MATH2349 Data Wrangling Assignment 2

Code

Assignment 2

Kristian James Guinto (s3826723)

Required Packages

Hide

```
library(dplyr)
library(readr)
library(tidyr)
library(lubridate)
library(ggplot2)
library(Hmisc)
library(knitr)
knitr::opts_chunk$set(warning = FALSE)
```

Executive Summary

This notebook shows the pre-processing and merging of historical Covid-19 Cases, Deaths & Test by State (Covid-19) data set from ESRI and the Google mobility data set. The objective of this project is to create a tidy data set that can easily be filtered by date and state information to analyse the relationship between Covid-19 cases and deaths, and mobility.
Both data sets were pre-processed by checking and converting the data types and scanning for missing values and outliers. Additionally, specific steps to clean the Covid data set were 1.) tidying the data by creating three new columns; state, cases and deaths, 2.) computing for daily new cases and finally 3.) applying log transformation to cumulative case numbers.
The two data sets were cleaned separately then merged at the end. The final output is a clean and tidy version of the combination of Covid-19 statistics and mobility data that can easily be filtered by state and date information.

Report Structure

This report is divided into three (3) main sections. The first 2 two sections discuss the pre-processing steps taken for the Covid-19 Data set and Google mobility data set respectively. The steps that will be discussed include **Loading the data**, **Understanding the data**, **Converting data types and subsetting**, **Missing values**, **Tidy and manipulate**, **Outlier detection** and **Data transformation**. The final section will discuss merging the cleaned versions of the two data sets and a summary describing the final data set.

Historical Covid-19 Cases, Deaths & Test by State

Data

Source: https://covid19-esriau.hub.arcgis.com/datasets/historical-cases-deaths-tests-by-state (https://covid19-esriau.hub.arcgis.com/datasets/historical-cases-deaths-tests-by-state) [1]
Description: Daily cumulative cases, deaths, and positive and negative tests per Australian state from 23

January 2020 until 14 October 2020. The states included are New South Wales (NSW), Victoria (VIC), Queensland (QLD), Western Australia (WA), Tasmania (TAS), Northern Territory (NT) and Australian Capital Territory (ACT). [1]

Variables:

Variable/s (# columns)	Description
ObjectId (1)	Unique indicator
Date (1)	Date of observation [23 January to 14 October]
Cases (9)	9 columns for cumulative number of cases per state (8) and nationwide (1)
Deaths (9)	9 columns for cumulative number of deaths per state (8) and nationwide (1)
Covid Tests (8)	8 columns for cumulative number of tests per state
Negative Tests (8)	8 columns for cumulative number of negative tests per state
Total Test (1)	Total tests in Australia
Total Negative Test (1)	Total negative tests

Hide

```
covid_AU <- read_csv('Historical_Cases%2C_Deaths_%26_Tests_by_State.csv')
covid_AU %>% head()
```

ObjectId <dbl>	Date <chr>	N… <dbl>	VIC <dbl>	Q… <dbl>	SA <dbl>	WA <dbl>	TAS <dbl>	NT <dbl>	ACT <dbl>
589	2020/01/23 00:00:00	*NA*	*NA*	*NA*	0	*NA*	*NA*	*NA*	*NA*
590	2020/01/25 00:00:00	3	1	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
591	2020/01/27 00:00:00	4	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
592	2020/01/28 00:00:00	*NA*	*NA*	0	*NA*	*NA*	*NA*	*NA*	*NA*
593	2020/01/29 00:00:00	*NA*	2	1	0	*NA*	*NA*	*NA*	*NA*
594	2020/01/30 00:00:00	4	3	2	*NA*	*NA*	*NA*	*NA*	*NA*

6 rows | 1-10 of 38 columns

Understanding the Data (Understand)

Hide

```
# Check data types
covid_AU %>% glimpse()
```

```
Rows: 261
Columns: 38
$ ObjectId              <dbl> 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 6
00, 601, 602, 603,…
$ Date                  <chr> "2020/01/23 00:00:00", "2020/01/25 00:00:00", "2020/01/2
7 00:00:00", "2020…
$ NSW                   <dbl> NA, 3, 4, NA, NA, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4,…
$ VIC                   <dbl> NA, 1, NA, NA, 2, 3, NA, 4, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ QLD                   <dbl> NA, NA, NA, 0, 1, 2, NA, NA, NA, NA, 3, 4, 5, NA, NA, N
A, NA, NA, NA, NA, …
$ SA                    <dbl> 0, NA, NA, NA, 0, NA, 0, 0, 2, 2, 2, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ WA                    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ TAS                   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ NT                    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ ACT                   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ Total_Cases           <dbl> 0, 4, 4, 0, 3, 9, 4, 8, 6, 6, 9, 8, 9, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4,…
$ NSW_Deaths            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ VIC_Deaths            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ QLD_Deaths            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ SA_Deaths             <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ WA_Deaths             <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ TAS_Deaths            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ NT_Deaths             <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ ACT_Deaths            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ Total_Deaths          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0,…
$ NSW_Tests             <dbl> NA, NA, NA, NA, NA, 74, 76, 100, 128, 158, 199, 335, 34
5, 393, 562, 629, 6…
$ VIC_Tests             <dbl> NA, NA, NA, NA, NA, 79, NA, 162, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ QLD_Tests             <dbl> NA, NA, NA, 6, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA,…
$ SA_Tests              <dbl> 6, NA, NA, 6, 10, NA, 17, 25, 34, 56, 88, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ WA_Tests              <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ TAS_Tests             <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ NT_Tests              <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ ACT_Tests             <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
```

```
NA, NA, NA, NA, NA…
$ NSW_Tests_Negative   <dbl> NA, NA, NA, NA, NA, 74, 76, 100, 128, 158, 199, 335, 34
5, 393, 562, 629, 6…
$ VIC_Tests_Negative   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ QLD_Tests_Negative   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ SA_Tests_Negative    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ WA_Tests_Negative    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ TAS_Tests_Negative   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ NT_Tests_Negative    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ ACT_Tests_Negative   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA…
$ Total_Tests          <dbl> 6, 0, 0, 12, 10, 153, 93, 287, 162, 214, 287, 335, 345,
393, 562, 629, 674…
$ Total_Tests_Negative <dbl> 0, 0, 0, 0, 0, 74, 76, 100, 128, 158, 199, 335, 345, 39
3, 562, 629, 674, 6…
```

All numeric variables have the appropriate numeric data type. However, the date variable was assigned a character data type and this will be converted to date in the next step. The next step will also include extracting data on Covid cases and deaths as these are the target variables of this project. Additionally, since the Google mobility data is only available from 15 February to 09 October only rows within this date range will be used.

Converting Data Types and Subsetting (Understand)

Hide

```
# convert Date from chr -> date for easy subsetting and to match with mobility data s
et
covid_AU$Date <- ymd(ymd_hms(covid_AU$Date))
# select the needed columns and filter date between 15 February to 09 October
covid_AU <- covid_AU %>%
  select(Date:Total_Deaths) %>%
  filter(Date >= ymd("2020-02-15") & Date <= ymd("2020-10-09"))

covid_AU %>% head()
```

Date <date>	NSW <dbl>	VIC <dbl>	QLD <dbl>	SA <dbl>	WA <dbl>	TAS <dbl>	NT <dbl>	ACT <dbl>	Total_Cases <dbl>
2020-02-16	4	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	4
2020-02-17	4	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	4
2020-02-18	4	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	4
2020-02-19	4	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	4
2020-02-20	4	*NA*	*NA*	2	*NA*	*NA*	*NA*	*NA*	6
2020-02-21	4	*NA*	*NA*	*NA*	1	*NA*	*NA*	*NA*	5

6 rows | 1-10 of 19 columns

Missing values (Scan I)

Hide

```
## helper function to map NA -> 1 and non NA -> 0
missingDummy <- function(x){
  if(is.numeric(x)){
    x[!is.na(x)] <- 0
    x[is.na(x)] <- 1
  }
  return(x)
}
## function to plot missing values in a heat map
plotMissing <- function(df){
  colors <- c("grey", "black")
  df %>%
  lapply(FUN = missingDummy) %>% # map NA -> 1 and non NA -> 0
  as.data.frame() %>% # convert back to data frame
  gather(key = "Column", value = "value", -Date) %>% # gather for plotting
  ggplot() + # ggplot
    geom_tile(aes(x = Date, y = Column, fill = factor(value))) +
    scale_fill_manual(values=colors, name = "legend", labels = c("Present", "Missing"
)) +
    ggtitle("Missing Values")
}
```

Missing dates in the data set will be first checked before checking for missing data in the other variables. This will be done by creating a complete sequence of dates from 15 February to 10 October then cross checking the date variable with this sequence.

