • Font size 11 or 12pt, single spacing.

Assignment 1

Due Date	**11.55pm Friday 17th April 2026**
Submission	• Submit a single PDF file **and** single video file on Moodle. • Note that submission of a video report is a hurdle requirement. • Use the naming convention: *FirstnameSecondnameID.{pdf, mp4, mov etc.}* • Turnitin will be used for similarity checking of all written submissions.
Generative AI Use	• In this assessment, you can use generative artificial intelligence (AI) in order to search for R functions and examples to perform tasks that you specify only. Any use of generative AI must be appropriately acknowledged (see Learn HQ).
Late Penalties	• 5% (**2** mark) deduction per calendar day for up to one week. • Submissions more than 7 calendar days after the due date will receive a mark of zero (0) and no assessment feedback will be provided.

Assignment 1

Instructions

Address each of the research questions below and report the results of your analysis and your interpretation of those results. Report any assumptions you've made in your analysis. Include your R code as an appendix. Your R code must be machine readable text as the university requires all student submissions to be processed by plagiarism detection software. See information on Generative AI below.

There are two options for compiling your written report:
(1) You can create your report using any word processor with your R code pasted in as machine-readable text as an appendix, and save as a pdf, or
(2) As an R Markdown document that contains the R code with results and discussion interleaved. Render this as a HTML file and save as a pdf.

Your video report should be less than 100MB in size. You may need to reduce the resolution of your original recording to achieve this. Use a standard file format such as .mp4, or mov for submission.

Assignment 1

Software

It is expected that you will use R for your data analysis, graphics and tables. You are free to use any R packages you need but must document these in your report and include in your R code.

Use of Generative AI

AI & Generative AI tools may be used in GUIDED ways within this assessment/task as per the guidelines provided.

In this assessment, you can use generative artificial intelligence (AI) in order to search for R functions and examples to perform tasks that you specify only. Any use of generative AI must be appropriately acknowledged (see Learn HQ).

If you do use Generative AI for your assignment, then you must include the statement "Generative AI was used in this assignment." in the introductory/first paragraph of your report. You must also include the following information as an appendix in your report: (1) the technology you used (e.g. ChatGPT), (2) the information that was generated (e.g. R code fragments), (3) the prompts used (i.e. the questions you asked), and (4) how the output was used in your work.

If you did not use generative AI in your assignment, then include the statement "Generative AI was not used in this assignment." in the introductory/first paragraph of your report.

Assignment 1

Questions

The World Values Survey (WVS) is an international research program that studies the social, political, economic, religious and cultural attitudes and values of people around the world. You can read more here: https://www.worldvaluessurvey.org/WVSContents.jsp.

For this assignment you will analyse data collected over Waves 1 - 7, from 1981 to 2022. The aim of this assignment is to understand country-level differences in participant responses and the predictors of confidence in social organisations, and how these responses and predictors of confidence have changed over time.

Social organisations include aspects of society such as religion, armed forces, the press, television, trade unions, police, the courts, government, banks, and international and environmental organisations etc. They are indicated in your data by column names having the prefix "C". Predictor variables (**attributes**) include personal information such as age and gender, happiness indicators, attitudes and values towards others, political and social views and participation.

Each student will be assigned a **different** subset of organisations and attributes to study. Your task is to analyse **all** the survey data assigned to you, with a **focus** on the country you have been allocated.

Assignment 1

1. Descriptive analysis. (5 Marks)

(a) Describe the data overall, including things such as dimension, data types, distribution of numerical responses, variety of non-numerical (text) responses, missing values, and anything else of interest or relevance.

Assignment 1

2. Focus country vs all other countries as a group (independent of time). (13 Marks)

For Question 2 ignore the effect of time. That is, do not separate your data by years or waves when answering the questions below.

(a) Identify your focus country from the accompanying list (**WVSFocusCountry.pdf**). How do participant responses for your focus country differ from the other countries in the survey (treating them as a group)?

(b) How well do participant responses (attributes) predict confidence in social organisations in your **focus country**? Which attributes seem to be the best predictors? Confidence in which social organisations can be more reliably predicted? Explain your reasoning.

(c) Repeat Question 2(b) for the **other countries** as a group. Which attributes are the strongest predictors? Confidence in which social organisations can be more reliably predicted? How do these results compare to those of your focus country?

Assignment 1

3. Focus country vs all other countries as a group (over time). (12 Marks)

For Question 3 study the effect of time by separating your data by years or waves when answering the questions below.

(a) How do participant responses for your **focus country** vary over time (using either years or successive waves)? Describe these changes over time and comment on whether they are significant or not. Perform the same analysis for the **other countries** (as a group) and compare the results with your focus country. What are the major differences between the two groups? Create a graphic enabling a reader to compare results (focus *vs* other countries) over time, for the most interesting results. Describe your reasoning for the design of the graphic.

(b) How does the ability of participant responses (attributes) to predict confidence in social organisations in your **focus country** change over time? Do the important attributes for predicting confidence change over time? Perform the same analysis for the **other countries** (as a group) and compare the results. What are the major differences between the two groups? Create a graphic enabling a reader to compare results (focus *vs* other countries) over time, for the strongest predictors. Describe your reasoning for the design of the graphic.

Assignment 1

4. Video Presentation: (Submission Hurdle and 4 Marks)

Record a short presentation using your smartphone, Zoom, or similar method. Your presentation should be approximately 5 minutes in length and summarise your main findings for Sections 1 – 3, as well as describing how you conducted your research, any assumptions made, and how you designed your graphics.

5 Overall considerations (6 Marks)

This includes: the quality and clarity of your reasoning and assumptions; the strength of support for your findings; the quality of your writing in general and communication of results;-the quality of your graphics throughout; the quality of your R coding.

Assignment 1

Data

The data for this assignment is a reduced version of the World Values Survey Waves 1 -7 data. The filename is "WVSExtract.csv". The data includes ordinal data coded on a numerical scale. For this assignment assume it is reasonable to treat these responses as numerical.

Create your individual data as follows:

```
rm(list = ls())
set.seed(12345678) # Your Student Number
VCData = read.csv("WVSExtract.csv")
VC = VCData[sample(1:nrow(VCData),100000, replace=FALSE),]
VC = VC[,c(1:3,sort(sample(4:50,25,replace=FALSE)),
sort(sample(51:65,8,replace=FALSE)))]
#write.csv(VC, "FIT3152A1Data_YourName.csv", row.names = FALSE)
```

You can save the “VC” file you created by uncommenting the last line above. You can then delete the WVSExtract.csv file you downloaded.

Locate your focus country using the accompanying document FocusCountryByID.pdf. A list matching country names with three letter code is in WVSCountryCodes.pdf.

Assignment 1

Data fields and brief descriptor

Most fields are on integer scales over varying range. The convention is that larger numbers generally indicate greater agreement with statement or frequency of occurrence. Some exceptions given below. Fields in bold indicate confidence in social organisations.

You can access more detail on each field in your data from the *WVS-7 Master Questionnaire 2017-2020 English.pdf*, linked from https://www.worldvaluessurvey.org/WVSDocumentationWV7.jsp. Use the question ID given in the **WVS Wave 7 Reference** in the table below.

Column Name	**Original Descriptor**	**WVS Wave 7 Reference**
Wave	Chronology of EVS-WVS waves	A_WAVE
Country	ISO 3166-1 alpha-3 country code	B_COUNTRY_ALPHA
Year	Year survey	A_YEAR
ILFam	Important in life: Family	Q1
ILFriends	Important in life: Friends	Q2
ILLeisure	Important in life: Leisure time	Q3
ILPolitics	Important in life: Politics	Q4
ILWork	Important in life: Work	Q5

Assignment 1

CChurches	Confidence: Churches	Q64
CArmedForces	Confidence: Armed Forces	Q65
CPress	Confidence: The Press	Q66
CUnions	Confidence: Labour Unions	Q68
CPolice	Confidence: The Police	Q69
CParliament	Confidence: Parliament	Q73
CCivilService	Confidence: The Civil Services	Q74
CTelevision	Confidence: Television	Q67
CGovernment	Confidence: The Government	Q71
CPolParties	Confidence: The Political Parties	Q72
CMajComp	Confidence: Major Companies	Q77
CEnvProt	Confidence: The Environmental Protection Movement	Q79
CWomensMvt	Confidence: The Womens' Movement	Q80
CCourts	Confidence: Justice System/Courts	Q70
CEU	Confidence: The European Union	Q82_EU

Assignment 1

JB EDA V00.R | VC

Filter

	Wave	Country	Year	ILFam	ILWork	ICQIndependence	ICQHardWork	ICQTolerance	ICQDetermination	ICQFaith	ACTReligion	ACTSport	ACTUnions
149999	5	JOR	2007	1	1	1	1	0	0	1	0	0	0
57968	5	UKR	2006	1	2	1	0	1	0	1	2	1	0
34203	7	AUS	2018	1	2	1	1	1	1	0	1	0	0
128452	6	BLR	2011	1	1	0	1	1	1	0	0	0	0
196160	7	NLD	2022	1	2	1	1	1	0	0	1	2	0
54028	7	USA	2017	2	1	0	1	1	1	0	0	0	0
45965	5	HUN	2009	2	2	1	0	1	1	0	0	0	0
102378	3	GEO	1996	1	1	1	1	1	0	0	0	0	0
84273	4	SGP	2002	1	3	1	0	1	0	1	-4	-4	-4
92075	5	FIN	2005	1	1	1	0	1	1	0	2	2	0
180253	6	MEX	2012	1	1	1	0	1	1	1	0	0	0
111049	3	NZL	1998	1	2	0	0	1	1	0	1	2	0
135207	5	GBR	2005	1	1	1	1	1	1	0	2	1	0
146977	7	MYS	2018	1	1	0	0	1	0	1	2	0	2
59242	7	CHN	2018	1	1	0	1	1	0	0	0	0	0
149092	5	CHE	2007	1	1	0	1	0	0	0	0	0	0
39163	4	ZAF	2001	1	1	0	0	1	0	1	-4	-4	-4
79507	7	GBR	2022	1	1	0	0	1	0	0	1	0	0
107645	3	SVK	1998	1	1	0	1	0	0	1	2	0	0
115588	7	ROU	2018	1	1	0	0	0	1	0	0	0	0
180425	6	ARM	2011	1	1	0	1	1	0	0	0	0	0
84826	2	NGA	1990	1	1	0	1	0	0	1	-4	-4	-4
53654	5	USA	2006	1	1	0	1	1	0	1	0	0	1
65630	5	UKR	2006	1	2	0	1	1	0	0	0	0	0
139335	5	GEO	2009	1	1	1	1	0	0	1	0	0	1
160800	6	MAR	2011	2	2	0	1	0	0	0	0	0	0
96936	3	FIN	1996	1	1	1	0	0	0	0	2	0	0

Assignment 1

Assignment 1 Notes

- Students who joined the unit late (and are not on the FocusCountryByID.pdf) need to email john.betts@monash.edu to be assigned a focus country.
- Data may contain missing/NA values. Check the survey documentation: WVS-7 Master Questionnaire 2017-2020 English.pdf
- It is likely many attributes will have low predictive power. The aim of the analysis is to find the "best" ones.

Assignment 1 Notes

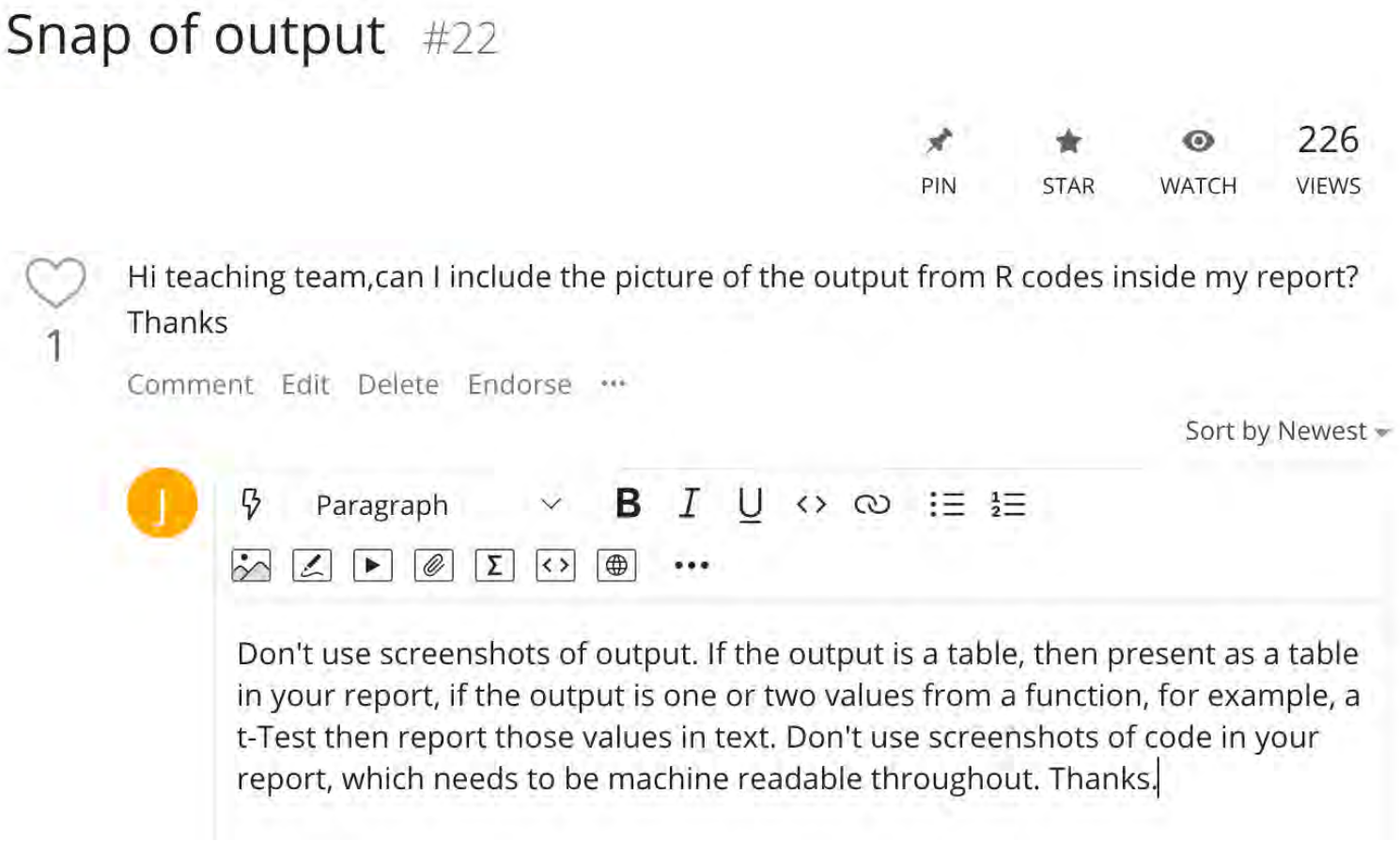
Examples of bad summaries in the assignment report.

