ETC5513: Assignment Solution

Assignment 1 instructions

In this assignment, we are going to look at the data of COVID-19 cases recorded in China outside Hubei province and on a few other countries. The data set contains information on patient’s country of origin, gender, and age among others as you will see. The data has been obtained from a public data base reported in the Lancet (https://www.thelancet.com/coronavirus). You will:

- Use Rmarkdown and Knitr R packages for reproducible reporting
- Familiarise yourself with markdown syntax
- Continue learning R coding skills
- Acquire practice with the tidyverse *package*, in particular, the packages *dplyr* and *ggplot2*
- Create a reproducible html report using Rmarkdown

Marking rubrick: The marking scheme for this assignment is displayed in the script below and will be used to mark your individual assignments. In addition, **it is essential that the report you submit can be knitted into an html report with the R code chunks set to eval = TRUE** (otherwise you will receive 0 marks, regardless of whether the individual R code chunks run). In this assignment you simply need to feel the gaps marked with —. The R code and the R code ouput must be visible in the knitted report. For this assignment, you will need to upload the following into Moodle:

- Your Rmd file,
- Your html file, and
- A PDF copy of your html file (you can do that by simply opening the html with a browser and printing it to PDF)

Loading libraries

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.3
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
## Warning: package 'dplyr' was built under R version 4.0.2
```

```
library(readr)
library(kableExtra)
library(ggplot2)
library(lubridate)
library(gridExtra)
library(rmdformats)
```

```
## Warning: package 'rmdformats' was built under R version 4.0.4
```

```
# Reading data (you do not need to modify this file)
dat <- read_csv("Data/COVID19_2020_outsideHubei_23March.csv")

dat <- dat %>%
  mutate(sex = ifelse(sex == "female", "Female", sex),
         sex = ifelse(sex == "male", "Male", sex))
```

Question 1: How many variables and observations has this data set?

There are different ways to find out the dimension (number of rows and columns) of a data set and below are a few options that you can use:

```
dim(dat) # 1pt
```

```
## [1] 26033    13
```

Question 2: Inline R code

Using inline R code, complete the sentence where you report the number of rows and columns in the data set.

The data set has 26033 rows and 13 variables.

Question 3: Report the data set variable names in a table

Ensure that the table is captioned "These are the variables included in the COVID-19 data set". Make sure that you choose 2 kable_styling() options.

```
names(dat) %>% # 1pt
  kable(caption="These are the variables  included in the COVID-19 data set") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))  # 1pt
```

These are the variables included in the COVID-19 data set

x
ID
age
sex
city
province
country
latitude
longitude
geo_resolution
date_onset_symptoms
date_admission_hospital
date_confirmation
symptoms

Question 4: Data Wrangling: Practising with *dplyr*

Create a new data set that contains only the following variables: country, age, sex, city, province, latitude, longitude, and display the first 5 rows.

```
dat2 <- dat %>%
  dplyr::select(country,    # 1pt
                age,
                sex,
                city,
                province,
                latitude,
                longitude)

head(dat2, 5) # 1pt
```

country <chr>	… <chr>	sex <chr>	city <chr>	province <chr>	latitude <dbl>	longitude <dbl>
China	30	Male	Chaohu City, Hefei City	Anhui	31.64696	117.7166
China	47	Male	Baohe District, Hefei City	Anhui	31.77863	117.3319
China	49	Male	High-Tech Zone, Hefei City	Anhui	31.82831	117.2248

country	…	sex	city	province	latitude	longitude
<chr>	<chr>	<chr>	<chr>	<chr>	<dbl>	<dbl>
China	47	Female	High-Tech Zone, Hefei City	Anhui	31.82831	117.2248
China	50	Female	Feidong County, Hefei City	Anhui	32.00123	117.5681

5 rows

Question 5: Data variable definitions

Inspect your data set in Question 4 and describe on a list (using markdown syntax) the type of variables (character, numeric, factor, etc.) in the data set and print the name of the variables in bold text.

- country is a character variable.
- age is a character variable.
- sex is a character variable
- city is a character variable.
- province is a character
- latitude is a numeric variable.
- longitude is a numeric variable.

Question 6: Data wrangling: Change variable attributes

Make sure the variables latitude, longitude, and age are defined as numeric variables in your data set dat2. Do not create a new data set but instead modify dat2 to accommodate the changes. Display the first 3 rows of the data set dat2.