Hide

```
# check for missing dates
dates_seq <- seq(as.Date("2020-02-15"), # complete sequence of dates from 15 February
             as.Date("2020-10-09"), # to 10 October
             by = 1)
dates_seq[!dates_seq %in% covid_AU$Date] # cross check dates in data with dates_seq
```

```
[1] "2020-02-15" "2020-02-27" "2020-03-02"
```

Results show that there are three missing dates in the date variable. Since “2020-02-15” should have been the first date in the data set this row will just be ignored and will also be considered for the pre-processing steps of the Google Mobility data. The two remaining missing dates will be added to the data set and initialised with *NA* as values for the other variables.

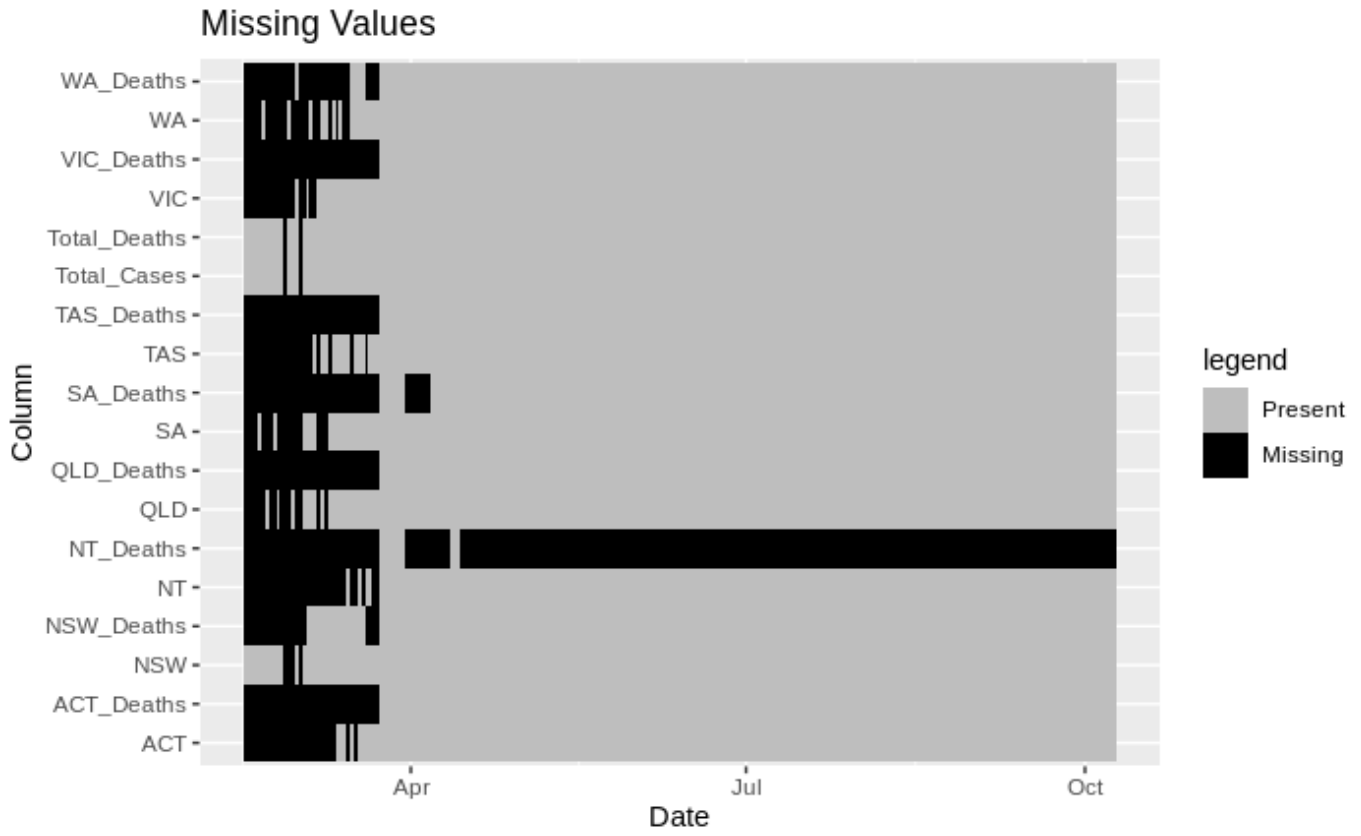
Hide

```
# add missing dates with NA as value for other columns
newRow <- data.frame(matrix(nrow = 2, ncol = ncol(covid_AU))) # initiliase empty df
colnames(newRow) = colnames(covid_AU) # copy header names
newRow$Date <- as.Date(newRow$Date) # make date
newRow[1,]$Date = as.Date("2020-02-27")
newRow[2,]$Date = as.Date("2020-03-02")

covid_AU <- rbind(covid_AU, newRow) %>% arrange(Date) # add to covid_AU then sort
```

Graphical methods will be used to scan for missing values in the data set. The function *plotMissing* was shown at the beginning of this section and was created to show missing values sequentially.

Hide

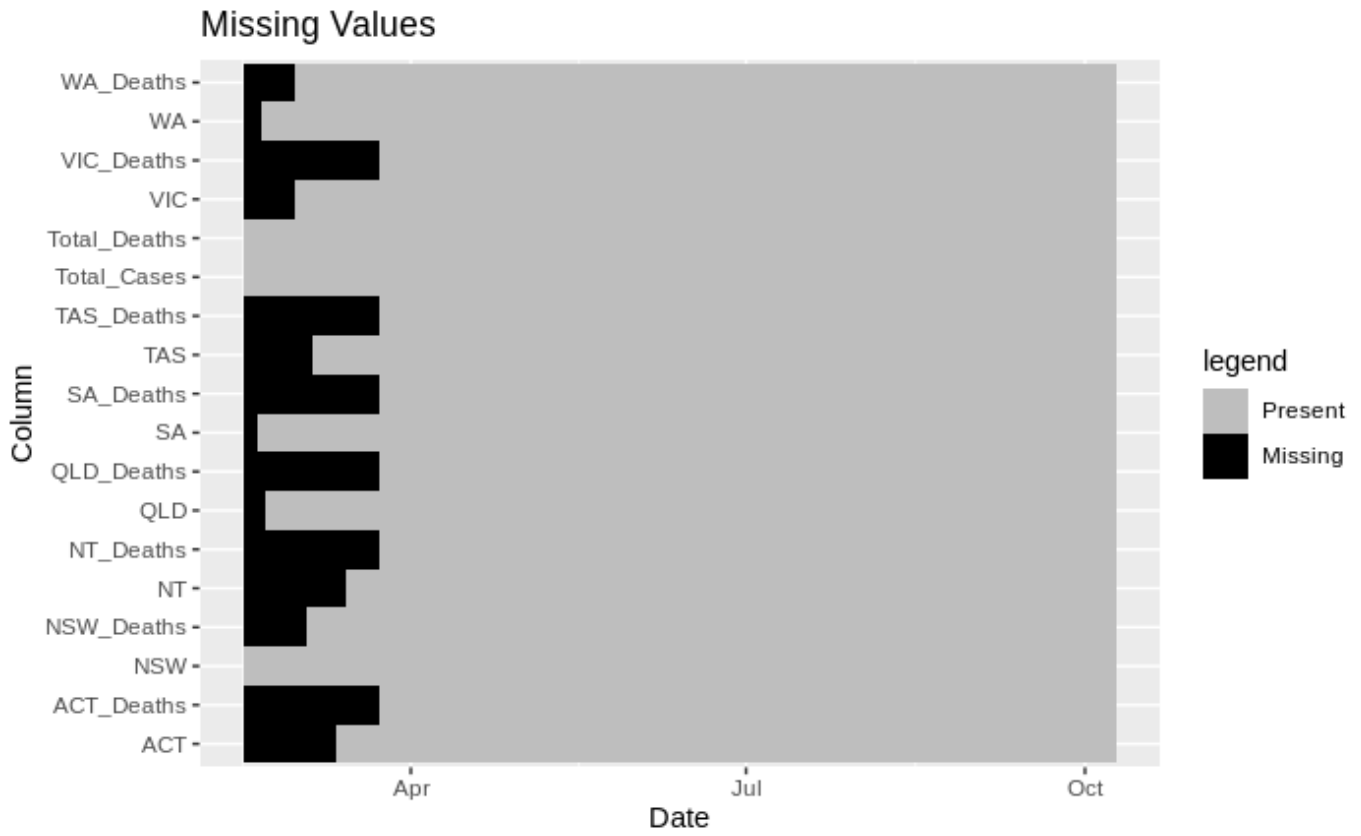
```
# Scan for missing values
plotMissing(covid_AU) # graphical method
```



The Y-axis on the plot are the different variables in the data set and the X-axis are the dates. The black sections represent missing data for each variable.
The variables in this data set are cumulative numbers of cases and deaths thus the missing data can be filled with the values from the previous day. The *fill()* function from base R will be used for this step.

Hide

```
# since the data are cumulative cases, fill null values with previous value
covid_AU <- covid_AU %>%
  fill(NSW:Total_Deaths, .direction = "down") # fill()

plotMissing(covid_AU) # graphical method
```



Most of the missing values were filled from the previous step but there are still some missing values that remain. The remaining missing values occur from the start date and ended after a few observations. These missing values are interpreted as rows before the first cases/deaths were recorded and thus can be filled with 0.