Data Types: ordinal data coded on a numerical scale and hence treated as numerical data
Distribution of Numerical Attributes: summary statistics of data, (mean, median, mode)
Variety of Non-Numerical (text) Attributes: discuss data that is not numbers
Missing Values: sum of missing values, number of missing values in each column

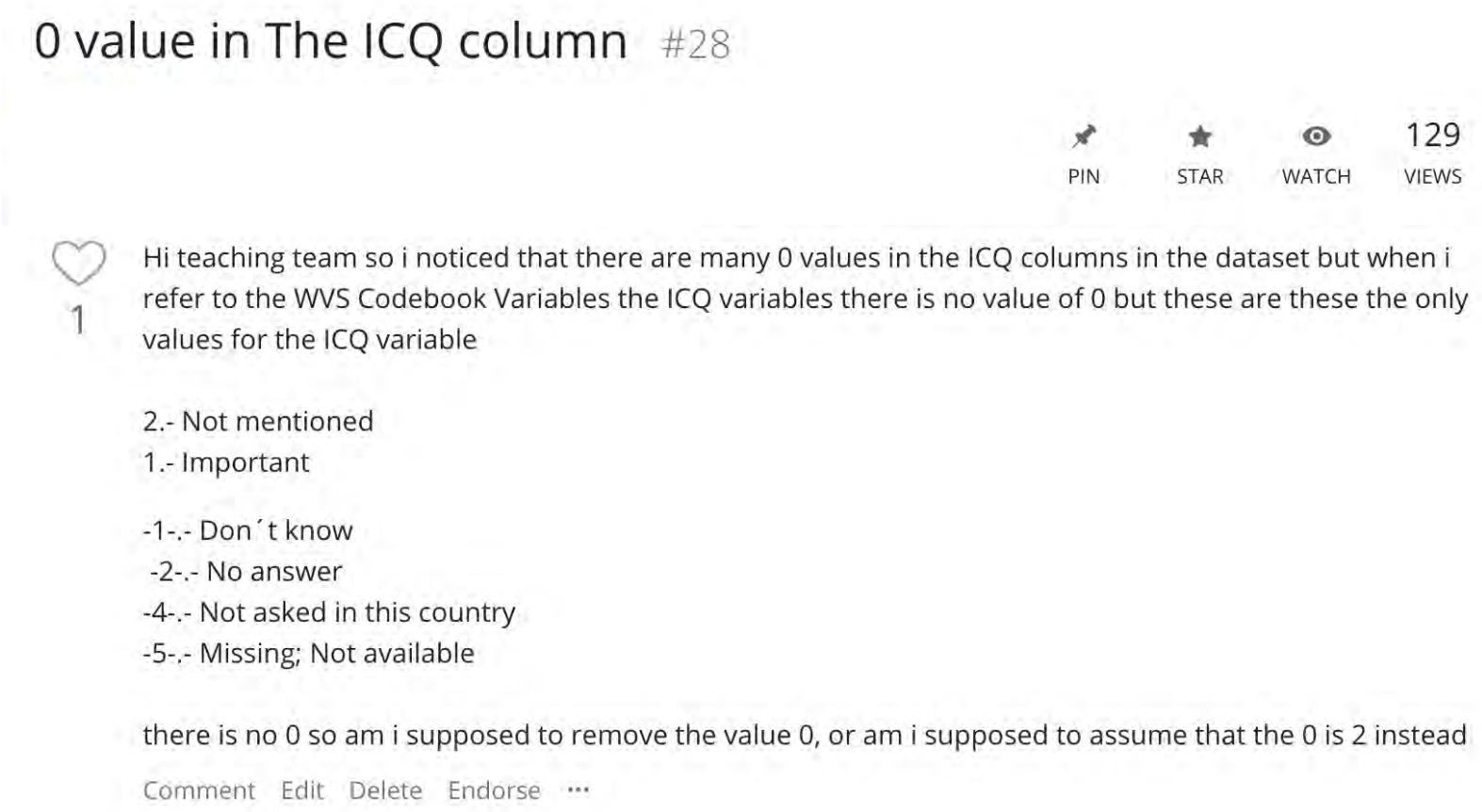
```
> str(cvbase)
'data.frame':	40000 obs. of  52 variables:
 $ employstatus_1     : int  NA NA NA NA 1 NA NA NA NA NA ...
 $ employstatus_2     : int  NA NA NA 1 NA 1 1 1 NA NA ...
 $ employstatus_3     : int  NA NA NA NA NA NA NA NA 1 1 ...
 $ employstatus_4     : int  NA NA NA NA 1 NA NA NA NA NA ...
 $ employstatus_5     : int  NA NA NA NA NA NA NA NA NA NA ...
 $ employstatus_6     : int  NA NA NA NA NA NA NA NA NA NA ...
 $ employstatus_7     : int  1 1 NA NA NA NA NA NA NA NA ...
 $ employstatus_8     : int  NA NA NA NA NA NA NA NA NA NA ...
 $ employstatus_9     : int  NA NA 1 NA 1 NA NA NA NA NA ...
 $ employstatus_10    : int  NA NA NA NA NA NA NA NA NA NA ...
 $ isoFriends_inPerson: int  0 0 0 0 1 5 0 4 2 1 ...
 $ isoOthPpl_inPerson : int  7 0 0 7 1 6 2 7 7 7 ...
 $ isoFriends_online  : int  0 5 6 7 7 6 7 7 0 2 ...
 $ isoOthPpl_online   : int  0 7 0 7 0 6 0 7 1 3 ...
 $ lone01             : int  1 2 3 3 3 4 1 4 2 2 ...
 $ lone02             : int  1 2 4 3 2 4 2 3 4 2 ...
 $ lone03             : int  1 2 1 3 2 5 1 3 2 2 ...
 $ happy              : int  6 10 6 3 6 9 6 5 6 4 ...
 $ lifeSat            : int  4 6 3 2 3 5 5 2 4 3 ...
 $ MLQ                : int  0 2 -3 1 1 2 2 -3 0 3 ...
 $ bor01              : int  0 1 3 0 0 1 3 -3 -1 -1 ...
 $ bor02              : int  -2 2 3 0 0 2 1 0 1 0 ...
 $ bor03              : int  0 -1 -1 0 -1 2 -3 -1 0 0 ...
 $ consp01            : int  5 3 10 5 7 8 8 10 7 5 ...
 $ consp02            : int  5 4 10 3 10 9 6 10 8 5 ...
 $ consp03            : int  5 3 6 3 5 8 5 10 5 5 ...
 $ rankOrdLife_1      : chr  "E" NA "E" "F" ...
 $ rankOrdLife_2      : chr  "A" NA "D" "B" ...
 $ rankOrdLife_3      : chr  "F" NA "F" "D" ...
 $ rankOrdLife_4      : chr  "B" NA "B" "A" ...
```

```
##  employstatus_9  employstatus_10 isoFriends_inPerson isoOthPpl_inPerson
##  Min.   :1       Min.   :1       Min.   :0.00        Min.   :0.000
##  1st Qu.:1       1st Qu.:1       1st Qu.:0.00        1st Qu.:0.000
##  Median :1       Median :1       Median :1.00        Median :1.000
##  Mean   :1       Mean   :1       Mean   :2.07        Mean   :1.955
##  3rd Qu.:1       3rd Qu.:1       3rd Qu.:4.00        3rd Qu.:3.000
##  Max.   :1       Max.   :1       Max.   :7.00        Max.   :7.000
##  NA's   :46138   NA's   :56601   NA's   :468         NA's   :749
##  isoFriends_online isoOthPpl_online     lone01          lone02
##  Min.   :0.000     Min.   :0.000    Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000     1st Qu.:0.000    1st Qu.:1.000   1st Qu.:2.000
##  Median :5.000     Median :2.000    Median :2.000   Median :3.000
##  Mean   :4.399     Mean   :2.855    Mean   :2.419   Mean   :2.665
##  3rd Qu.:7.000     3rd Qu.:5.000    3rd Qu.:3.000   3rd Qu.:4.000
##  Max.   :7.000     Max.   :7.000    Max.   :5.000   Max.   :5.000
##  NA's   :1358      NA's   :1667     NA's   :120     NA's   :173
##      lone03          happy           lifeSat           MLQ
##  Min.   :1.000   Min.   : 1.000   Min.   :1.000   Min.   :-3.0000
##  1st Qu.:1.000   1st Qu.: 5.000   1st Qu.:3.000   1st Qu.: 0.0000
##  Median :2.000   Median : 7.000   Median :4.000   Median : 1.0000
##  Mean   :2.079   Mean   : 6.333   Mean   :4.138   Mean   : 0.8439
##  3rd Qu.:3.000   3rd Qu.: 8.000   3rd Qu.:5.000   3rd Qu.: 2.0000
##  Max.   :5.000   Max.   :10.000   Max.   :6.000   Max.   : 3.0000
##  NA's   :198     NA's   :732      NA's   :161     NA's   :167
##      bor01             bor02             bor03             consp01
##  Min.   :-3.0000   Min.   :-3.00000   Min.   :-3.0000   Min.   : 0.000
##  1st Qu.:-1.0000   1st Qu.:-2.00000   1st Qu.:-1.0000   1st Qu.: 5.000
##  Median : 0.0000   Median : 0.00000   Median : 0.0000   Median : 7.000
##  Mean   : 0.3271   Mean   : 0.04387   Mean   : 0.3101   Mean   : 6.835
##  3rd Qu.: 2.0000   3rd Qu.: 2.00000   3rd Qu.: 2.0000   3rd Qu.: 9.000
##  Max.   : 3.0000   Max.   : 3.00000   Max.   : 3.0000   Max.   :10.000
##  NA's   :226       NA's   :244        NA's   :249       NA's   :2187
##     consp02           consp03        c19perBeh01     c19perBeh02
##  Min.   : 0.000   Min.   : 0.000   Min.   :-3.00   Min.   :-3.000
##  1st Qu.: 5.000   1st Qu.: 4.000   1st Qu.: 2.00   1st Qu.: 2.000
##  Median : 8.000   Median : 5.000   Median : 3.00   Median : 3.000
##  Mean   : 7.151   Mean   : 5.585   Mean   : 2.31   Mean   : 2.426
##  3rd Qu.: 9.000   3rd Qu.: 8.000   3rd Qu.: 3.00   3rd Qu.: 3.000
##  Max.   :10.000   Max.   :10.000   Max.   : 3.00   Max.   : 3.000
```

Assignment 1 Notes



Assignment 1 Notes



0 value in The ICQ column #28

PIN STAR WATCH 129 VIEWS

1

Hi teaching team so i noticed that there are many 0 values in the ICQ columns in the dataset but when i refer to the WVS Codebook Variables the ICQ variables there is no value of 0 but these are these the only values for the ICQ variable

2.- Not mentioned
1.- Important

-1-.- Don´t know
-2-.- No answer
-4-.- Not asked in this country
-5-.- Missing; Not available

there is no 0 so am i supposed to remove the value 0, or am i supposed to assume that the 0 is 2 instead

Comment Edit Delete Endorse ···

Assignment 1 Notes

Excellent observation! Something I hadn't noticed when choosing these attributions. I couldn't resolve the answer from the earlier code books I could locate. Have a look the screenshot of the first few rows of survey results for all ICQ attributes. The maximum number of 1s is 5 whereas in some cases the number of zeros exceeds 5. This suggests to me the "mentioned" response is 1 and the "not mentioned" response is 0.

ICQIndependence	ICQHardWork	ICQResonsibility	ICQImagination	ICQTolerance	ICQThrift	ICQDetermination	ICQFaith	ICQUnselfishness	ICQObedience
1	0	0	1	1	1	0	0	0	1
0	1	0	0	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0
0	0	1	0	1	0	1	1	0	0
0	1	1	0	0	1	0	0	0	1
0	0	1	0	1	1	0	0	0	1
1	0	1	0	1	0	1	0	0	0
1	0	0	1	1	1	0	0	0	0
0	1	1	0	1	1	0	1	0	0
1	0	1	1	0	0	0	0	1	1
1	0	0	0	1	1	0	1	0	1
0	0	0	1	1	1	1	0	0	1

Comment Edit Delete Endorse ···

R tips

R tips

Scripts:

- Very important: learn how to use these now if you've not done so already.

R Markdown:

- Useful if you're doing a job that requires a lot of routine reporting, but not essential. Also, Notebooks.

User-Defined Functions

- Useful, and they improve your R code. See also how to create anonymous functions defined on the fly.

Scripts

Scripts allow you to save your working from session to session.