```
dat2 <- dat2 %>%
  mutate(latitude = as.numeric(latitude), # 1pt
         longitude = as.numeric(longitude), # 1pt
         age = as.numeric(age)) # 1pt

head(dat2,3)  # 1pt
```

country	…	sex	city	province	latitude	longitude
<chr>	<dbl>	<chr>	<chr>	<chr>	<dbl>	<dbl>
China	30	Male	Chaohu City, Hefei City	Anhui	31.64696	117.7166
China	47	Male	Baohe District, Hefei City	Anhui	31.77863	117.3319
China	49	Male	High-Tech Zone, Hefei City	Anhui	31.82831	117.2248

3 rows

Question 7: Cleaning up data set

Remove the cases of which we do not have information on the patient's age and keep those of which the gender of the patient is known. Name this newly created data set as dat3.

```
dat3 <- dat2 %>% dplyr::filter(!is.na(age),  # 1pt
                               sex %in% c("Female",   # 1pt
                                          "Male") # 1pt
                               )
```

Question 8: Remove patient entries with an age below 1 and name the new data set as dat4

```
dat4 <- dat3 %>%
  dplyr::filter(age >= 1) # 1pt
```

Question 9: Summarise in a table the variable age in dat4 using kable()

Using inline R, write a sentence describing the age of the oldest patient in this data set.