Hide

```
# As seen in the graph, remaining NA's are those values before the first case or deat
h
covid_AU[is.na(covid_AU)] <- 0 # fill with 0's
colSums(is.na(covid_AU))
```

```
        Date          NSW          VIC          QLD           SA           WA
TAS           NT
           0            0            0            0            0            0
0            0
         ACT  Total_Cases   NSW_Deaths   VIC_Deaths   QLD_Deaths    SA_Deaths    WA_D
eaths   TAS_Deaths
           0            0            0            0            0            0
0            0
   NT_Deaths   ACT_Deaths Total_Deaths
           0            0            0
```

Infinite and Nan values will also be scanned using a function taken from the lecture notes [2].

Hide

```
is.special <- function(x){
  if (is.numeric(x)) (is.infinite(x) | is.nan(x))
} # taken from lecture notes

covid_AU %>%
  sapply(., is.special) %>%
  sapply(., sum)
```

```
       Date          NSW          VIC          QLD           SA           WA
TAS           NT
          0            0            0            0            0            0
0            0
         ACT  Total_Cases   NSW_Deaths   VIC_Deaths   QLD_Deaths    SA_Deaths    WA_D
eaths   TAS_Deaths
          0            0            0            0            0            0
0            0
   NT_Deaths   ACT_Deaths Total_Deaths
          0            0            0
```

Tidy & Manipulate

Tidy & Manipulate II

Another variable of interest to consider when analysing Covid-19 are daily new cases. New columns will be created for daily new cases of each state using *mutate_at()* function. Finally, a total daily new cases column will also be created by summing the new columns of daily cases.

Hide

```
# helper function to compute daily new cases
daily <- function(x){
  return (x - lag(x)) # subtracts the data with a lagged 1 copy of the data
}
# adds new columns for daily cases of each state
covid_AU_wDaily <- covid_AU %>%
  mutate_at(
    c("NSW","VIC","QLD","SA","WA","TAS","NT","ACT"),
    funs(Daily = daily)
    )
# there will new NA entries at the first row of daily cases.
# Since this is just one row, this will just be dropped
covid_AU_wDaily <- covid_AU_wDaily %>% drop_na()
#covid_AU_wDaily %>% glimpse()

# add additional column for daily new cases in Australia
covid_AU_wDaily <- covid_AU_wDaily %>%
  mutate(Total_Daily = rowSums(.[20:27]))

covid_AU_wDaily %>% head()
```

Total_Deaths <dbl>	NSW_D… <dbl>	VIC_Daily <dbl>	QLD_Daily <dbl>	SA_Daily <dbl>	WA_D… <dbl>	TAS_Daily <dbl>	NT_Daily <dbl>	A
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	

Total_Deaths <dbl>	NSW_D… <dbl>	VIC_Daily <dbl>	QLD_Daily <dbl>	SA_Daily <dbl>	WA_D… <dbl>	TAS_Daily <dbl>	NT_Daily <dbl>	A
0	0	0	0	0	0	0	0	
0	0	0	0	2	0	0	0	
0	0	0	0	0	1	0	0	
0	0	0	7	0	0	0	0	

6 rows | 19-27 of 28 columns

Tidy & Manipulate I

So far the covid data set used is untidy as the columns for cases, deaths and the created daily cases also contain state information. The state information can instead be created as a new column to create tidy data.

Hide

```
# divide data into cases, deaths and daily
covid_cases <- covid_AU_wDaily %>% select(Date:Total_Cases) # subset of cases
covid_deaths <- covid_AU_wDaily %>% select(Date, NSW_Deaths:Total_Deaths) # subset of
deaths
covid_daily <- covid_AU_wDaily %>% select(Date, NSW_Daily:Total_Daily) # subset of de
aths

# gather cases
newName <- c("Date","NSW","VIC","QLD","SA","WA","TAS","NT","ACT","AUS")
colnames(covid_cases) <- newName
colnames(covid_deaths) <- newName
colnames(covid_daily) <- newName
tidy_cases <- covid_cases %>% # tidy covid cases subset
  gather(key = "State", value = "Cases", -Date)
tidy_death <- covid_deaths %>% # tidy covid deaths subset
  gather(key = "State", value = "Deaths", -Date)
tidy_daily <- covid_daily %>%
  gather(key = "State", value = "Daily_Cases", -Date)

# merge tidy data sets
tidy_covid_au <- left_join(tidy_cases, tidy_death, by=c("Date", "State"))
tidy_covid_au <- left_join(tidy_covid_au, tidy_daily, by=c("Date", "State"))
tidy_covid_au %>% head()
```

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>
2020-02-17	NSW	4	0	0
2020-02-18	NSW	4	0	0
2020-02-19	NSW	4	0	0
2020-02-20	NSW	4	0	0
2020-02-21	NSW	4	0	0
2020-02-22	NSW	4	0	0

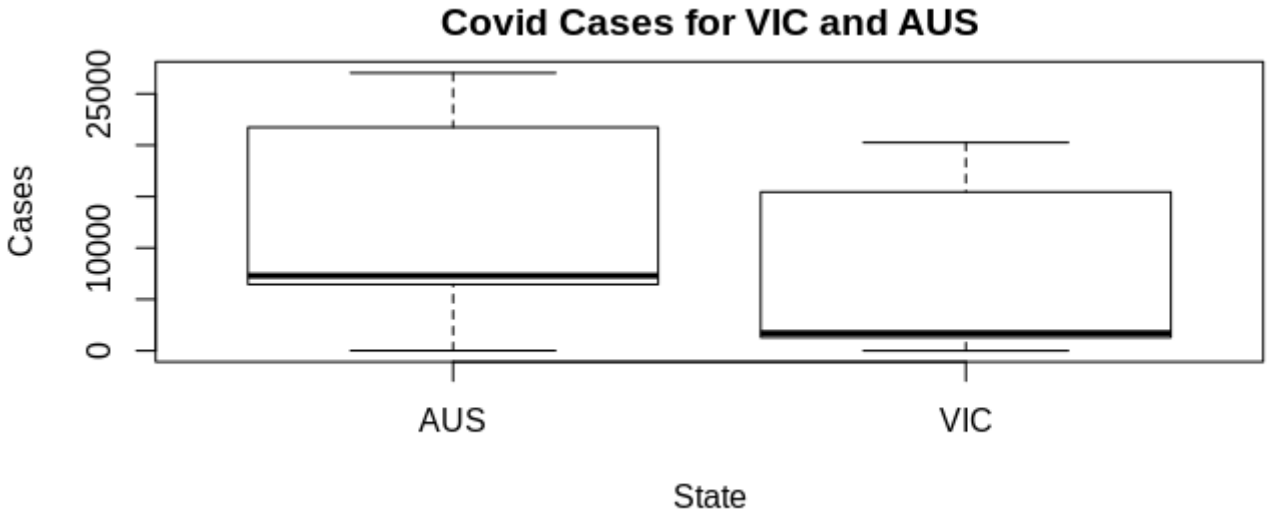
6 rows

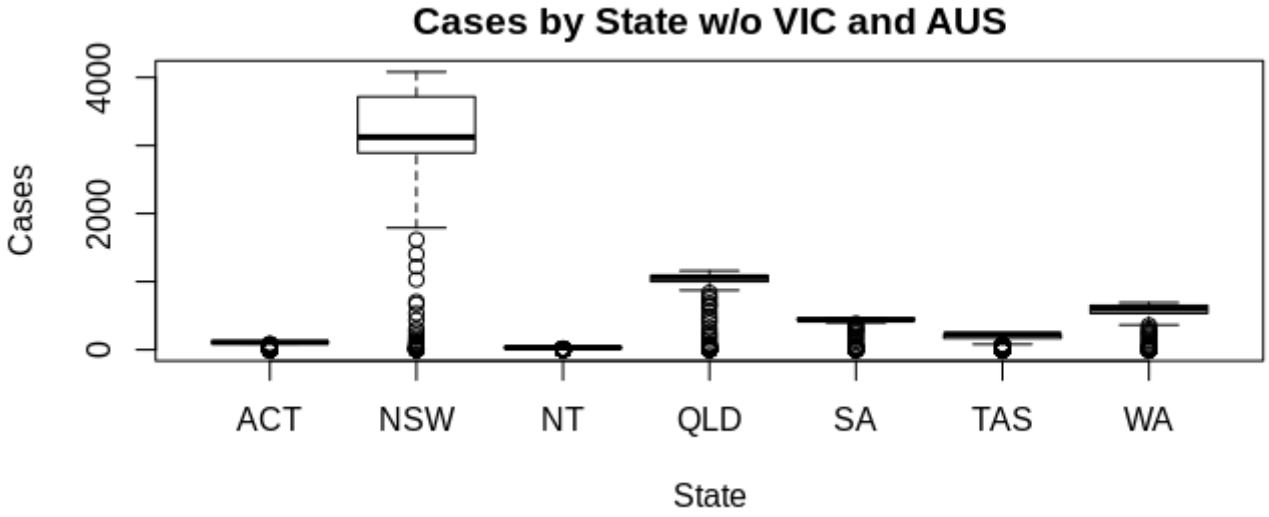
Outlier Detection (Scan II)