- Use them to automate environment settings etc.
- Create a new script: File > New File > R Script
- Save with a filename
- Use “Source” to evaluate on the fly
- Note: # comments, pre-emptive text
- Next slide shows example from last week as a script…

Scripts

Week 02.R

Source on Save | Run | Source

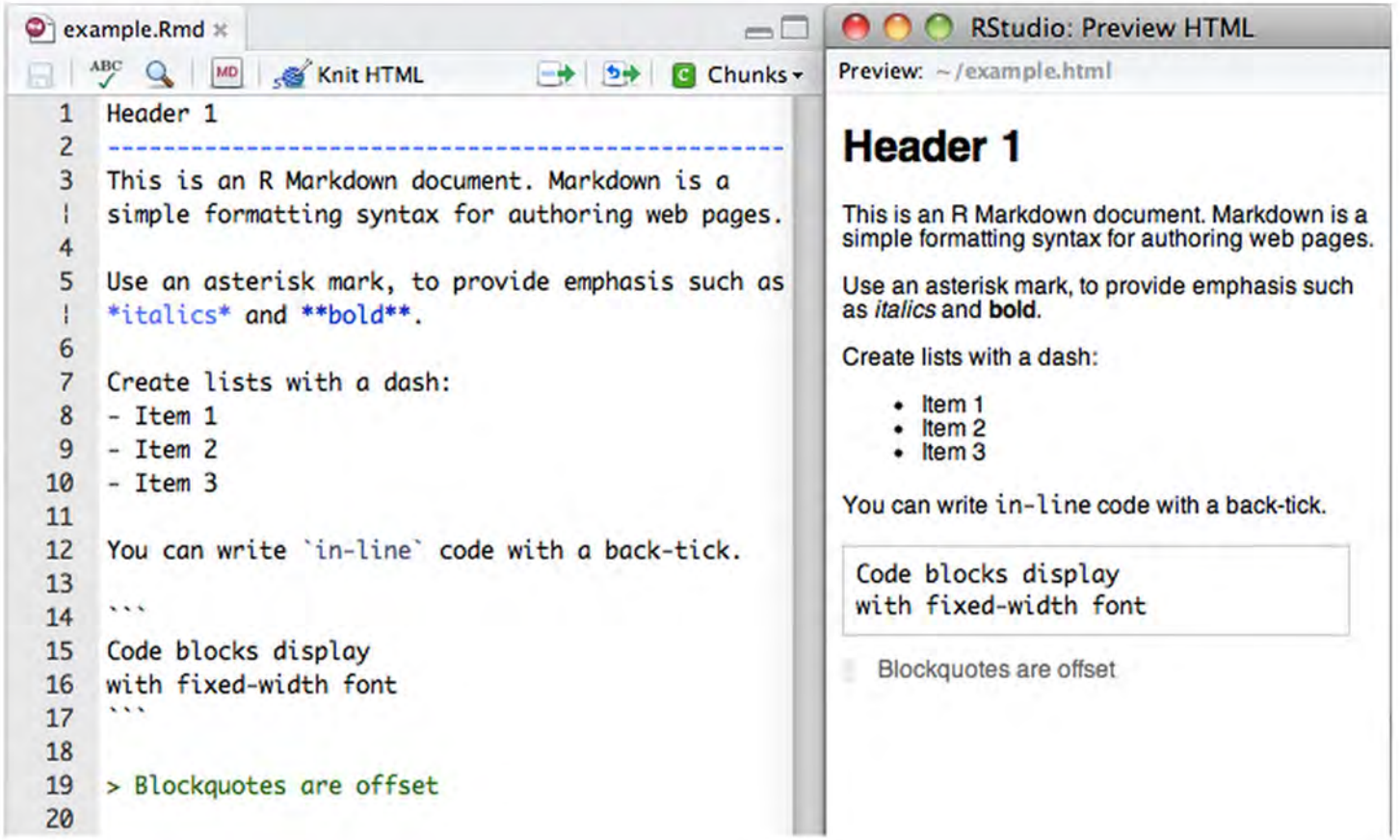
```
1  # LECTURE 2 examples
2  rm(list = ls()) #clean up environment
3  #install.packages("ggplot2")
4  #install.packages("lattice")
5  #install.packages("GGally")
6  library(ggplot2); library(lattice); library(GGally)
7
8  #Week 2 Video 2
9
10 iris
11 head(iris)
12 tail(iris)
13 dim(iris)
14 names(iris)
15 str(iris)
16 iris[10:15,]
17 iris[11,]
18 iris[10:20, "Sepal.Length"] # identify column by name
19 iris[10:20,1] # identify column by number
20 iris$Sepal.Length[10:20] # identify column first then select rows
21 summary(iris)
22
```

R Markdown

Is a package that enables the creation of HTML and PDF documents etc. based on your R session. You may choose to use it, but it is optional.

- You can embed R code and graphics.
- You can get started with R Markdown by creating a new R Markdown file in R Studio (the required files will be automatically installed).
- http://rmarkdown.rstudio.com/

R Markdown



- http://rmarkdown.rstudio.com/

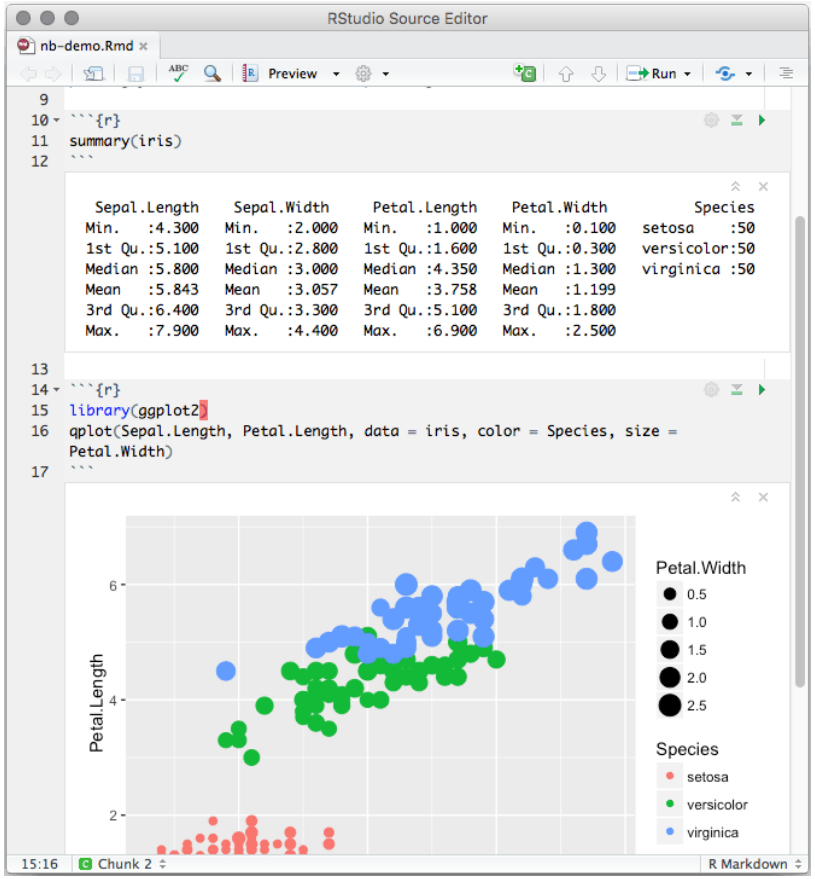
R Notebooks

Notebooks:

- These are HTML documents that enable the interleaving of text and chunks of executable code.
- File > New File > R Notebook

See:

https://rmarkdown.rstudio.com/



Source: https://bookdown.org/

Creating user-defined functions

It is possible to create named, user-defined, functions that can be saved between sessions using a script (see ATHR pp. 40 – 41).

Syntax:

```
> my_function <- function(arg1, arg2, …) {
> object <- Calculations(arg1, arg2, …)
> Return(object)
> }
```

Creating user-defined functions

Example:

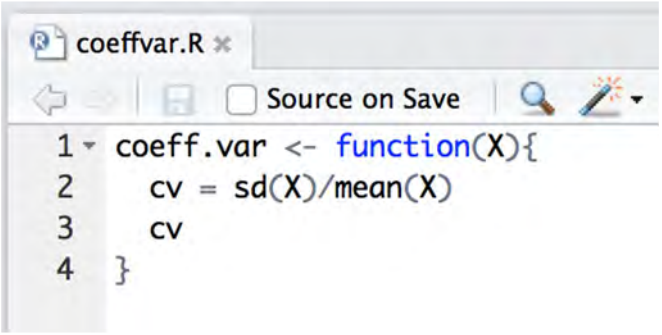
```
> coeff.var <- function(X){
> cv = sd(X)/mean(X)
> cv}

> Y = c(1, 2, 3, 4, 5, 6)
> coeff.var(Y)
[1] 0.5345225
```

Saving and re-using functions

In Rstudio:

- Create a new R script,
- Write function in script editor,
- Save as (filename.R)



To run function in a new session of R studio:

- Open and run script: code > source file (filename.R)

Data manipulation

Data manipulation

In this section we'll cover:

- Summarising data by groups
- Using the "by" function
- Creating summary columns
- Making data summary files
- Introduction to "dplyr"
- Recoding and indexing
- Using the Iris data as an example

Edgar Anderson's Iris data

50 samples from 3 species:

- Iris setosa, – virginica, – versicolor

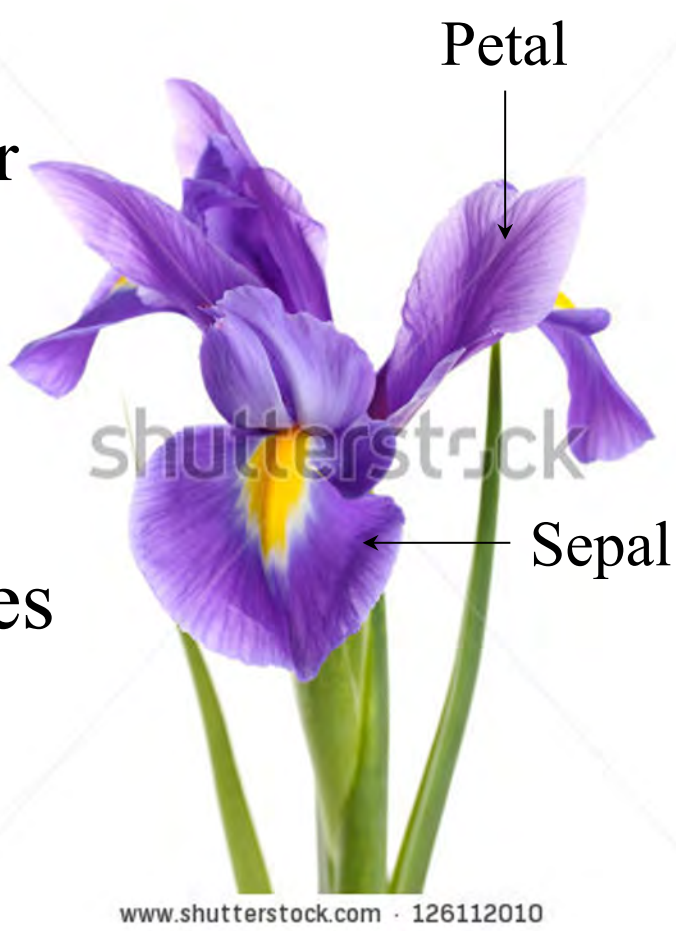
Four features measured:

- Sepal width and length
- Petal width and length

Is it possible to distinguish species using physical measurements?

- Data is packaged with R: "iris"

http://en.wikipedia.org/wiki/Iris_flower_data_set



Summarising data by groups

Data grouped by factors:

- Applying a function to a single column
- Applying a function to a group of columns

Why do we need to do this?

- To simplify the data, making comparisons easier
- Reduce data complexity, enabling further analysis

Print

> iris *# = print(iris)*

```
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
2            4.9         3.0          1.4         0.2     setosa
3            4.7         3.2          1.3         0.2     setosa
4            4.6         3.1          1.5         0.2     setosa
5            5.0         3.6          1.4         0.2     setosa
6            5.4         3.9          1.7         0.4     setosa
7            4.6         3.4          1.4         0.3     setosa
8            5.0         3.4          1.5         0.2     setosa
9            4.4         2.9          1.4         0.2     setosa
10           4.9         3.1          1.5         0.1     setosa
11           5.4         3.7          1.5         0.2     setosa
12           4.8         3.4          1.6         0.2     setosa
13           4.8         3.0          1.4         0.1     setosa
14           4.3         3.0          1.1         0.1     setosa
15           5.8         4.0          1.2         0.2     setosa
...
```

Two challenges

(1) Easy!

- Create a table of column means grouped by species.

(2) Harder!

- Create a CSV file containing the correlation between sepal length and sepal width, and petal length and petal width for each species.

High level view

Before we start, data analysis is easier if you have a high-level view of the data:

- 4 columns + 1 factor (Species)
- Two pairs of related columns: sepals & petals

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Setosa
Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Virginica
Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Versicolor

Challenge 1. Function: aggregate

The ‘aggregate’ function applies a function to data in individual columns grouped by a factor (or factors) and reports results as a data frame. To calculate averages:

- Note: columns referred to by their index [(number)] for compactness
  - > aggregate(iris[1:4], iris[5], mean)

```
     Species Sepal.Length Sepal.Width Petal.Length Petal.Width
1     setosa         5.01        3.43         1.46       0.246
2 versicolor         5.94        2.77         4.26       1.326
3  virginica         6.59        2.97         5.55       2.026
```

?aggregate

- Description

  aggregate(x, ...) : Splits the data into subsets, computes summary statistics for each, and returns the result in a convenient form.

- Usage

  ```
  aggregate(x, by, FUN, ..., simplify = TRUE)
  ```

- Arguments

  X : An R object.

  By : List of grouping elements

  FUN : Function to compute the summary statistics

  Simplify : Indicates whether results should be simplified to a vector or matrix if possible.

Challenge 2: correlation

Recall, correlation:

- Gives us an idea of the strength of the (linear) relationship between variables.
- Knowing the strength of this relationship is sometimes used to reduce the number of variables we need to analyse. That is, *if two variables are strongly correlated, we may only need to analyse one of them!*
- We'll look at several options for viewing the correlation between variables.