```
dat4 %>% dplyr::select(age) %>% # 1pt
  summary() %>% # Nothing to add here
  kable() %>% # 1pt
  kable_styling(bootstrap_options = c("striped", "hover")) # 1pt
```

	age
	Min. : 1.00
	1st Qu.:32.00
	Median :45.00
	Mean :45.33
	3rd Qu.:58.00
	Max. :96.00

The oldest patient in this data set is 96 (2pts) years old.

Looking at individual country names (nothing to complete here. I just completed for you!)

Count the number of cases per country, arrange the countries in decreasing order of cases, and display a table using kable() of the top 5 countries. Store the results into an object called dat5.

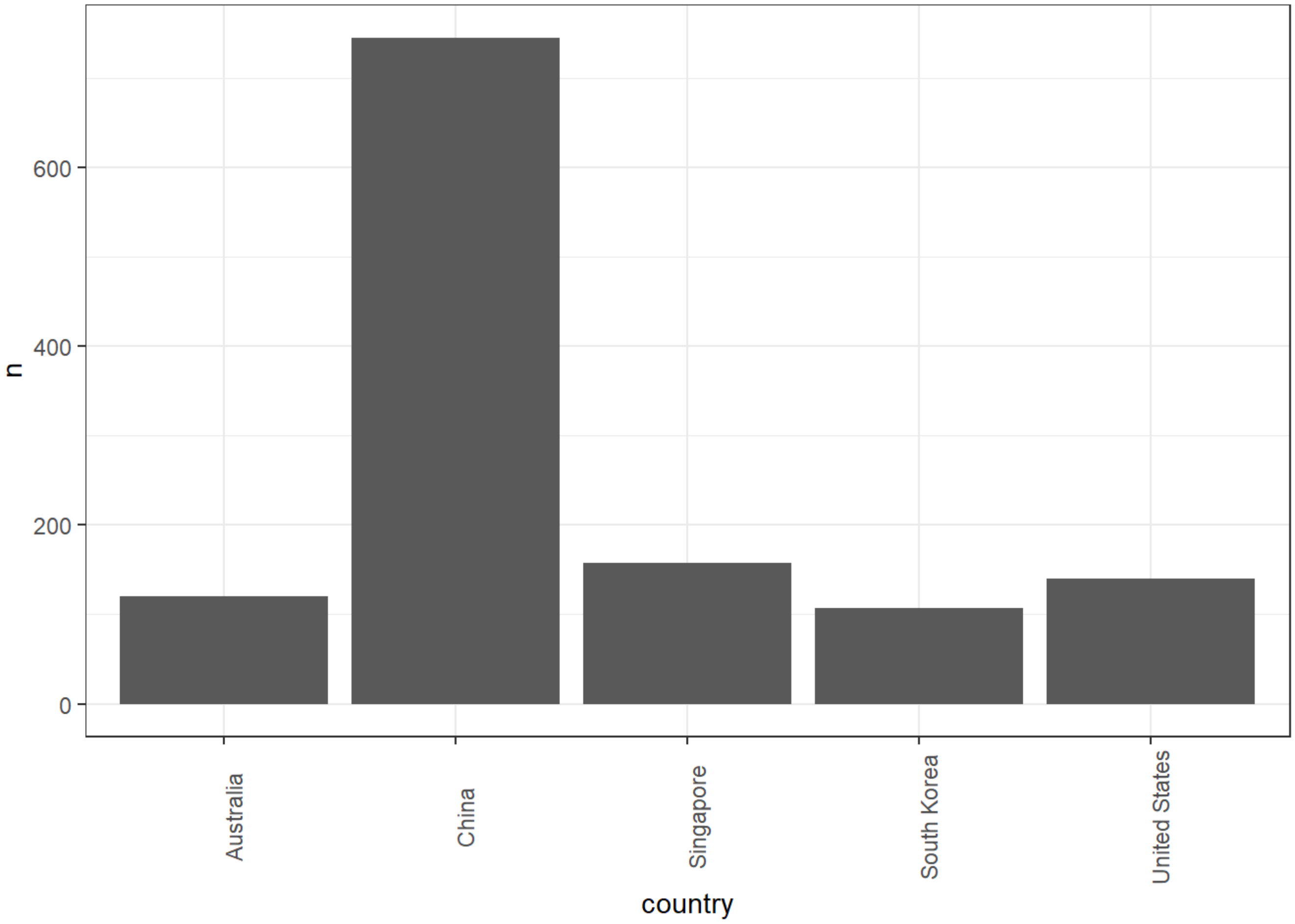
```