To account for within State effects on Covid cases and deaths, outlier detection will be done individually for each state. Outliers will be inspected for all numerical variables (Cases, Deaths and Daily_Cases)

Hide

```
par(mfrow=c(2,1))
temp1 <- tidy_covid_au %>%
  filter(State == "AUS" | State == "VIC")
boxplot(temp1$Cases ~ temp1$State, main="Covid Cases for VIC and AUS",
        ylab = "Cases", xlab = "State")
temp2 <- tidy_covid_au %>%
  filter(State != "AUS" & State != "VIC")
boxplot(temp2$Cases ~ temp2$State,main = "Cases by State w/o VIC and AUS",
        ylab = "Cases",xlab = "State")
```

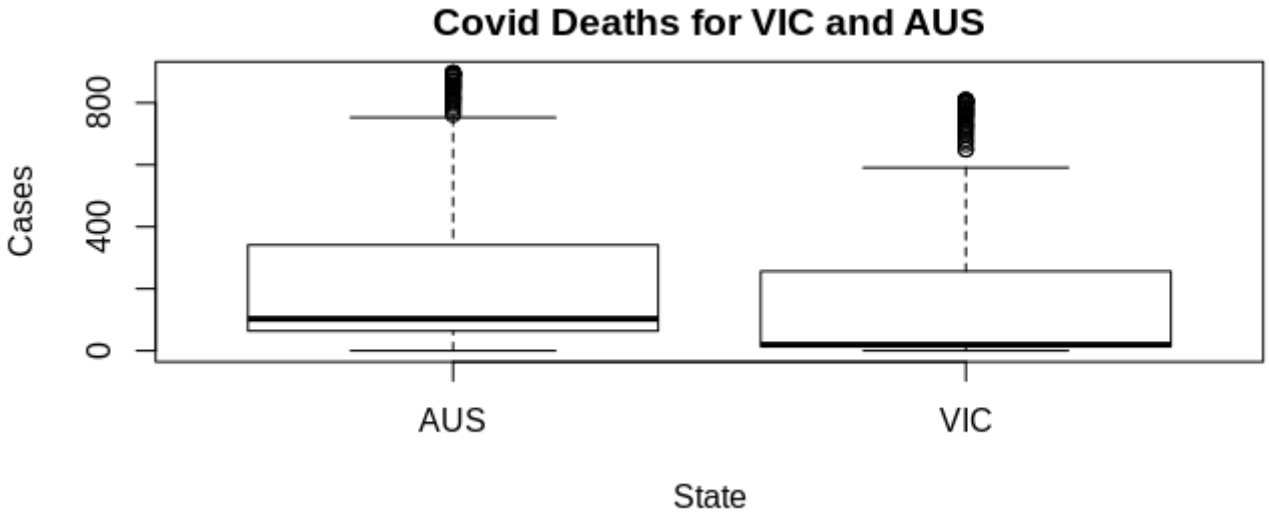


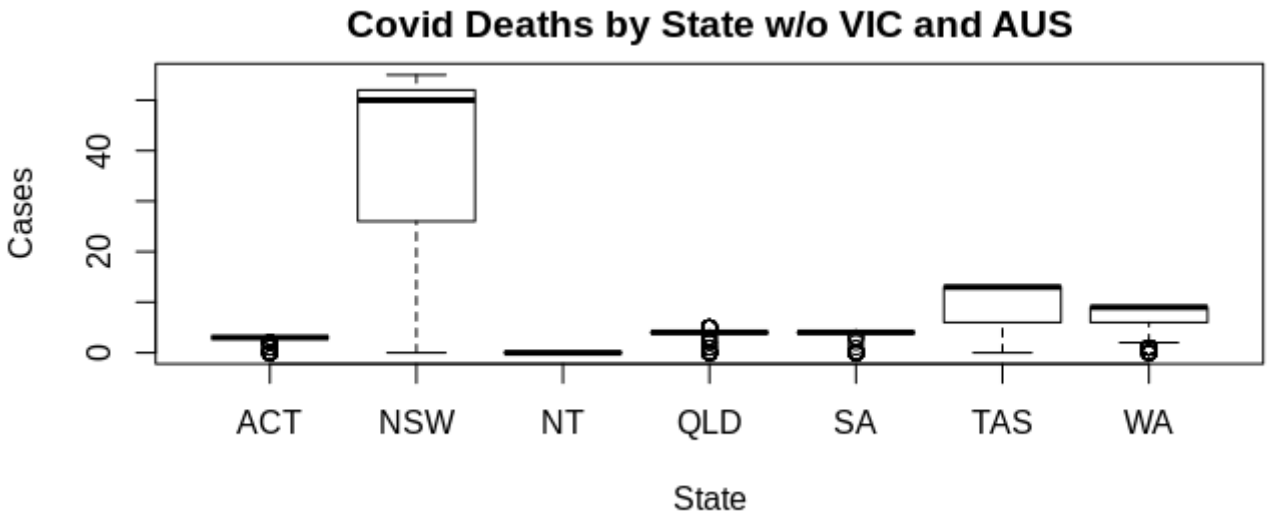


Outliers are expected for this variable as we are dealing with time-series cumulative cases so these values will not be imputed nor removed.

Hide

```
par(mfrow=c(2,1))
temp1 <- tidy_covid_au %>%
  filter(State == "AUS" | State == "VIC")
boxplot(temp1$Deaths ~ temp1$State, main="Covid Deaths for VIC and AUS",
        ylab = "Cases", xlab = "State")
temp2 <- tidy_covid_au %>%
  filter(State != "AUS" & State != "VIC")
boxplot(temp2$Deaths ~ temp2$State,main = "Covid Deaths by State w/o VIC and AUS",
        ylab = "Cases",xlab = "State")
```