Correlation matrix

The pairwise correlation between each numeric variable

```
> round(cor(iris[1:4]), digits = 3)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Sepal.Length	1.000	-0.118	0.872	0.818
Sepal.Width	-0.118	1.000	-0.428	-0.366
Petal.Length	0.872	-0.428	1.000	0.963
Petal.Width	0.818	-0.366	0.963	1.000

What are the limitations of this approach?

Correlation matrix – by factor

Pairwise correlation by species

```
> by(iris[1:4], factor(iris$Species), cor)
```

Using "by" to separate species

```
factor(iris$Species): setosa
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000   0.7425467    0.2671758   0.2780984
Sepal.Width     0.7425467   1.0000000    0.1777000   0.2327520
Petal.Length    0.2671758   0.1777000    1.0000000   0.3316300
Petal.Width     0.2780984   0.2327520    0.3316300   1.0000000
------------------------------------------------------------
factor(iris$Species): versicolor
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000   0.5259107    0.7540490   0.5464611
Sepal.Width     0.5259107   1.0000000    0.5605221   0.6639987
Petal.Length    0.7540490   0.5605221    1.0000000   0.7866681
Petal.Width     0.5464611   0.6639987    0.7866681   1.0000000
------------------------------------------------------------
factor(iris$Species): virginica
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000   0.4572278    0.8642247   0.2811077
Sepal.Width     0.4572278   1.0000000    0.4010446   0.5377280
Petal.Length    0.8642247   0.4010446    1.0000000   0.3221082
Petal.Width     0.2811077   0.5377280    0.3221082   1.0000000
```

Limitations?

Challenge 2. Function: by

'by' enables a function to be applied across individual or multiple columns of a data frame grouped by a factor or factors.

- To calculate the correlation of sepal length and width
  - > by(iris, iris[5], function(df) cor(df$Sepal.Length, df$Sepal.Width))

```
Species: setosa
[1] 0.743
Species: versicolor
[1] 0.526
Species: virginica
[1] 0.457
```

?by

- Description

  Apply a Function to a Data Frame Split by Factors

- Usage

  ```
  by(data, INDICES, FUN, ..., simplify = TRUE)
  ```

- Arguments

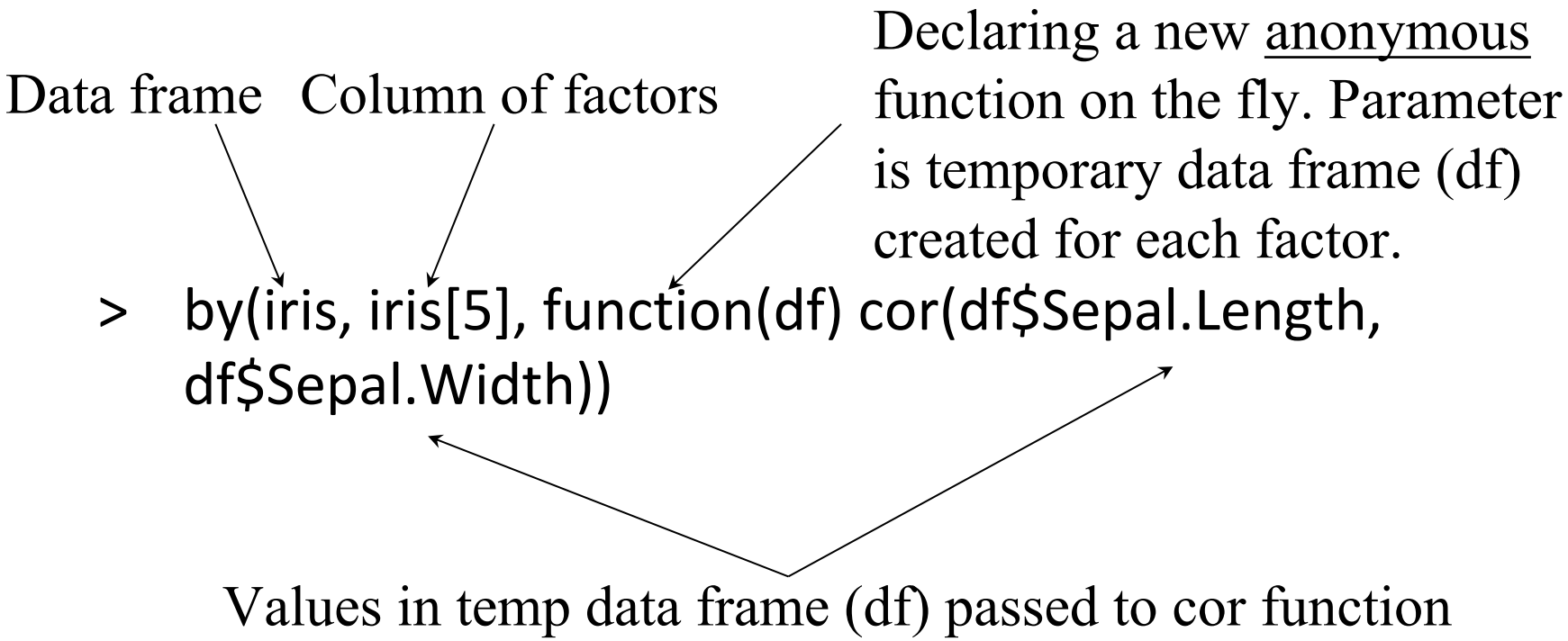
  Data : an R object, normally a data frame, possibly a matrix.

  INDICES : a factor or a list of factors, each of length nrow(data).

  FUN : a function to be applied to data frame subsets of data...

?by: applying the cor function

Looking more closely at the way correlation is calculated:



Anonymous functions

If a function is only to be used once, it can be defined when it is used. These are anonymous functions (having no name) see ATHR p.41.

See previous slide for an example:

```
> by(iris, iris[5], function(df) cor(df$Sepal.Length,
df$Sepal.Width))
```

. . .

Changing earlier example to a more compact notation, using column indexes.

From:

```
> by(iris, iris[5], function(df) cor(df$Sepal.Length, df$Sepal.Width))
```

To:

```
> by(iris, iris[5], function(df) cor(df[1], df[2]))
```

Function: as.table

This function converts the output format of the previous function from a list to a table

```
> as.table(by(iris, iris[5], function(df) cor(df[1], df[2])))
Species
    setosa versicolor  virginica
     0.743      0.526      0.457
```

Function: as.data.frame

This function converts "coerces" the table into a data frame.

Note "Freq" is the generic column name for calculated values in df created this way.

```
> Sepal.cor <- as.data.frame(as.table(by(iris, iris[5], function(df) cor(df[1], df[2]))))
> Sepal.cor
      Species  Freq
1      setosa 0.743
2  versicolor 0.526
3   virginica 0.457
```

Function: colnames

This function assigns new column names to a data frame.

```
> colnames(Sepal.cor) <- c("Species", "Sepal.cor")
> Sepal.cor
     Species Sepal.cor
1     setosa     0.743
2 versicolor     0.526
3  virginica     0.457
```

Now for petals…

Repeating the previous code for petals…

```
> Petal.cor <- as.data.frame(as.table(by(iris, iris[5],
  function(df) cor(df[3], df[4]))))
> colnames(Petal.cor) <- c("Species", "Petal.cor")
> Petal.cor
      Species Petal.cor
1      setosa      0.332
2  versicolor      0.787
3   virginica      0.322
```

Merging data frames (and saving)

Using a common column, “Species” as index and rounding data.

Note: we could have used cbind to combine data frames since they have same structure.

```
> iris.cor <- merge(Sepal.cor, Petal.cor, by = "Species")
> iris.cor[,2] = round(iris.cor[,2], digits = 3)
> iris.cor[,3] = round(iris.cor[,3], digits = 3)
> write.csv(iris.cor, file = "Iris.cor.csv",
  row.names=FALSE)
```

The saved file

SepalPetalcor.csv

Species	Sepal.cor	Petal.cor
setosa	0.743	0.332
versicolor	0.526	0.787
virginica	0.457	0.322

This gives a much more compact presentation of the main correlations compared to the table created previously

```
> by(iris[1:4], factor(iris$Species), cor).
```

Two more challenges

(3) Easy!

- Examine the difference between the aspect ratios (Length / Width) for sepals and petals between the different species.

(4) A little harder!

- Report the data for the flower having the longest petal in each species. We'll use dplyr for this.

Challenge 3: Add/remove columns

- By default, R will add a new column to a data frame if the output of a column operation is specified as a new column.
- This lets us store the results of row operations, including factor generation.
- Alternatively, the cbind function can be used to append a (column) vector to a data frame.

Making new columns

Add two columns containing the aspect ratio (length/width) for sepals and petals:

```
> niris <- iris # creating a new data frame
> niris$Sepal.ar <- niris$Sepal.Length/niris$Sepal.Width
  # add new column
> niris$Petal.ar <- niris$Petal.Length/niris$Petal.Width
  # add new column
> head(niris)
```

The augmented data frame: niris

```
> head(niris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species Sepal.ar Petal.ar
1          5.1         3.5          1.4         0.2  setosa     1.46     7.00
2          4.9         3.0          1.4         0.2  setosa     1.63     7.00
3          4.7         3.2          1.3         0.2  setosa     1.47     6.50
4          4.6         3.1          1.5         0.2  setosa     1.48     7.50
5          5.0         3.6          1.4         0.2  setosa     1.39     7.00
6          5.4         3.9          1.7         0.4  setosa     1.38     4.25
```

Deleting columns

This is easy – but cannot be undone!

To remove a single column, do it by name.

To remove the first column:

```
> niris$Sepal.Length <- NULL
```

Tedious for multiple columns. A quicker but potentially dangerous way to remove first 4 columns:

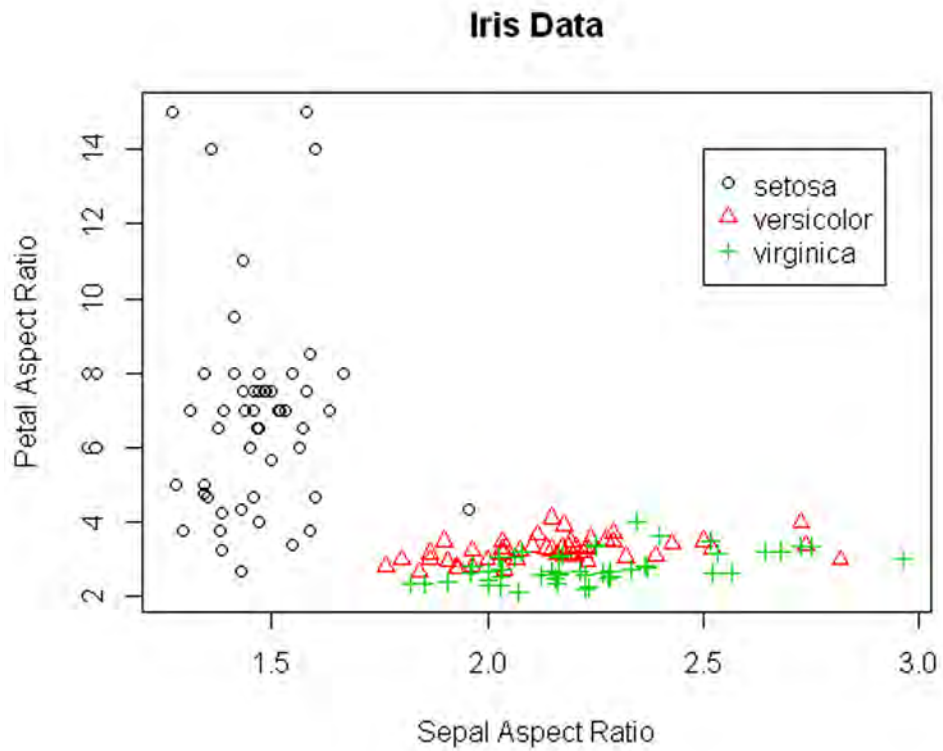
```
> niris <- niris[,c(5:7)] # reassign cols 5:7 on to itself!
```

After removing columns:

```
> head(niris)
  Species Sepal.ar Petal.ar
1  setosa     1.46     7.00
2  setosa     1.63     7.00
3  setosa     1.47     6.50
4  setosa     1.48     7.50
5  setosa     1.39     7.00
6  setosa     1.38     4.25
```

Scatterplot

Petal vs Sepal aspect ratio (Length / Width)



Code for scatterplot on previous slide:

```
> with(niris, plot(Sepal.ar, Petal.ar, col = Species,
  pch=as.numeric(Species), main = ("Iris Data"), xlab =
  "Sepal Aspect Ratio", ylab = ("Petal Aspect Ratio")))
> with(niris, legend(2.5, 14, as.vector(unique(Species)),
  pch=unique(Species), col = unique(Species)))
```

dplyr

If some of the manipulation we've done so far looks a bit intimidating, you might want to try the 'dplyr' package. dplyr:

- is a *Grammar of Data* manipulation,
- provides a consistent set of verbs to simplify the most common data manipulation challenges.
- https://dplyr.tidyverse.org/
- See Chapter 3, Data Transformation in R for Data Science https://r4ds.hadley.nz/

dplyr

dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges:

- `mutate()` adds new variables that are functions of existing variables
- `select()` picks variables based on their names.
- `filter()` picks cases based on their values.
- `summarise()` reduces multiple values down to a single summary.
- `arrange()` changes the ordering of the rows.

These all combine naturally with `group_by()` which allows you to perform any operation "by group". You can learn more about them in `vignette("dplyr")`. As well as these single-table verbs, dplyr also provides a variety of two-table verbs, which you can learn about in `vignette("two-table")`.

From: https://dplyr.tidyverse.org/

dplyr

Quick start:

- Use pipes, %>%, to connect data to a grouping variable, and then apply a function.
- For example, to find average sepal length by species::

```
> iris %>% group_by(Species) %>%
  summarise(Ave.Sepal.len = mean(Sepal.Length))
```

```
# A tibble: 3 × 2
  Species    Ave.Sepal.len
  <fct>              <dbl>
1 setosa              5.01
2 versicolor          5.94
3 virginica           6.59
```

dplyr

Tibbles…

- Dplyr creates tibbles instead of data frames. To get an overview of the difference between these, see:
- https://r4ds.had.co.nz/tibbles.html

- You can convert a tibble to a data frame if preferred using:

```
> NewDataFrame = as.data.frame(TibbleName)
```

dplyr (Challenge 1)

- For Challenge 1, find column means by species:
  - > iris %>% group_by(Species) %>% summarise(ASL = mean(Sepal.Length), ASW = mean(Sepal.Width), APL = mean(Petal.Length), APW = mean(Petal.Width))

```
# A tibble: 3 × 5
  Species      ASL   ASW   APL   APW
  <fct>      <dbl> <dbl> <dbl> <dbl>
1 setosa      5.01  3.43  1.46 0.246
2 versicolor  5.94  2.77  4.26 1.33
3 virginica   6.59  2.97  5.55 2.03
```

dplyr (Challenge 2)

- For Challenge 2, find the correlation between sepal length and width, and petal length and width by species – first step is shown below:

```
> iris %>% group_by(Species) %>% summarise(Sepal.cor
  = cor(Sepal.Length, Sepal.Width))
```

```
# A tibble: 3 × 2
  Species    Sepal.cor
  <fct>          <dbl>
1 setosa         0.743
2 versicolor     0.526
3 virginica      0.457
```

Challenge 4: using dplyr

This task shows off how useful dplyr is:

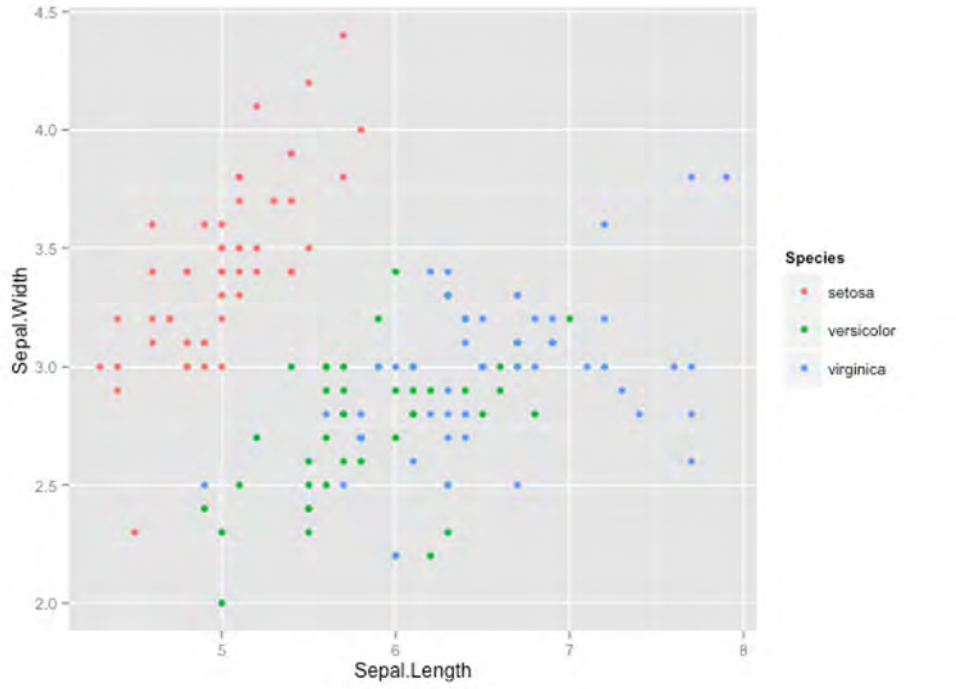
To find the flower having the longest petal in each species:

```
>   iris %>% group_by(Species) %>% top_n(1, Petal.Length)
# A tibble: 4 × 5
# Groups:   Species [3]
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
         <dbl>       <dbl>        <dbl>       <dbl> <fct>
1          4.8         3.4          1.9         0.2 setosa
2          5.1         3.8          1.9         0.4 setosa
3          6           2.7          5.1         1.6 versicolor
4          7.7         2.6          6.9         2.3 virginica
```

Results show two I.setosa flowers having equally long petals.

Challenge 5: recoding and indexing

Does Iris setosa have an average sepal width greater than I.versicolor and virginica as a group?



Challenge 5: recoding

To compare I.setosa against the other two species combined, we need to create a new index as a column that groups I.versicolor and virginica.

- Note: use the function “recode” from the “car” package
  - > niris = iris # clone iris data
  - > install.packages("car")
  - > library(car)

Challenge 5: create new factor column

```
> ...
> niris$vvs = recode(niris$Species," 'versicolor' =
  '0';'virginica' = '0';'setosa' = '1' ")
> print(niris[c(1,51,101),]) # as a check
```

```
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species vvs
1            5.1         3.5          1.4         0.2     setosa   1
51           7.0         3.2          4.7         1.4 versicolor   0
101          6.3         3.3          6.0         2.5  virginica   0
```

Challenge 5: t-Test

```
> ...
> t.test(niris$Sepal.Width~niris$vvs, alternative = "less")
```

```
    Welch Two Sample t-test
data:  niris$Sepal.Width by niris$vvs
t = -8.8121, df = 87.596, p-value = 5.177e-14
alternative hypothesis: true difference in means between group 0 and group 1
    is less than 0
95 percent confidence interval:
      -Inf -0.451108
sample estimates:
mean in group 0 mean in group 1
          2.872           3.428
```

Challenge 5: Data frames as subsets

Alternatively, we could have made two new data frames from the original iris data, one for I.setosa, and one combining I.versicolor and I.virginica.

- Note: use of logical operators "==" (is equal to), and "%in%" (is contained in)…

```
> iris.set = iris[iris$Species == "setosa",]
> iris.ver.vir = iris[(iris$Species %in%
  c("virginica","versicolor")),]
```

Challenge 5: Data frames as subsets

> t.test(iris.ver.vir$Sepal.Width, iris.set$Sepal.Width, alternative = "less")

```
Welch Two Sample t-test
data:  iris.ver.vir$Sepal.Width and iris.set$Sepal.Width
t = -8.8121, df = 87.596, p-value = 5.177e-14
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval: -Inf -0.451108
sample estimates:
mean of x mean of y
    2.872     3.428
```

Compact graphics

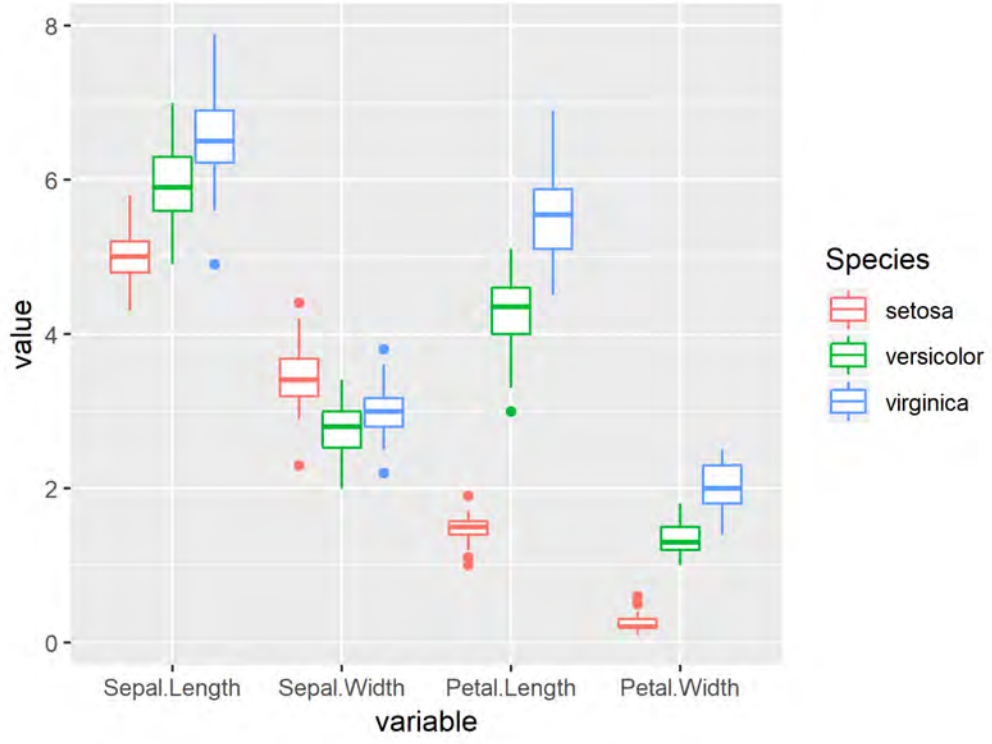
Three examples of compact graphics to display attributes of multiple variables follow:

- Side-by-side boxplots
- Heatmaps
- Correlation Matrix

See data formatting examples necessary to make some these plots following…

Side-by-side boxplots

- Using ggplot2.
- Factor levels of each variable shown side-by-side.
- Data needs to be in a "long" format – see following slides.



https://statisticsglobe.com/draw-multiple-boxplots-in-one-graph-in-r

Long format

	Species	variable	value
143	virginica	Sepal.Length	5.8
144	virginica	Sepal.Length	6.8
145	virginica	Sepal.Length	6.7
146	virginica	Sepal.Length	6.7
147	virginica	Sepal.Length	6.3
148	virginica	Sepal.Length	6.5
149	virginica	Sepal.Length	6.2
150	virginica	Sepal.Length	5.9
151	setosa	Sepal.Width	3.5
152	setosa	Sepal.Width	3.0
153	setosa	Sepal.Width	3.2
154	setosa	Sepal.Width	3.1
155	setosa	Sepal.Width	3.6
156	setosa	Sepal.Width	3.9
157	setosa	Sepal.Width	3.4

Showing 143 to 157 of 600 entries, 3 total columns

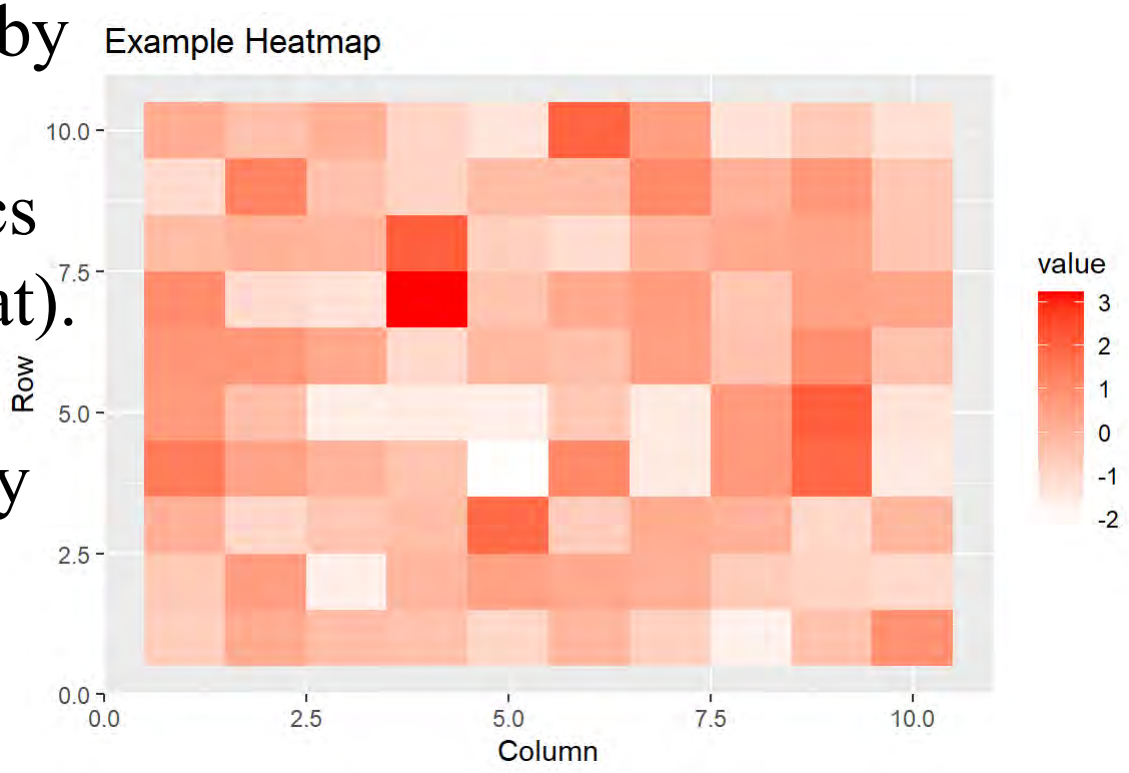
Side-by-side boxplots

Using code adapted from link on previous page.