# Please turn eval = TRUE above when you have completed Questions 1-9
dat5 <- dat4 %>%
  dplyr::select(country) %>%
  dplyr::filter(!is.na(country)) %>%
  group_by(country) %>%
  mutate(n = n()) %>%
  unique() %>%
  arrange(-n)
kable(dat5[1:5,])
```

country	n
China	745
Singapore	158
United States	140
Australia	120
South Korea	107

Question 10: Looking at the data using different plots

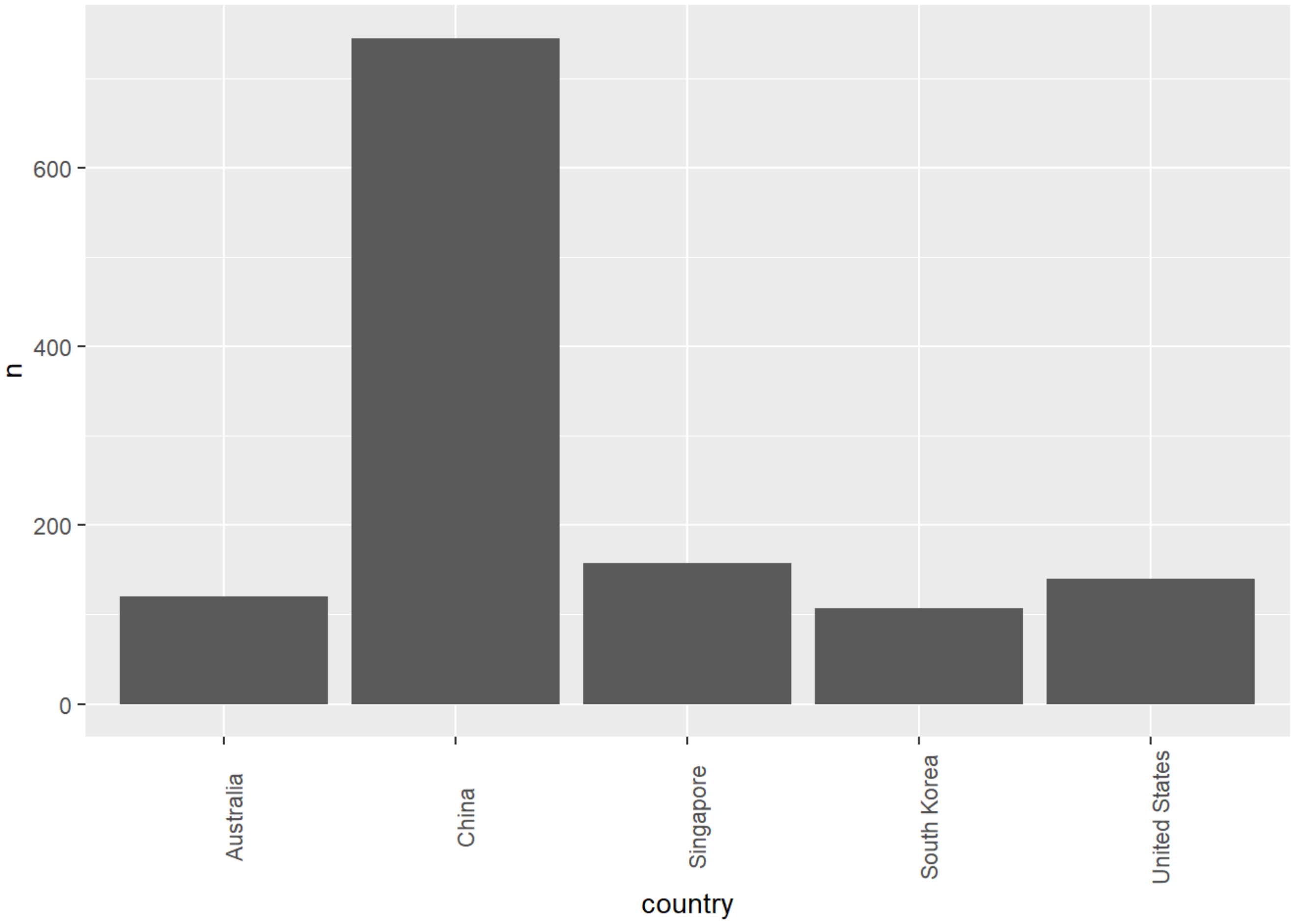
Use geom_point to plot the top 5 countries with the most cases using dat5. Store the plot in a variable called p1. Ensure that the plot is displayed in this section too. Also, make sure you output the plot in this section.