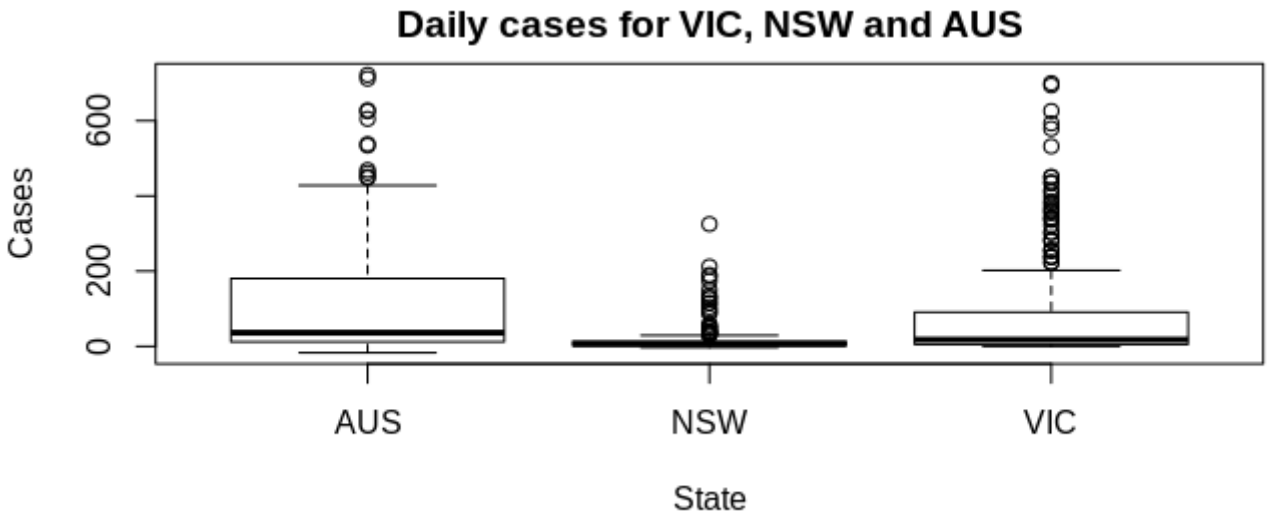


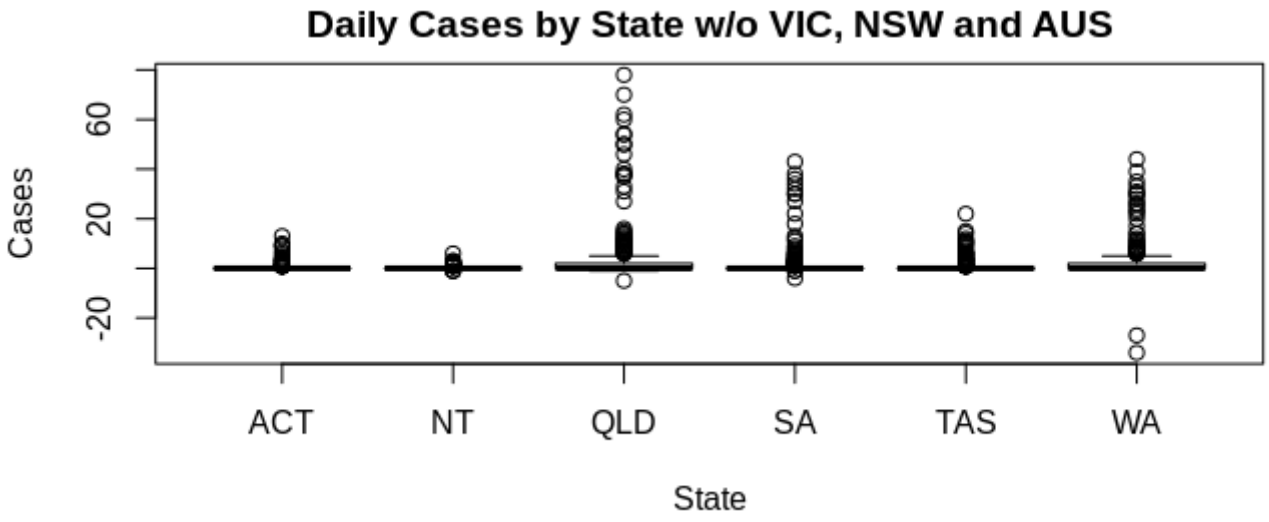


Similar to the number of cases, outliers in the number of deaths are also expected hence these values will not be imputed nor removed.

Hide

```
par(mfrow=c(2,1))
temp1 <- tidy_covid_au %>%
  filter(State == "AUS" | State == "VIC" | State == "NSW")
boxplot(temp1$Daily_Cases ~ temp1$State, main="Daily cases for VIC, NSW and AUS",
        ylab = "Cases", xlab = "State")
temp2 <- tidy_covid_au %>%
  filter(State != "AUS" & State != "VIC" & State != "NSW")
boxplot(temp2$Daily_Cases ~ temp2$State,main = "Daily Cases by State w/o VIC, NSW and
AUS",
        ylab = "Cases",xlab = "State")
```





For the daily new cases, the negative outlier values in Western Australia (WA) need further investigation as this variable should not have large negative values. Small negative values for daily cases can be attributed to minor changes due to corrections on the number of cases.

Hide

```
tidy_covid_au %>%
  filter(State == "WA") %>% # filter on Western Australia only
  filter(Daily_Cases < 0) # filter on negative daily cases
```

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>
2020-06-04	WA	557	9	-34
2020-08-02	WA	641	9	-27

2 rows

Further investigation reveal the following about these dates:
2020-06-04: Data entry error since the 557 was the number of recoveries and 592 was the true number of cases[3]. This corrections will be made on the next step.
2020-08-02: According to the website of the Department of Health of Western Australia the reduction was because “historical cases diagnosed through a blood test and indicative of infection in the past are no longer being reported in the daily report or in overall figures.” [4]

Hide

```
# Correcting erroneous outliers.
# Before correction
tidy_covid_au %>%
  filter(State == "WA") %>%
  filter(Date >= "2020-06-03" & Date <= "2020-06-05")
```

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>
2020-06-03	WA	591	9	0
2020-06-04	WA	557	9	-34
2020-06-05	WA	596	9	39

3 rows

Hide

```
tidy_covid_au <- tidy_covid_au %>%
  mutate(Cases=replace(Cases, Date=="2020-06-04" & State == "WA",592)) %>%  # replace
557 with 592
  mutate(Daily_Cases=replace(Daily_Cases, Date=="2020-06-04" & State == "WA", 1)) %>%
# new daily case will be 1
  mutate(Daily_Cases=replace(Daily_Cases, Date=="2020-06-05" & State == "WA", 4)) # n
ew daily case for 05 June will be 4
# After correction
tidy_covid_au %>%
  filter(State == "WA") %>%
  filter(Date >= "2020-06-03" & Date <= "2020-06-05")
```

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>
2020-06-03	WA	591	9	0
2020-06-04	WA	592	9	1

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>
2020-06-05	WA	596	9	4

3 rows

Data Transformation (Transform)

Covid-19 cases plotted on a linear scale can be deceptive because of its exponential rate of increase. Another visualisation option is to plot the cumulative cases on a logarithmic scale. This method highlights growth rate instead of just raw numbers which can be more valuable for analysis of a pandemic. [5]
The logarithmic transformation of the cumulative cases will be taken to prepare the data for potential future visualisation.

Hide

```
tidy_covid_au$Log_Cases <- log(tidy_covid_au$Cases)
tidy_covid_au %>% head()
```

Date <date>	State <chr>	Cases <dbl>	Deaths <dbl>	Daily_Cases <dbl>	Log_Cases <dbl>
2020-02-17	NSW	4	0	0	1.386294
2020-02-18	NSW	4	0	0	1.386294
2020-02-19	NSW	4	0	0	1.386294
2020-02-20	NSW	4	0	0	1.386294
2020-02-21	NSW	4	0	0	1.386294
2020-02-22	NSW	4	0	0	1.386294

6 rows

(Understand)

As a final step for this data set, the State variable will be converted from Character to unordered factor.

Hide

```
tidy_covid_au$State <- factor(tidy_covid_au$State)
```

Google Mobility

Data

Source: https://www.google.com/covid19/mobility/index.html?hl=en (https://www.google.com/covid19/mobility/index.html?hl=en) [6]
Description: Aggregate data collected and processed by Google showing the change on the number of visits and length of stay on 6 different place classifications relative to pre-Covid levels. According to Google the baseline value used for comparison were “the median values during the 5-week period of January 3 to February 6” [7]. This data set is available for various regions of the world but for this project only Australian data will be used. The data set contain information from 15 February 2020 to 09 October 2020.
Variables:

Variable	Description
county_region_code	Unique code assigned by Google
country_region	Name of the country
sub_region_1	Names of States (Australia)
sub_region_2	Local Government Areas (Australia)
metro_area	N/A for Australia Data
iso_3166_2_code	standard ISO codes for countries and regions
census_fips_code	Geographical codes applicable to USA
date	Date of observation
retail_and_recreation_percent_change_from_baseline	% change of mobility on retail and recreation places
grocery_and_pharmacy_percent_change_from_baseline	% change of mobility on grocery and pharmacy
parks_percent_change_from_baseline	% change of mobility on parks
transit_stations_percent_change_from_baseline	% change of mobility on transit stations
workplaces_percent_change_from_baseline	% change of mobility on workplaces
residential_percent_change_from_baseline	% change of mobility on residential

Hide

```
mobility_AU <- read_csv('/home/guinto/Documents/RProjects/Data Wrangling/HW2/2020_AU_
Region_Mobility_Report.csv')
mobility_AU %>% head()
```

country_region_code <chr>	**country_region** <chr>	**sub_region_1** <chr>	**sub_region_2** <chr>	**metro_area** <lgl>	**iso_3166_2** <chr>
AU	Australia	*NA*	*NA*	*NA*	*NA*
AU	Australia	*NA*	*NA*	*NA*	*NA*
AU	Australia	*NA*	*NA*	*NA*	*NA*
AU	Australia	*NA*	*NA*	*NA*	*NA*
AU	Australia	*NA*	*NA*	*NA*	*NA*
AU	Australia	*NA*	*NA*	*NA*	*NA*

6 rows | 1-6 of 14 columns

Understanding the data (Understand)

Hide

```
mobility_AU %>% glimpse()
```

```
Rows: 61,487
Columns: 14
$ country_region_code                                <chr> "AU", "AU", "AU", "AU", "A
U", "AU", "AU", "A…
$ country_region                                     <chr> "Australia", "Australia",
"Australia", "Aust…
$ sub_region_1                                       <chr> NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ sub_region_2                                       <chr> NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ metro_area                                         <lgl> NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ iso_3166_2_code                                    <chr> NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ census_fips_code                                   <lgl> NA, NA, NA, NA, NA, NA, N
A, NA, NA, NA, NA, …
$ date                                               <date> 2020-02-15, 2020-02-16, 2
020-02-17, 2020-02…
$ retail_and_recreation_percent_change_from_baseline <dbl> 4, 3, -1, -3, -1, 0, 3, 5,
3, -1, -3, -1, 0,…
$ grocery_and_pharmacy_percent_change_from_baseline  <dbl> 3, 5, 0, -2, -1, 1, 4, 4,
4, 1, -1, 0, 1, 6,…
$ parks_percent_change_from_baseline                 <dbl> -2, 9, -6, -13, -6, 5, -1,
10, 9, -10, -8, -…
$ transit_stations_percent_change_from_baseline      <dbl> 3, 3, 7, 7, 8, 9, 12, 8,
4, 8, 10, 10, 10, 1…
$ workplaces_percent_change_from_baseline            <dbl> 3, -1, 17, 14, 13, 13, 16,
3, -2, 17, 15, 14…
$ residential_percent_change_from_baseline           <dbl> 0, 0, -2, -1, -1, -2, -3,
-1, 0, -1, -1, -1,…
```

The target variables for this data set are sub_region_1 (contains state information), date and the mobility percent change variables. Current data types are suitable for this stage of pre-processing. Target variables and rows will be extracted in the next step.