```
> rm(list = ls())
> library(ggplot2)
> library("reshape2")
> iris_long <- melt(iris, id = "Species")
> ggplot(iris_long, aes(x = variable, y = value, color =
  Species)) + geom_boxplot()
> ggsave("Iris Boxplot by Species.pdf", width = 15,
  height = 15, units = "cm")
```

Heatmap

- Values of a variable by two (x,y) factors.
- Can use base graphics (data in matrix format).
- Or ggplot2 (data in "long" format). Many other packages too. This example uses plotly.



https://rpubs.com/lumumba99/1026665

Heatmap

Formatting the iris data

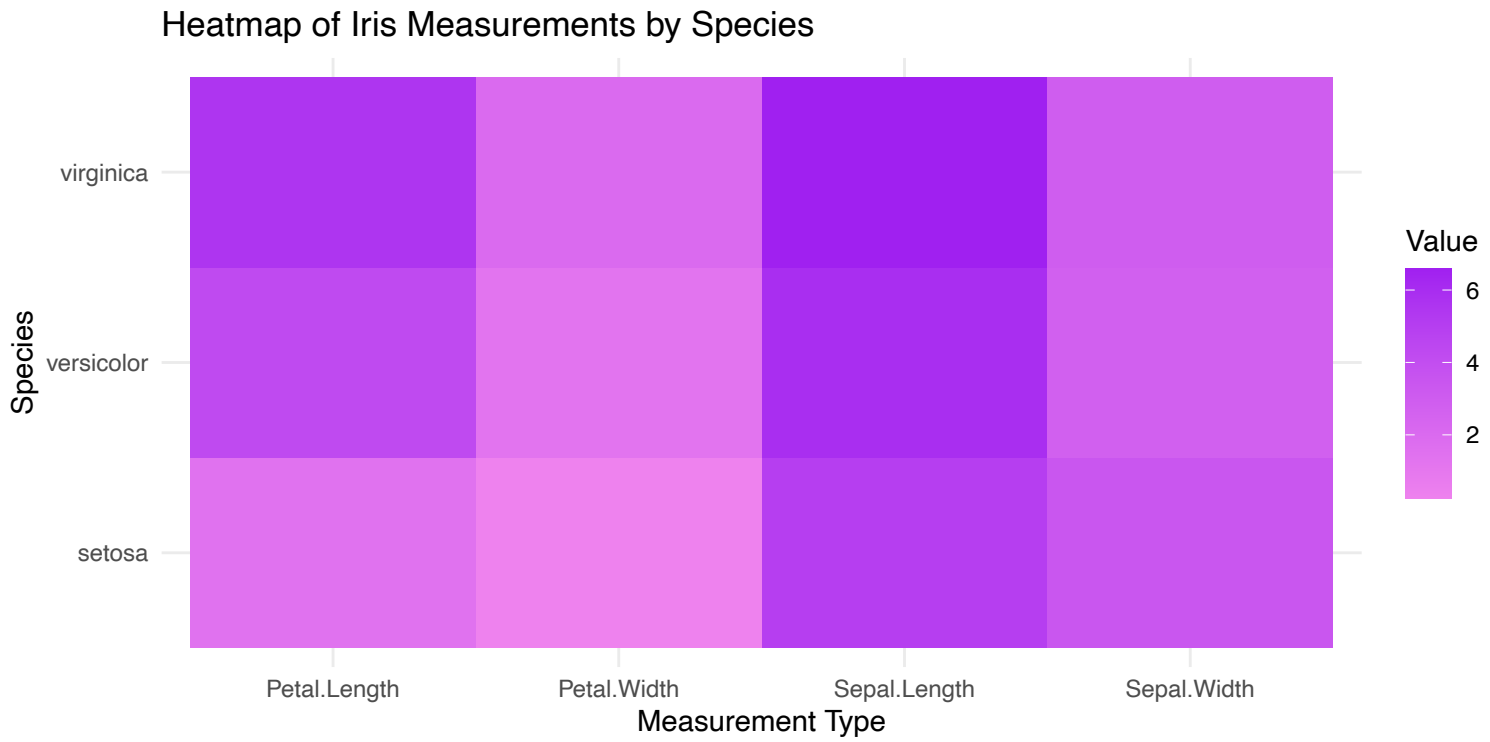
```
> rm(list = ls())
> library(dplyr)
> library(tidyr)
> iris_summary <- iris %>% group_by(Species) %>%
  summarise_all(mean)
> iris_long_avg <- iris_summary %>% pivot_longer(cols
  = -Species, names_to = "Measurement", values_to =
  "Average_Value")
```

Heatmap

Making the plot

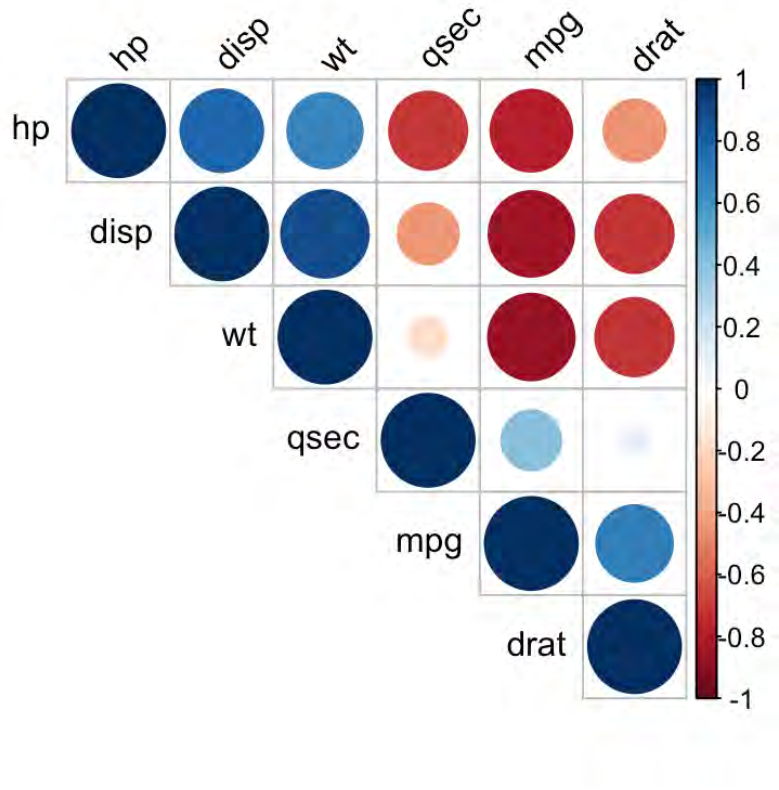
```
> g = ggplot(iris_long_avg, aes(x = Measurement, y = Species, fill = Average_Value))
> g = g + geom_tile()  + scale_fill_gradient(low = "violet", high = "purple")
> g = g + labs(title = "Heatmap of Iris Measurements by Species", x = "Measurement Type", y = "Species", fill = "Value") + theme_minimal()
> g
> ggsave("Iris Heatmap.pdf", g, width = 20, height = 10, units = "cm")
```

Heatmap


Heatmap of Iris Measurements by Species
Species
virginica
versicolor
setosa
Petal.Length
Petal.Width
Sepal.Length
Sepal.Width
Measurement Type
Value
6
4
2

Correlation matrix

- Displays pair-wise correlation for several variables.
- Calculate correlation first. This plot uses corrplot package.
- Colour and size are both based on correlation coefficient in this example.
- You can investigate further…



https://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide...

Brief summary

Summarising data by factors using:

- Base functions: “aggregate”, “by”
- dplyr package functions: “group_by”, “summarise”.