```
p1 = ggplot(dat5[1:5,], aes(x = country, y = n)) + # 1pt
  geom_col() + # 1pt
  theme_bw() + # Nothing to add here
  theme(axis.text.x = element_text(angle = 90)) # Nothing to add here
p1
```



Now repeat the same plot but without the command “theme_bw()”

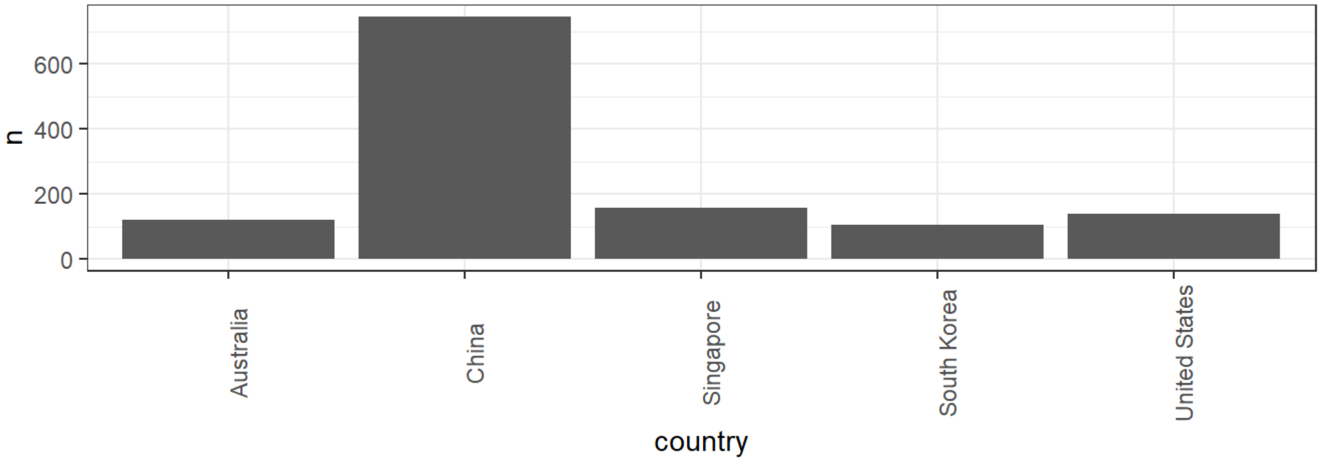
```
# No new code here so no new points assigned
p2 = ggplot(dat5[1:5,], aes(x = country,  y = n)) +
  geom_col() +
  theme(axis.text.x = element_text(angle = 90))
p2
```

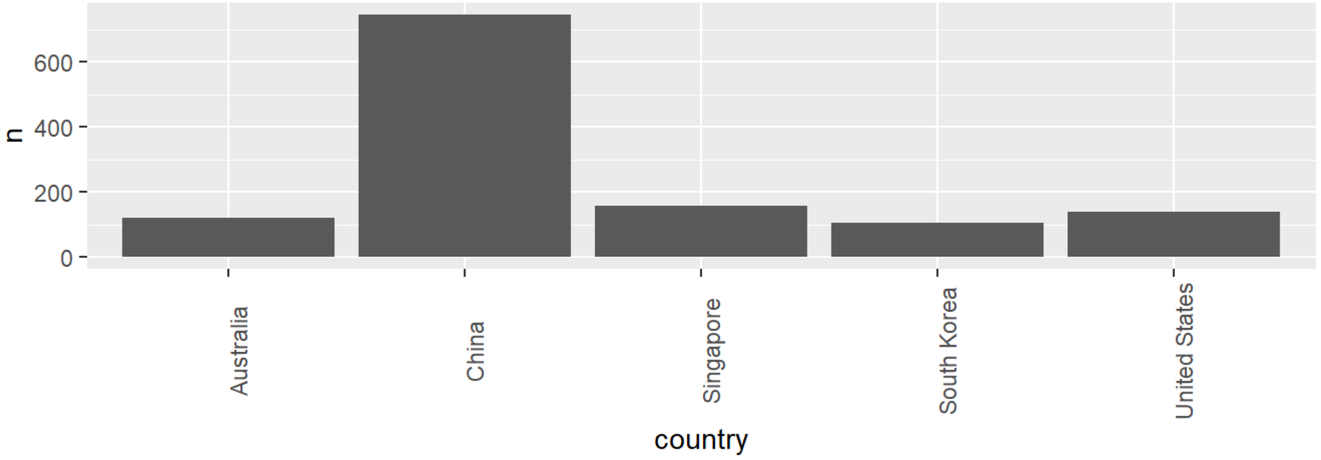


Question 11: Combining plots

Plot figures p1 and p2 in the same plot using grid.arrange() from the gridExtra R package.

```
grid.arrange(p1, p2) # 3pts
```


n
600
400
200
0
Australia
China
Singapore
South Korea
United States
country


n
600
400
200
0
Australia
China
Singapore
South Korea
United States
country