Country region variable will be processed later by renaming and converting to factor to match with the Covid data set.

Subsetting (Understand)

Hide

```
mobility_AU <- mobility_AU %>%
  filter(is.na(sub_region_2)) %>% # filter data only per region and entire Au
  filter(date >= ymd("2020-02-17") & date <= ymd("2020-10-10")) %>%
  select(sub_region_1, date:residential_percent_change_from_baseline) # select column
s
# rename columns
colnames(mobility_AU) <- c("State","Date","Retail_and_Recreation",
                           "Grocery_and_Pharmacy","Parks","Transit_Stations",
                           "Workplaces","Residential")
mobility_AU %>% head()
```

State <chr>	Date <date>	Retail_and_Recreation <dbl>	Grocery_and_Pharmacy <dbl>	Pa… <dbl>	Transit_Station <db
NA	2020-02-17	-1	0	-6	
NA	2020-02-18	-3	-2	-13	
NA	2020-02-19	-1	-1	-6	
NA	2020-02-20	0	1	5	
NA	2020-02-21	3	4	-1	1
NA	2020-02-22	5	4	10	

6 rows | 1-7 of 8 columns

The sub_region_2 variable represents different LGA's in Australia. Rows where this variable is NA represent data for larger geographical sections (States, Country). This information was used to remove data for LGA's. The dates were also filtered from 17 February to 09 October to match with the Covid data set.

Missing values (Scan I)

The date variable will also be checked for missing dates from 17 February to 09 October.

Hide

```
# check for missing dates
dates_seq <- seq(as.Date("2020-02-17"), # complete sequence of dates from 15 February
             as.Date("2020-10-09"), # to 10 October
             by = 1)
dates_seq[!dates_seq %in% mobility_AU$Date] # cross check dates in data with dates_se
q
```

```
Date of length 0
```

There were no missing dates in this data set. The next step will be to check for missing values on the other variables.

Hide

```
colSums(is.na(mobility_AU))
```

```
               State                    Date Retail_and_Recreation  Grocery_and_Pharma
cy
                 236                       0                     25
26
               Parks      Transit_Stations             Workplaces           Residenti
al
                  50                      30                      9
4
```

Similar to the variable sub_region_2, NA values in the State variable represent larger demographic sections. In this case it represents data for the entire Australia thus NA values of state will be filled with “AUS” to match the code used in the Covid data set.

Hide

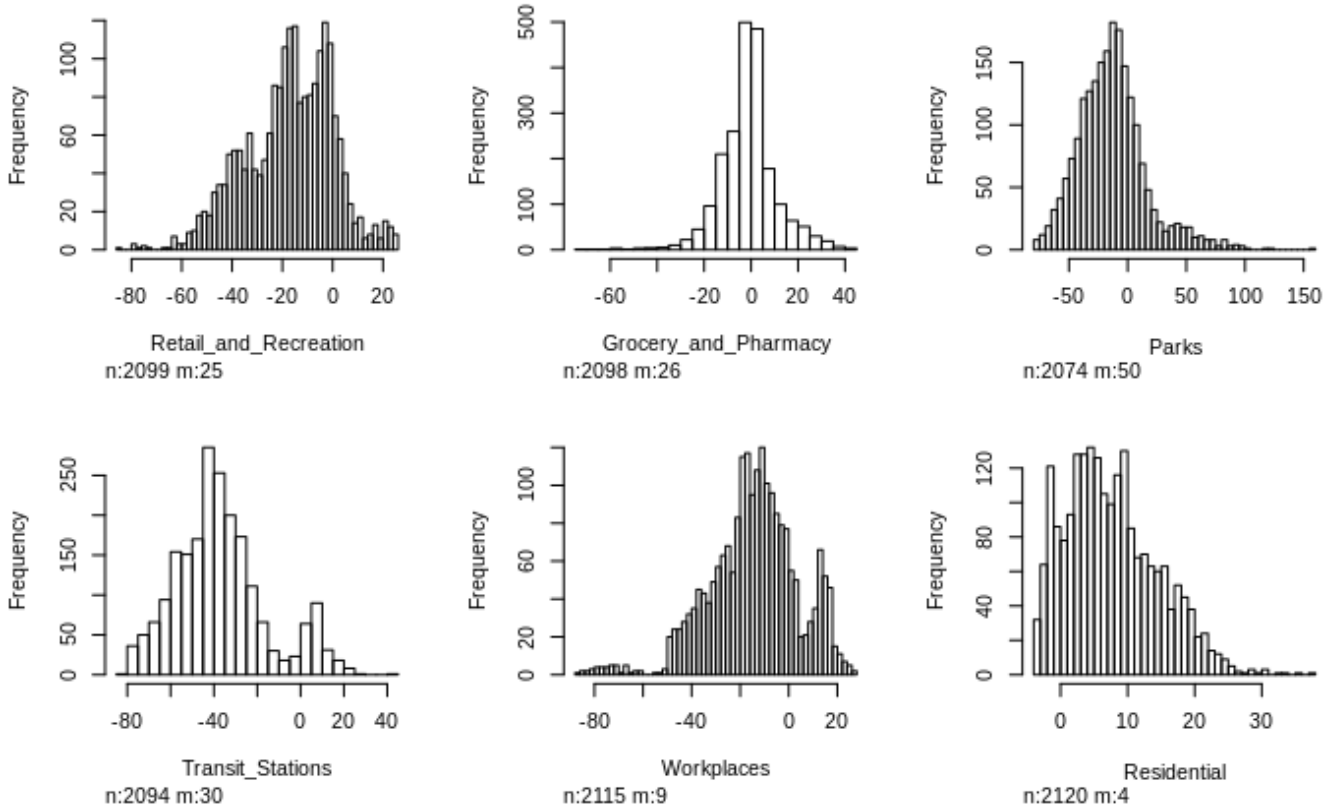
```
# fill NA State values with AU to represent data for entire Australia
mobility_AU$State[is.na(mobility_AU$State)] <- "AUS"
```

For missing values on the mobility percent change variables, histograms will be used to determine the appropriate statistic for replacement. Median will be used to replace variables with skewed distribution while mean for variables with symmetric distribution.

Hide

```
# plot histograms of mobility percent change variables
mobility_AU %>%
  select(Retail_and_Recreation:Residential) %>%
  hist()
```

```
click left mouse button to proceed
```



Mean will be used as replacement for NA's for grocery and pharmacy while median will be used for the other variables.

Hide

```
# impute with mean
mobility_AU$Grocery_and_Pharmacy <- impute(mobility_AU$Grocery_and_Pharmacy, fun = me
an)
# impute with median
mobility_AU$Retail_and_Recreation <-impute(mobility_AU$Retail_and_Recreation, fun = m
edian)
mobility_AU$Parks <- impute(mobility_AU$Parks, fun = median)
mobility_AU$Transit_Stations <- impute(mobility_AU$Transit_Stations, fun = median)
mobility_AU$Workplaces <- impute(mobility_AU$Workplaces, fun = median)
mobility_AU$Residential <- impute(mobility_AU$Residential, fun = median)
# check for missing values after processing
colSums(is.na(mobility_AU))
```

```
                State                 Date Retail_and_Recreation  Grocery_and_Pharma
cy
                    0                    0                     0
0
               Parks      Transit_Stations            Workplaces           Residenti
al
                    0                    0                     0
0
```

Infinite and Nan values will also be scanned using the function used for Covid.

Hide

```
mobility_AU %>%
  sapply(., is.special) %>%
  sapply(., sum)
```

```
               State                    Date Retail_and_Recreation  Grocery_and_Pharma
cy
                   0                       0                      0
0
               Parks      Transit_Stations             Workplaces           Residenti
al
                   0                       0                      0
0
```

Tidy & manipulate

No new variables will be formed from this data set. Furthermore, the percent change variables will not be converted to long form since our goal is to create a data set that can be analysed by state or by different ranges of date.

To prepare for merging with the Covid data set, the State variables will be mapped to codes used in the Covid data set and will also be converted to factor.