Creating and removing columns

Searching, indexing and combining rows

- dplyr functions: “group_by”, “top_n”.
- Base functions: “as.table”, “as.data.frame”, “colnames”, adding and removing columns.
- Logical operators “==” and “%in%”.

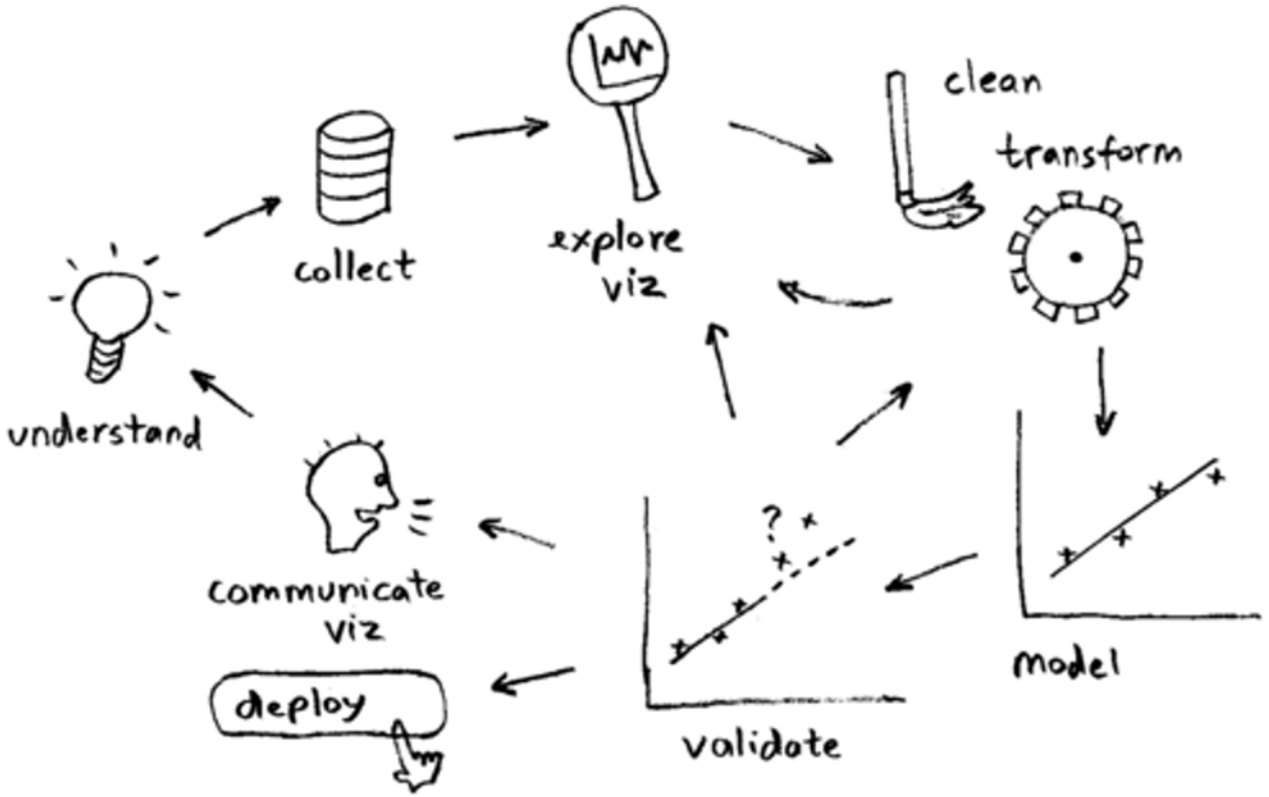
Exploring your data

Outline

In this section we'll cover:

- Some graphical methods for starting your analysis for Assignment 1. This approach is called:
- Exploratory Data Analysis (EDA)
- Almost all material in this lecture is from *R for Data Science*, 2nd Edition, https://r4ds.hadley.nz/eda Chapter 10.
- You can read through this chapter online.

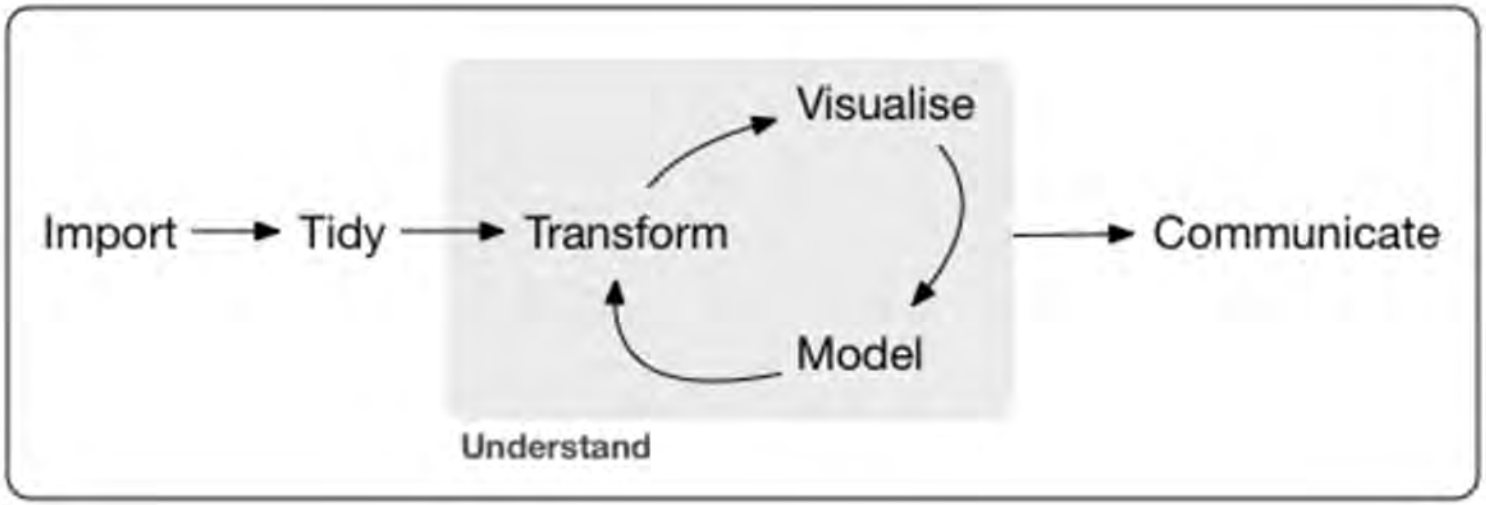
The data science workflow



datascience.la/

The data science workflow

From: R for Data Science



Think about the relevance of this slide and how you might tackle Assignment 1.

https://r4ds.hadley.nz/

Getting started!

Starting any data analysis project is challenging.

- Following the workflows in the previous slides is one effective way to start.
- Think of it as Exploratory Data Analysis, EDA.
- It is a cycle of visualising, transforming and modelling, to develop and refine questions about your data.
- Your first graphs or models might not tell you much about the data. Use them to refine your next attempt. Repeat the process until you have a story!

Graphics for exploratory data analysis

- Each slide following presents one visualisation.
- They are taken from R for Data Science 2nd Ed.,
- Uses the Diamond data (introduced in App Sess 2).
- The main insight from each will be discussed.
- You should follow up on those that are useful.
- The examples in the text use the "tidyverse," but can be adapted to your preferred coding method.
- This presentation doesn't cover all methods you might use for your assignment.

Inspiration

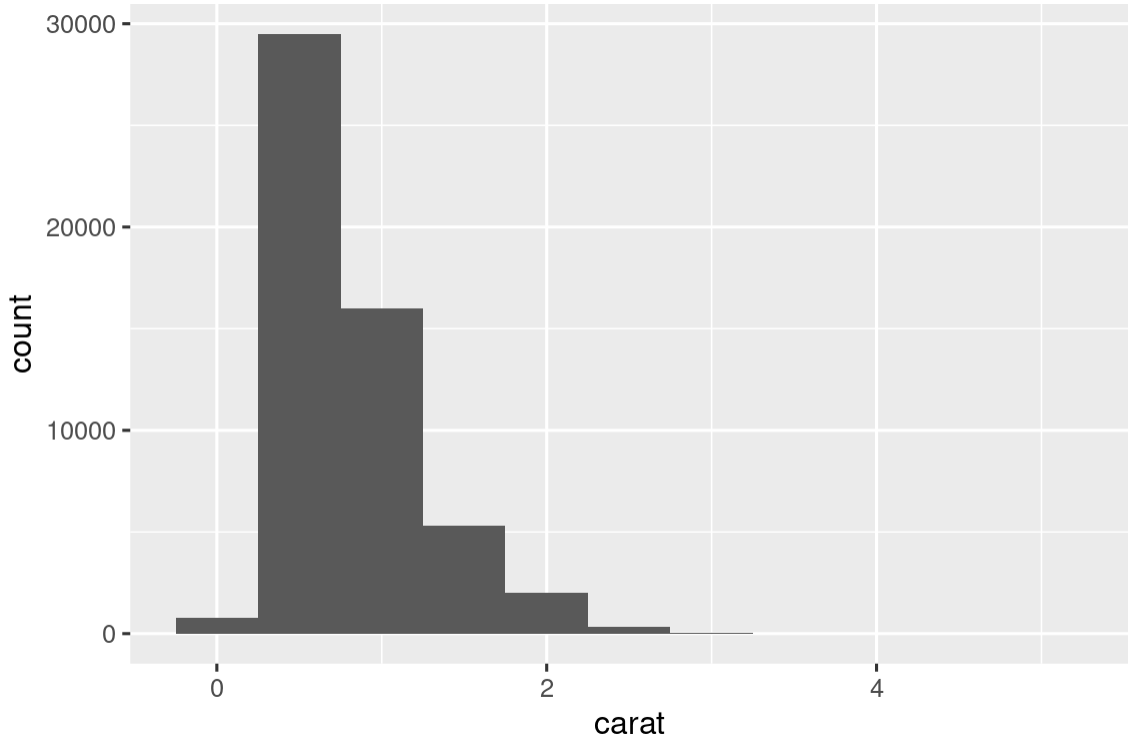
10.2 Comments about questions:

"There are no routine statistical questions, only questionable statistical routines." — Sir David Cox

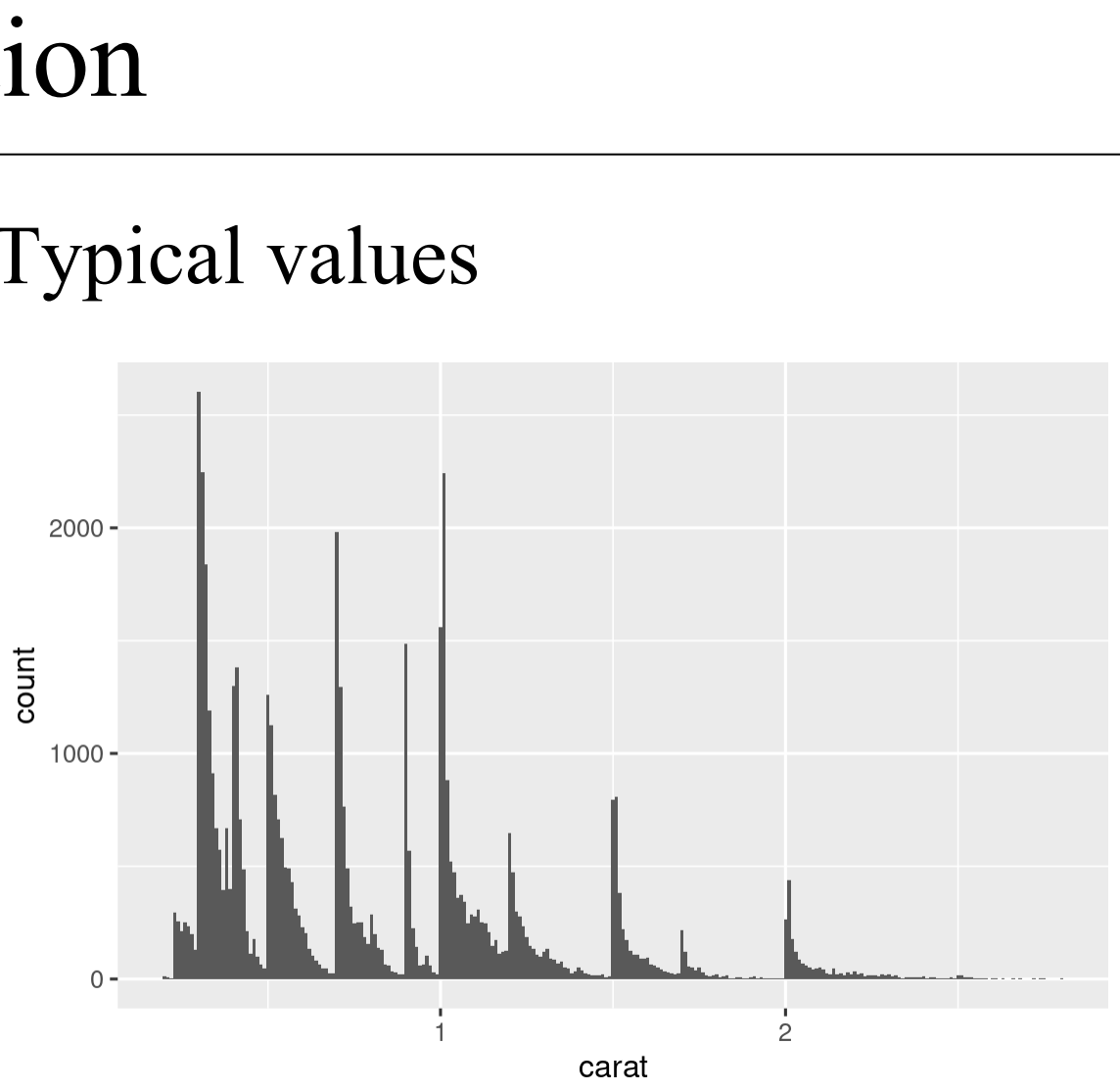
"Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise." — John Tukey

Variation

10.3 Variation in a single variable



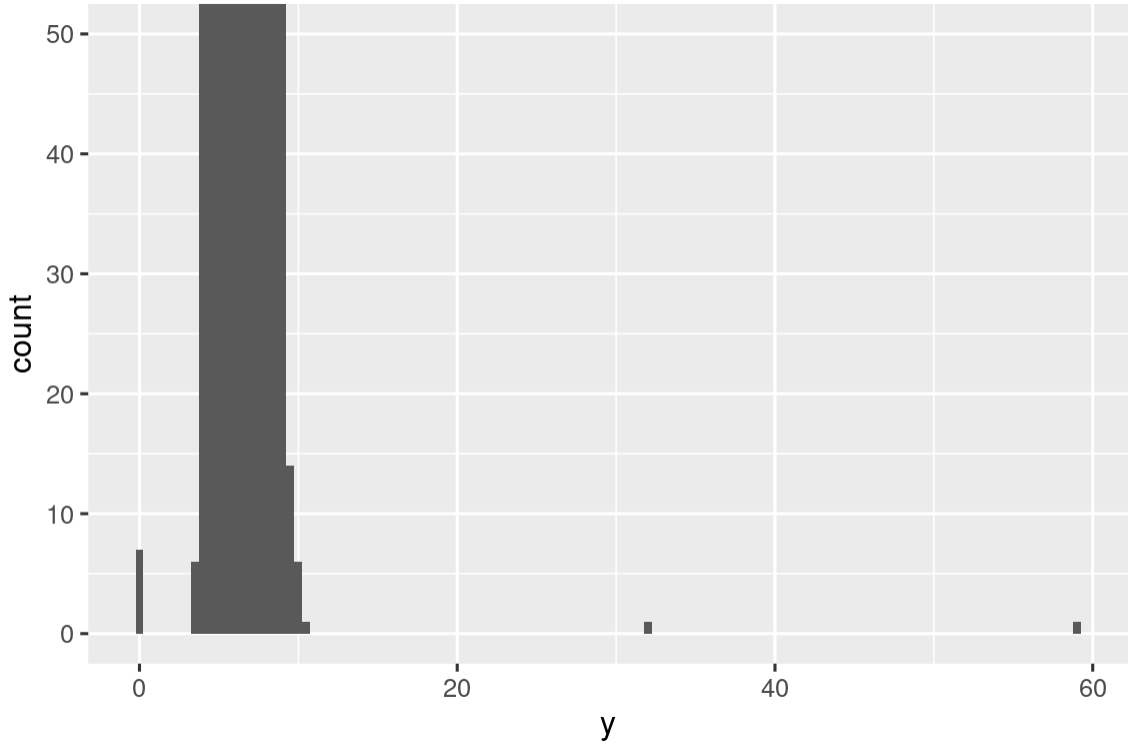
Variation



10.3.1 Typical values

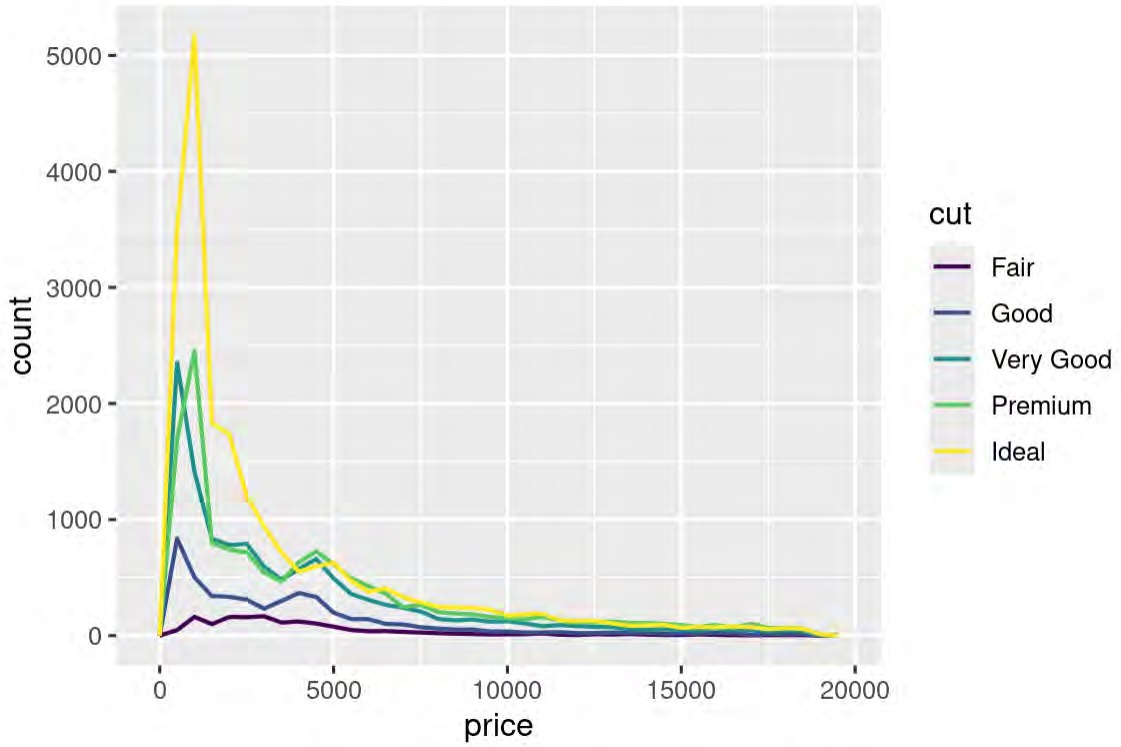
Variation

10.3.2 Unusual values



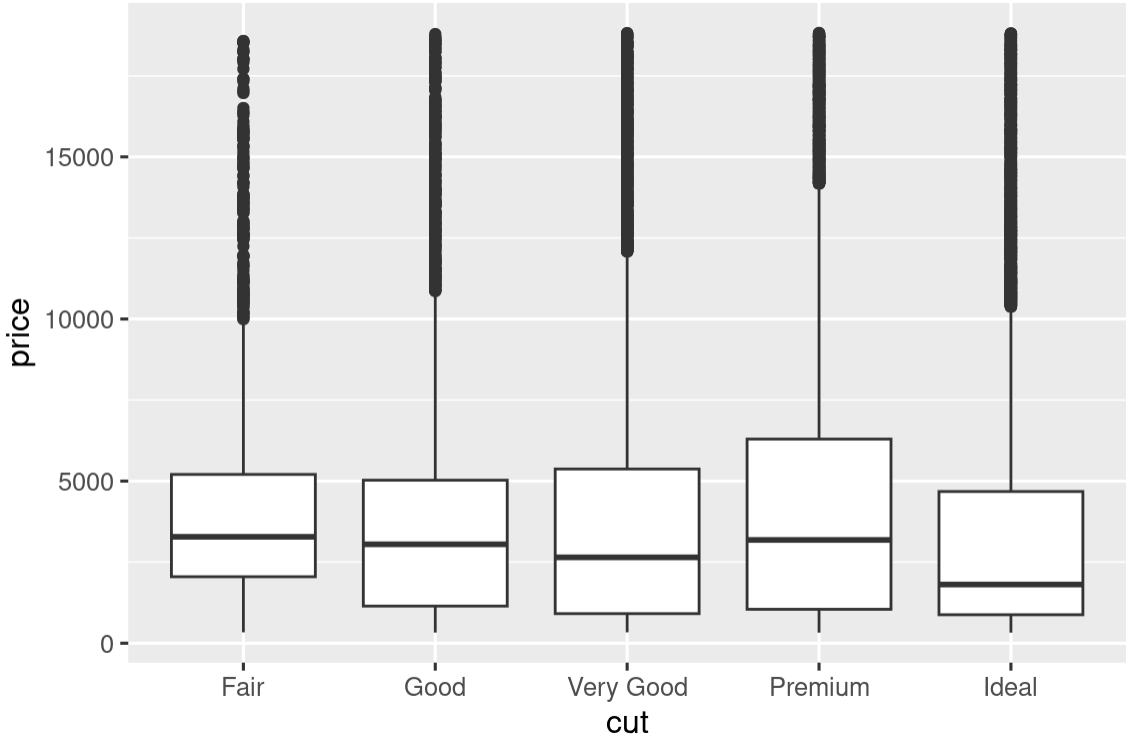
Covariation

10.5.1 Categorical and numerical



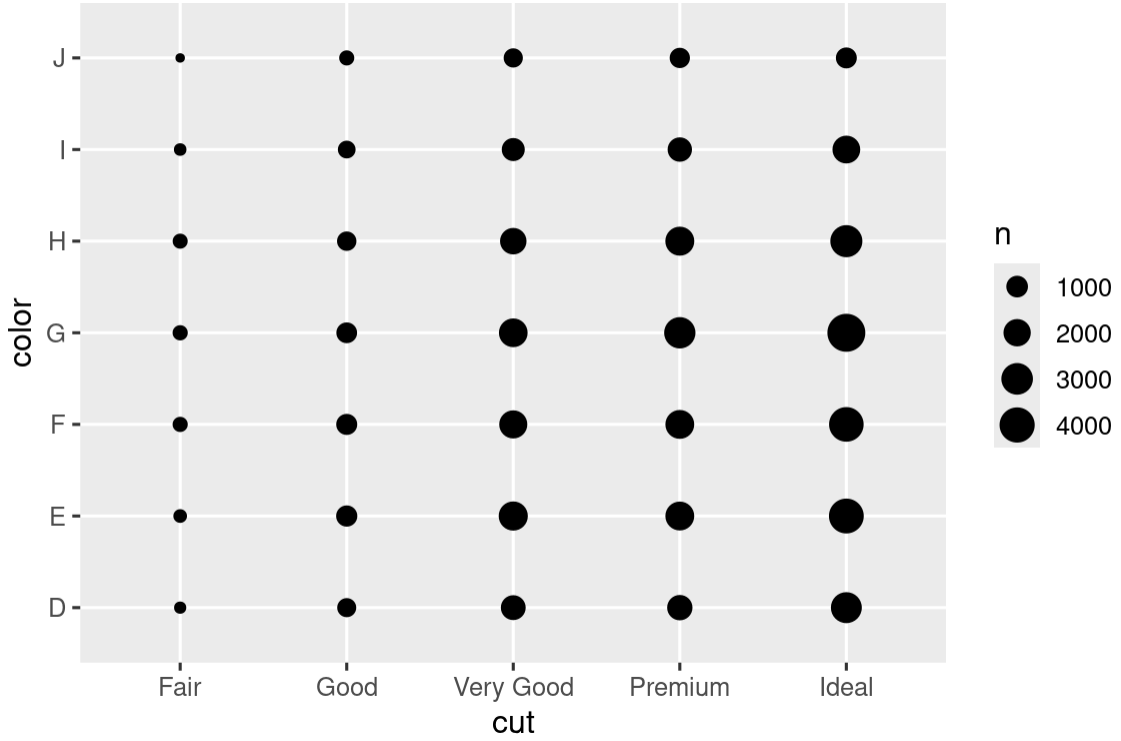
Covariation

10.5.1 Categorical and numerical



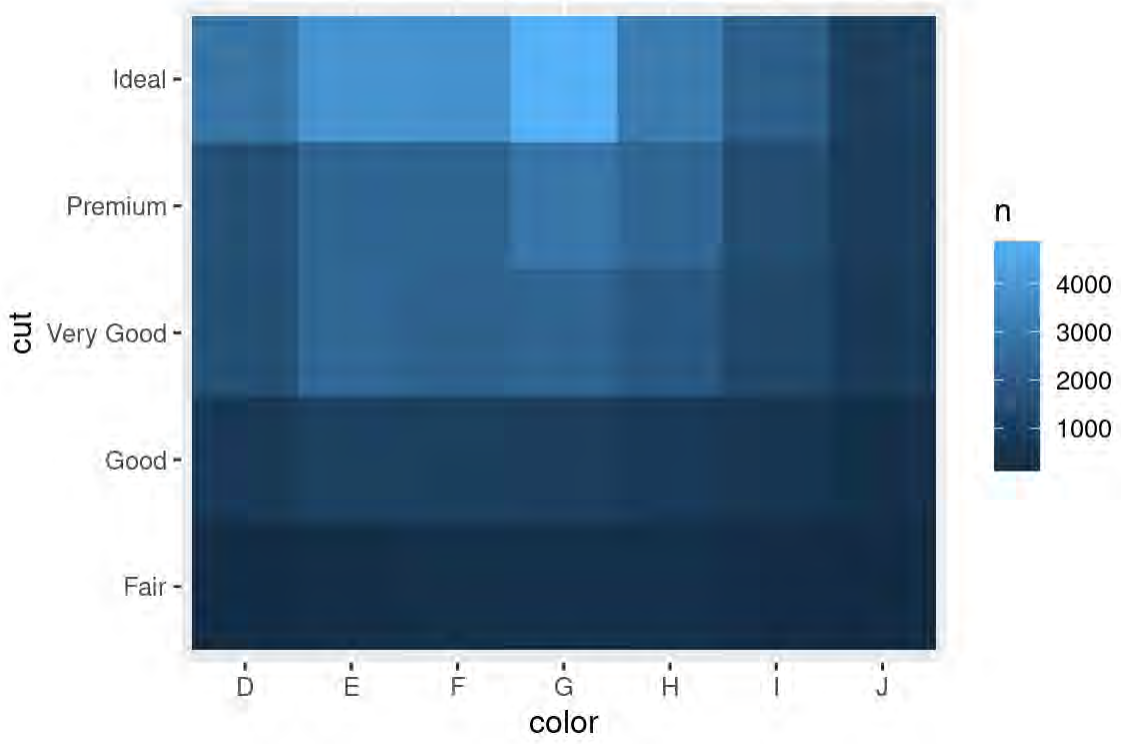
Covariation

10.5.2 Two categorical variables



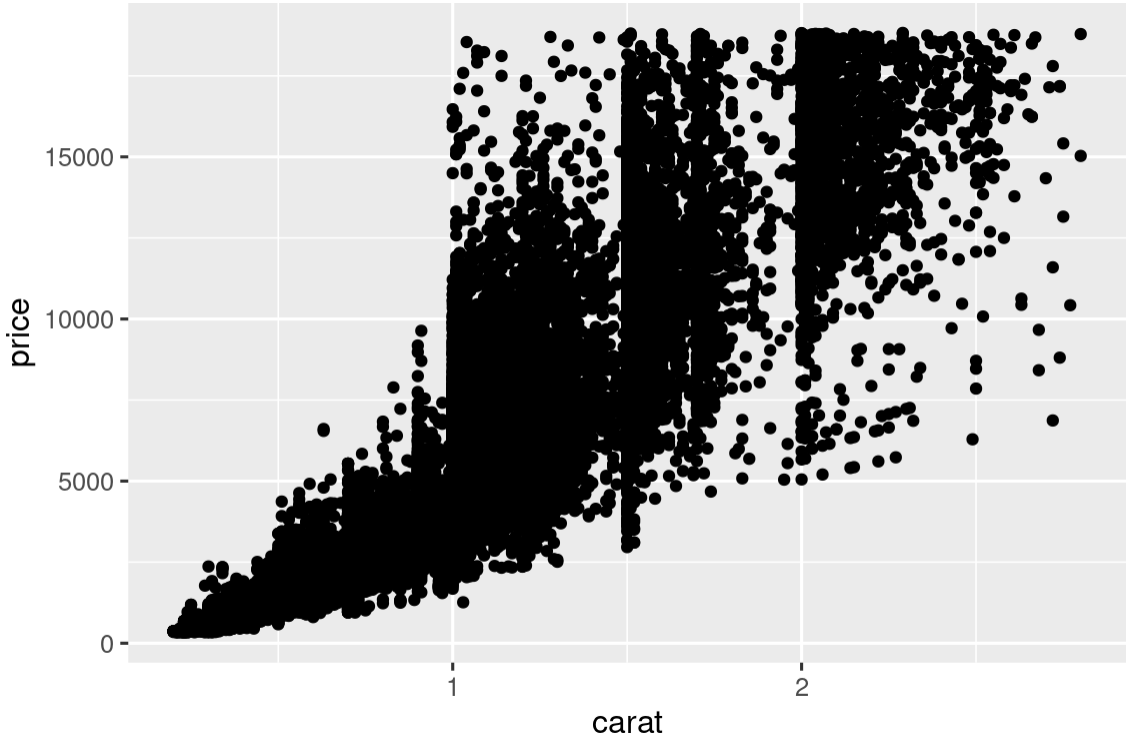
Covariation

10.5.2 Two categorical variables



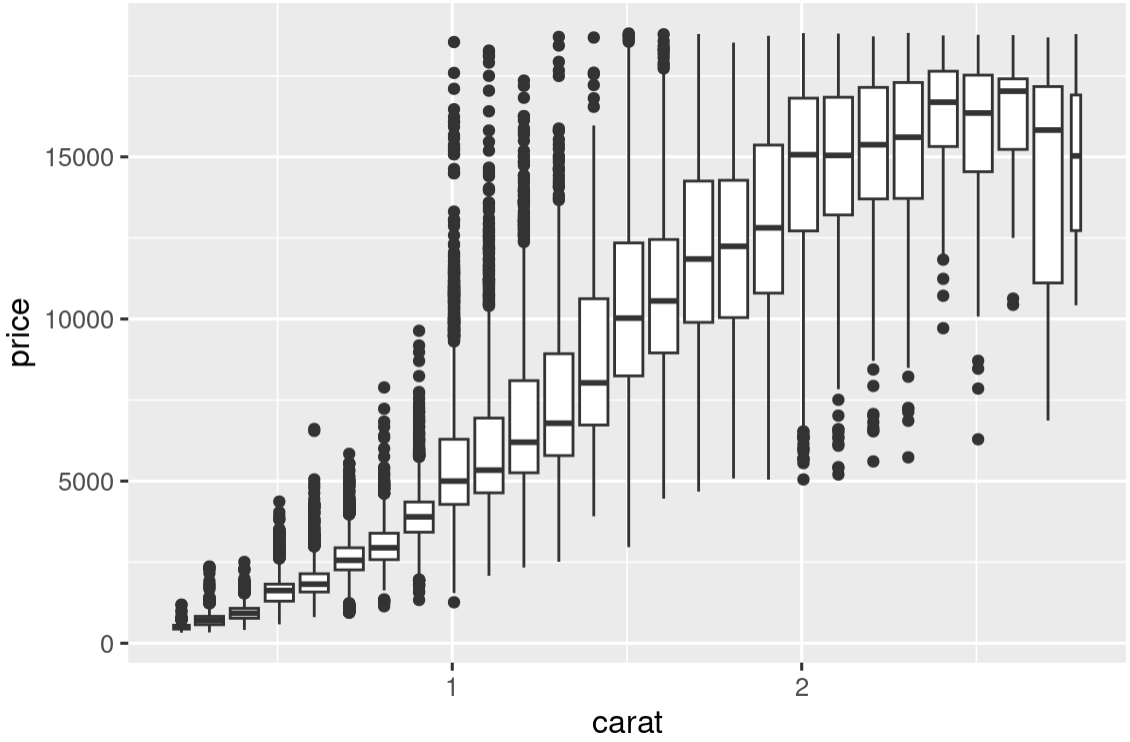
Covariation

10.5.3 Two numerical variables



Covariation

10.5.3 Two numerical variables



Binning carats at 0.1 and treating as a factor

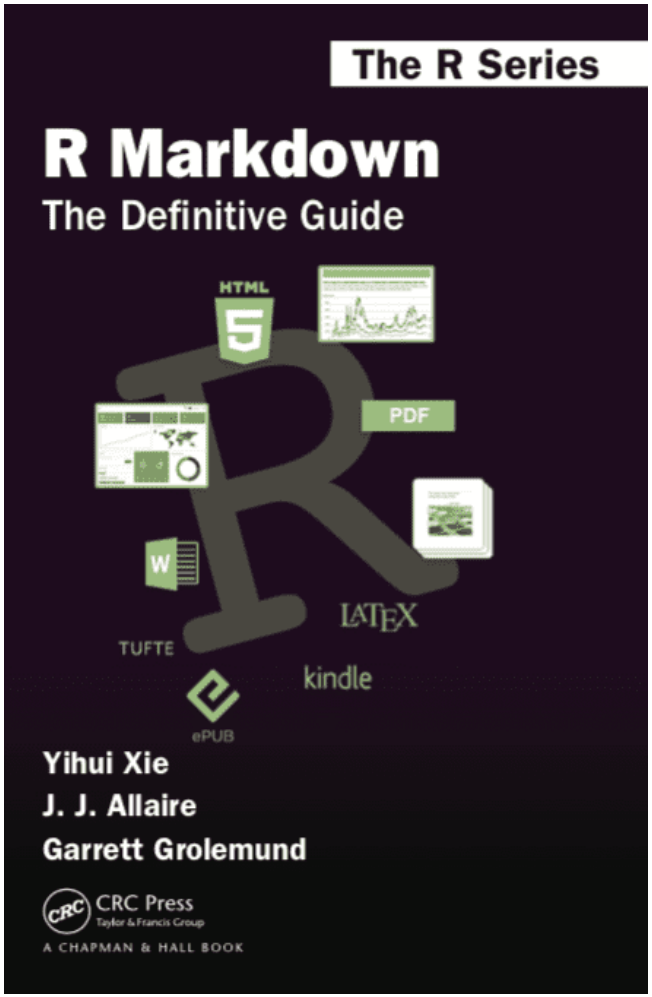
Closing comments

- There's no one "right" way to start your assignment.
- Making plots is a good way to start.
- We've covered data manipulation and making summary tables in Week 3.
- We also cover modelling data with correlation and regression in Week 4.
- All these methods may be useful for your assignment.

References

Online:

- R Markdown: the definitive guide (free online)
- https://bookdown.org/yihui/rmarkdown/
- Notebooks, see Section 3.2
- Functions: (From Quick-R)
- https://www.statmethods.net/management/userfunctions.html
- *R for Data Science*, 2nd Edition, https://r4ds.hadley.nz/



References

Books – online from the Monash Library

- Spector, P., Data manipulation with R.
- Wickham, H., ggplot2: elegant graphics for data analysis.

dplyr Cheat Sheet https://github.com/rstudio/cheatsheets

R Reference card (Tom Short) available from contributed documentation on CRAN site.

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