Hide

```
mobility_AU$State <- factor(mobility_AU$State,
                        levels = c("AUS","Australian Capital Territory","New South Wal
es",
                                   "Northern Territory","Queensland","South Australia"
,
                                   "Tasmania","Victoria","Western Australia"),
                        labels = c("AUS","ACT","NSW","NT","QLD","SA","TAS","VIC","WA"
))
```

Outliers (Scan II)

Hide

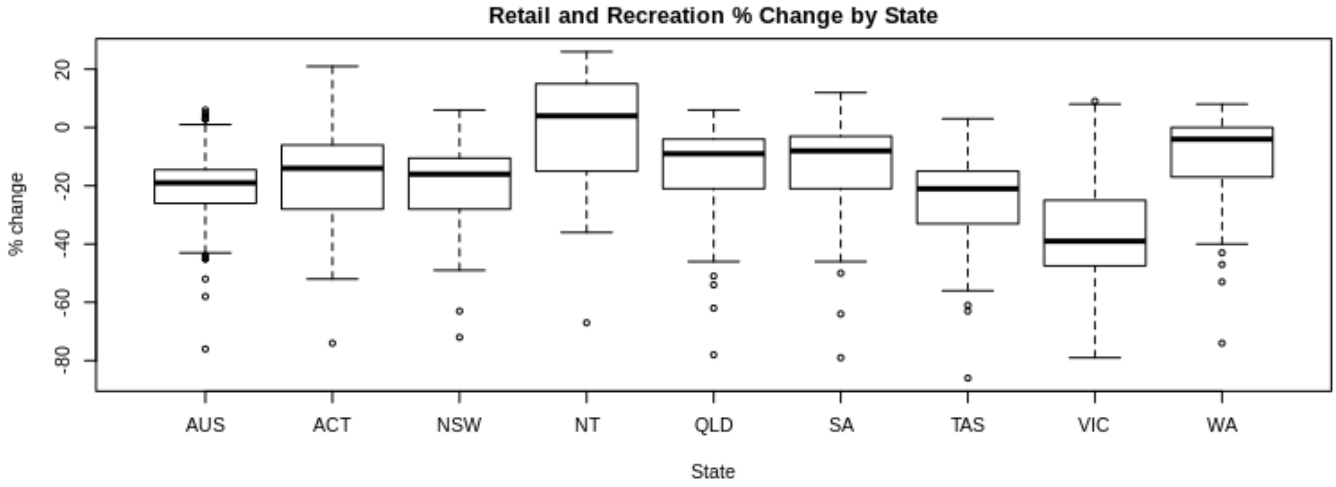
```
par(mfrow=c(6,1))

boxplot(mobility_AU$Retail_and_Recreation ~ mobility_AU$State, main="Retail and Recre
ation % Change by State",
        ylab = "% change", xlab = "State")
boxplot(mobility_AU$Grocery_and_Pharmacy ~ mobility_AU$State, main="Grocery and Pharm
acy % Change by State",
        ylab = "% change", xlab = "State")
```

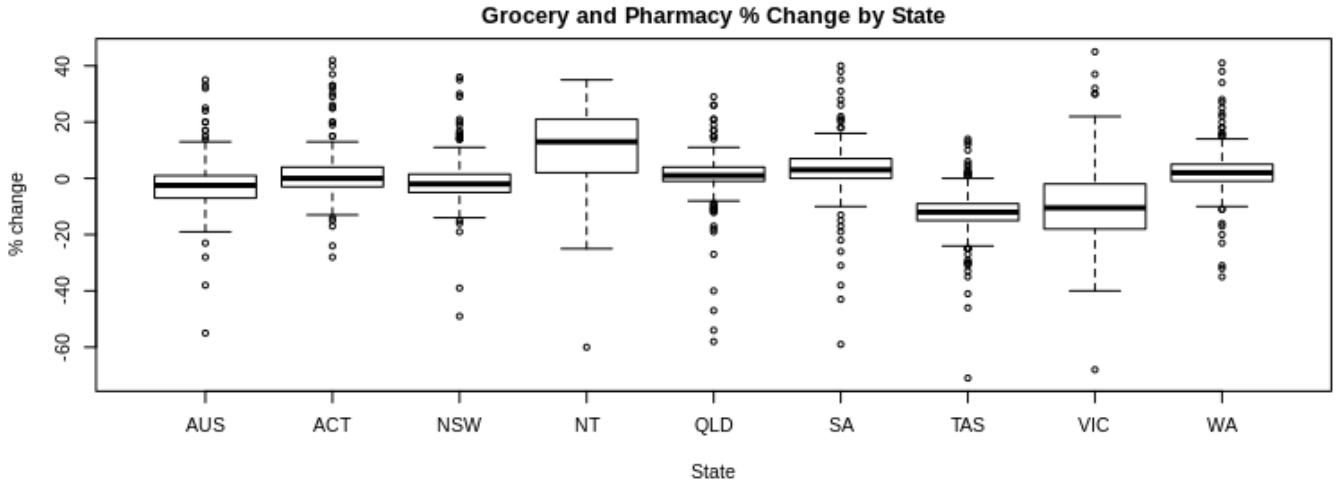
Hide

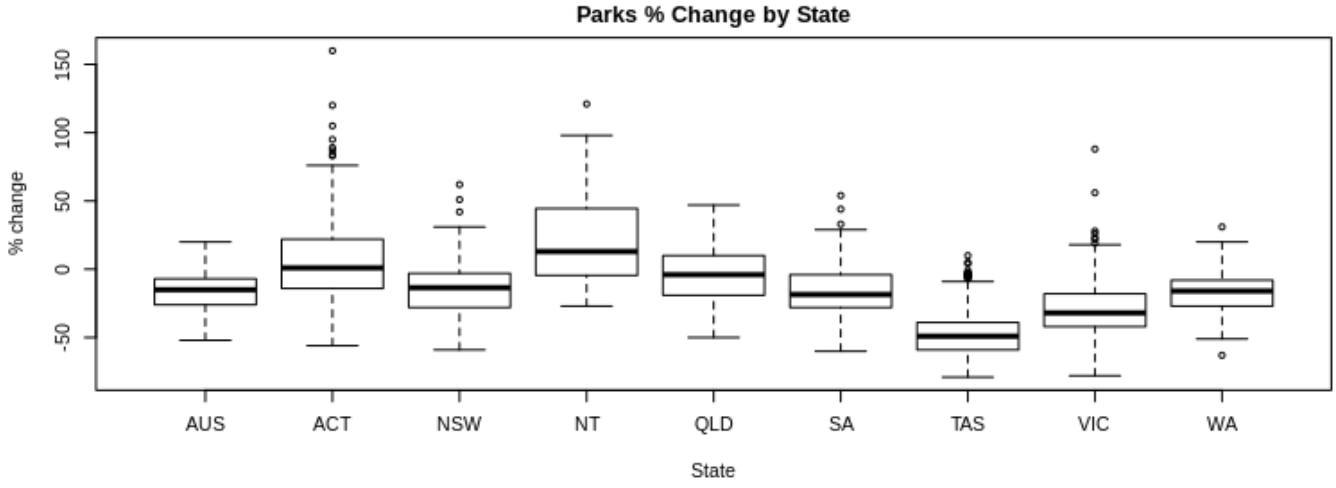
```
boxplot(mobility_AU$Parks ~ mobility_AU$State, main="Parks % Change by State",
        ylab = "% change", xlab = "State")
boxplot(mobility_AU$Transit_Stations ~ mobility_AU$State, main="Transit Stations % Ch
ange by State",
        ylab = "% change", xlab = "State")
```

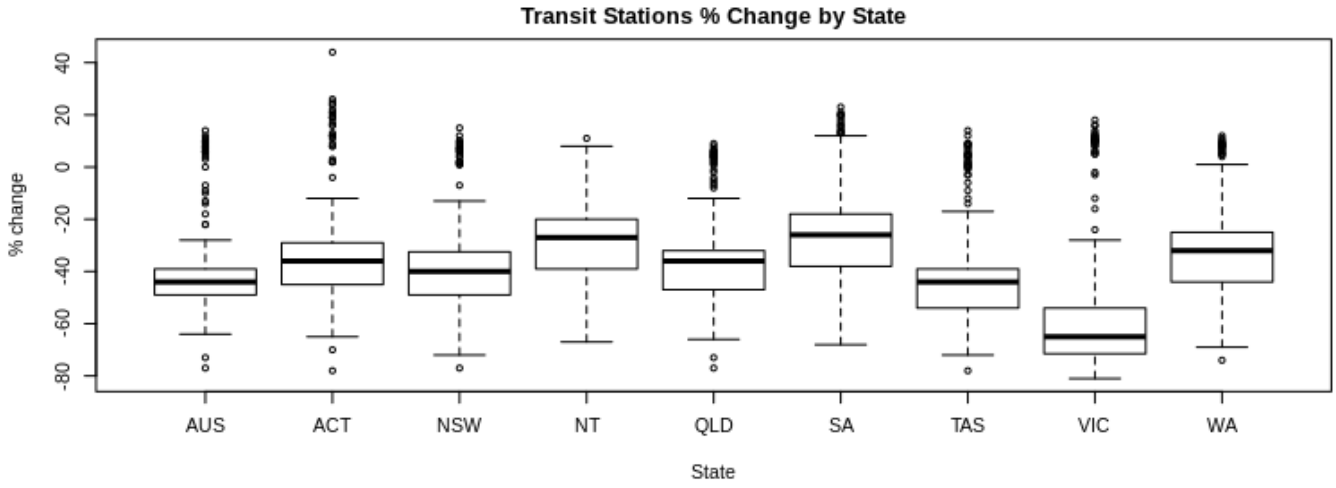
Hide

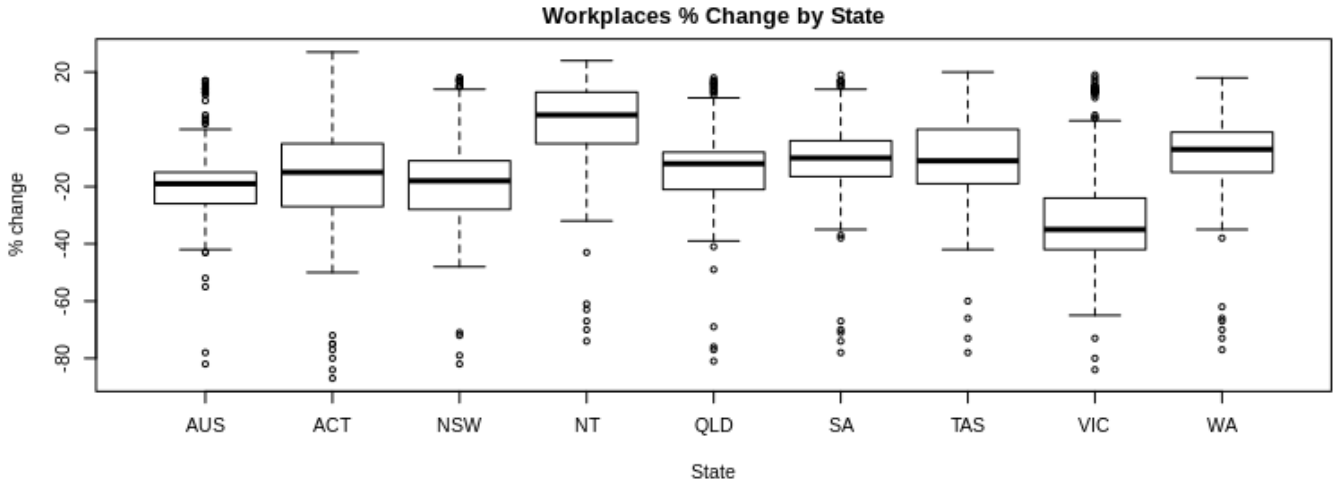
```
boxplot(mobility_AU$Workplaces ~ mobility_AU$State, main="Workplaces % Change by Stat
e",
        ylab = "% change", xlab = "State")
boxplot(mobility_AU$Residential ~ mobility_AU$State, main="Residential % Change by St
ate",
        ylab = "% change",xlab = "State")
```

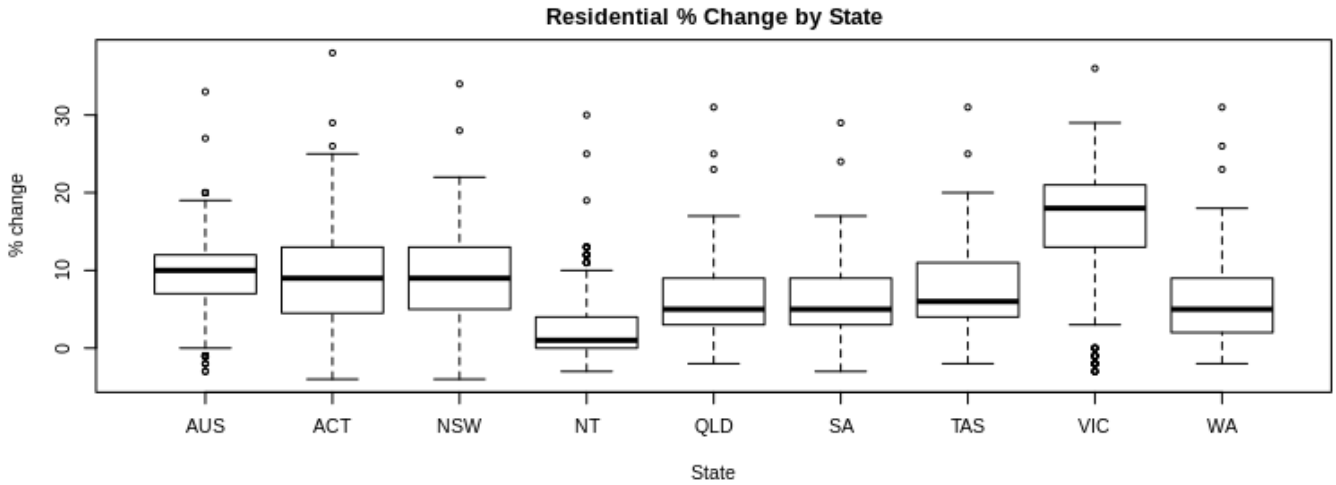












The boxplots show that there are multiple outliers in the numerical variables of this data set. However, there is not enough evidence to support that these outliers are erroneous thus they will be kept to avoid tampering with the data set.

Transformation (Transform)

The objectives of this project does not require these variables to be transformed or normalised.

Merging Datasets

The dimensions of the two data sets should be checked before merging them.

Hide

```
dimMobility <- nrow(mobility_AU)
dimCovid <- nrow(tidy_covid_au)
print(paste0("Number of rows of mobility data set: ", dimMobility))
```

```
[1] "Number of rows of mobility data set: 2124"
```

Hide

```
print(paste0("Number of rows of Covid data set: ", dimCovid))
```

```
[1] "Number of rows of Covid data set: 2124"
```

Hide

```
# Merge on State and Date variables
covid_and_mobility <- left_join(tidy_covid_au,mobility_AU,by=c("State","Date"))
covid_and_mobility %>% head()
```

Date <date>	State <fctr>	Ca… <dbl>	Dea… <dbl>	Daily_Cases <dbl>	Retail_and_Recreation <dbl>	Grocery_and_Pharma <d
2020-02-17	NSW	4	0	0	-2	
2020-02-18	NSW	4	0	0	-3	
2020-02-19	NSW	4	0	0	-2	
2020-02-20	NSW	4	0	0	2	
2020-02-21	NSW	4	0	0	1	
2020-02-22	NSW	4	0	0	4	

6 rows | 1-8 of 11 columns

Conclusion

The final output of this notebook is a data set containing cumulative cases and deaths, daily cases and percent change in mobility of six (6) different place classifications from 17 February to 09 October. Missing and special values have been scanned and replaced using appropriate values. Outliers for numeric variables were also scanned and those found to be erroneous were replaced. Lastly, this data set is also tidy and can be easily filtered by state and date.

References

[1] *Historical Cases, Deaths & Tests by State*, esri Australia, October 14, 2020 [Online]. Available: https://covid19-esriau.hub.arcgis.com/datasets/historical-cases-deaths-tests-by-state (https://covid19-esriau.hub.arcgis.com/datasets/historical-cases-deaths-tests-by-state) (Accessed 14 October 2020)

[2] A. Dolgun PhD. “Module 5 Scan: Missing Values”. http://rare-phoenix-161610.appspot.com/secured/Module_05.html (http://rare-phoenix-161610.appspot.com/secured/Module_05.html) (accessed 20 October 2020)

[3] Government of Western Australia Department of Health. “COVID-19 Update - 4 June 2020”. Government of Western Australia Department of Health. https://ww2.health.wa.gov.au/Media-releases/2020/COVID19-update-4-June-2020 (https://ww2.health.wa.gov.au/Media-releases/2020/COVID19-update-4-June-2020) (accessed 20 October 2020)

[4] Government of Western Australia Department of Health. “COVID-19 Update - 2 August 2020”. Government of Western Australia Department of Health. https://ww2.health.wa.gov.au/Media-releases/2020/COVID-19-update-2-August-2020 (https://ww2.health.wa.gov.au/Media-releases/2020/COVID-19-update-2-August-2020) (accessed 20 October 2020)

[5] Sevi, Semra et al. “Logarithmic versus Linear Visualizations of COVID-19 Cases Do Not Affect Citizens’ Support for Confinement.” Canadian Journal of Political Science. Revue Canadienne De Science Politique 1–6. 14 Apr. 2020, doi:10.1017/S000842392000030X (doi:10.1017/S000842392000030X)

[6] *COVID-19 Community Mobility Reports*, Google, October 14, 2020 [Online]. Available: https://www.gstatic.com/covid19/mobility/Region_Mobility_Report_CSVs.zip (https://www.gstatic.com/covid19/mobility/Region_Mobility_Report_CSVs.zip) (Accessed 14 October 2020)

[7] Google. “Mobility Report CSV Documentation”. COVID-19 Community Mobility Reports. Available: https://www.google.com/covid19/mobility/data_documentation.html?hl=en (https://www.google.com/covid19/mobility/data_documentation.html?hl=en) (Accessed 20 October 2020)