Analysis of COVID Data

Ryan Greenup

May 25, 2020

Contents

§ 1 Introduction

On December 31st 2019 a report regarding a case of a noteworthy viral pneumonia was reported in Wuhan, China, this was later found to be a result of a new strain of virus named *Sars-CoV2*, the disease caused by such an infection, usually resulting in viral pneumonia, is known as *Corona Virus Disease 2019* (*COVID-19*). The outbreak of this disease was declared a Public Health Emergency of International Concern on the 30th January 2020. [25]

A data set detailing the location, deaths, tests and cases related to the *COVID-19* pandemic has been made available through the website *Our World in Data* [14], documented in this report is a visual analyis performed entirely using the *Free Software* [1] ***R*** [19] primarily with the `ggplot2` package [21] (see listing 22 in the appendix).

§ 2 Chloropleth Map

A Chloropleth map of the number of deaths can offer an insight into the impact that the disease has had with respect to individual countries.

The Total deaths should be scaled relative to the population of the country, that way countries with a smaller and sparser population will still be represented by the visualisation (this is quite important given that many countries such as Italy have a small population compared to the US and much of Asia [6]).

A worldwide Chloropleth map visualising the total number of deaths attributed to *COVID-19* is shown in figure 1 and a Europe-centric visualisation is shown in figure 3.

¶ 2 Technical Details

2.2 Preliminary

1. Load Packages and Data

   Before any analysis can be undertaken it is necessary to load the necessary libraries within ***R***, this can be simplified by using a package manager as shown in listing 1.

2. Load the Data

   Next the data set must be loaded into ***R***, this is shown in listing 2.

2.2 Woldwide Map

In order to produce a chloropleth map the data must be aggregated in order to retrieve the total number of deaths, this can be acheived by taking the maximum of the total deaths across countries (the total number of death rates will be a strictly positive and monotone trend, otherwise the outbreak would be an entirely different type of pandemic!), this can be performed by using the `aggregate` function as demonstrated in listing 3.

It is next necessary to rename `location` to `region` so map data will be consistent with the provided data set, this is shown in listing 4.

[1] Free as in speech and beer

R

```
 if (require("pacman")) {
    library(pacman)
  }else{
    install.packages("pacman")
    library(pacman)
  }
  pacman::p_load(xts, sp, gstat, ggplot2, rmarkdown, reshape2, ggmap,
                 parallel, dplyr, plotly, tidyverse, reticulate,
                  ↪  UsingR, Rmpfr,
                 swirl, corrplot, gridExtra, mise, latex2exp,
                  ↪  tidyverse, xts, maptools, plyr, ggplot2, maps,
                  ↪  viridis)

mise()
```

Listing 1: Load the necessary libraries for analysis.

R

```
covid <- read.csv("/home/ryan/Notes/DataSci/Visual_Analytics/Assessment⌋
 ↪  2/owid-covid-data.csv")
```

Listing 2: Load the data into R

R

```
fatalprop <- aggregate(total_deaths_per_million ~ location, covid, max)
## Order the Values in Descending Order
fatalprop <- fatalprop[order(-fatalprop$total_deaths_per_million),]
## Rename USA
covid$location[covid$location=="United States"] <- "USA"
```

Listing 3: Use Aggregate to aggregate total number of deaths

R

```
## Rename to facilitate joining with map
names(fatalprop) <- c("region", "total_deaths_per_million")
```

Listing 4: Rename Features for consistency

For a broad overview of the data, small regions such as San Marino and Belgium will not be visible and will skew the colour pallete, so instead they should be removed and instead a seperate plot of Europe will be created as shown in figure 3, this removal is performed in listing 5.

R

```r
## San Marino will be shown by italy and this skews the results
## Belgium and San Marino are very hard to visualise from above
## They skew the rsults and so will be removed.
fatalprops <- fatalprop %>% filter(region!="San Marino")
fatalprops <- fatalprop %>% filter(region!="Belgium")
```

Listing 5: Filter out small dense regions to prevent scale issues

Next it is necessary to retrieve map data, this can be done using the `map_data` function, this data may then be combined by region with the provided data set using the `left_join` function, this is shown in listing 6.

R

```r
## Retrieve the map data
some_maps <- map_data("world", region = fatalprops$location)

## Join the Data Frames Together
fatalmap <- left_join(fatalprops, some_maps, by = "region")
```

Listing 6: Combine Map Data with Provided Data

Finally this data frame can be plotted by using `ggplot2` and the `geom_map` layer, modifying the `theme` layer will allow for a natural background to be implemented, this is demonstrated in listing 7 and the output is provided in figure 1.

A bubble overlay may also be implemented in order make clearer the spread of cases (see section ¶ 2 for a brief literature review), it is necessary however to adjust the *USA* location to represent the mainland population centre in order make the visualisation more effective. This is demonstrated in listing 8 and shown in figure 2

2.2 Europe Centric

The chloropleth map clearly shows that the disease has caused significiantly more fatalities per capita in Europe and so the plot will be adjusted central to Europe.

As before it is necessary to rename the features of the dataset, however in this instance small European countries such as Belgium should be retained (San marino is a very small italian provice that isn't detectable in the visualisation and skews the pallete, for this reason it will be removed), this is demonstrated in listing 9.

In this map it will be desirable to have labels for the European countries (whereas this would have made the worldwide map too busy), so this will be implemented by using `dyplyr` to generate a second data set as shown in listing 10 which can then be used to generate a plot with the `ggrepel` add on as shown in listing 11, this produces the output shown in figure 3, bubbles were also implemented in order to help visualise the number of relative cases.

R

```r
wmp <- ggplot(fatalmap, aes(map_id = region)) +
  geom_map(map = fatalmap,  color = "grey", aes(fill = total_deaths_per_million), lwd = 0.1, alpha = 0.6)+
  expand_limits(x = fatalmap$long, y = fatalmap$lat)+
  scale_fill_gradient(high = "darkred", low = "white") +
  guides(fill = guide_legend("Total Deaths \n per Million")) +
   # Change the colors of background
   # and the color of grid lines to white
   theme(
     panel.background = element_rect(fill = "lightblue",
                                     colour = "lightblue",
                                     size = 0.5, linetype = "solid"),
     legend.position = c(0.6, 0.1),
     legend.direction = "horizontal",
     legend.background = element_rect(fill = "white", size = 0.1, colour = "darkblue", linetype = "solid")) +
   labs(x = "Longitude", y = "Latitude", title = TeX("Total Deaths Attributed to \\textit{COVID-19}"))
#   geom_text(data = region_lab_df, aes(y = lat, x = long, label = region), size = 1)
wmp
```

Listing 7: use ggplot2 to create a chloropleth map from data, output in figure 1

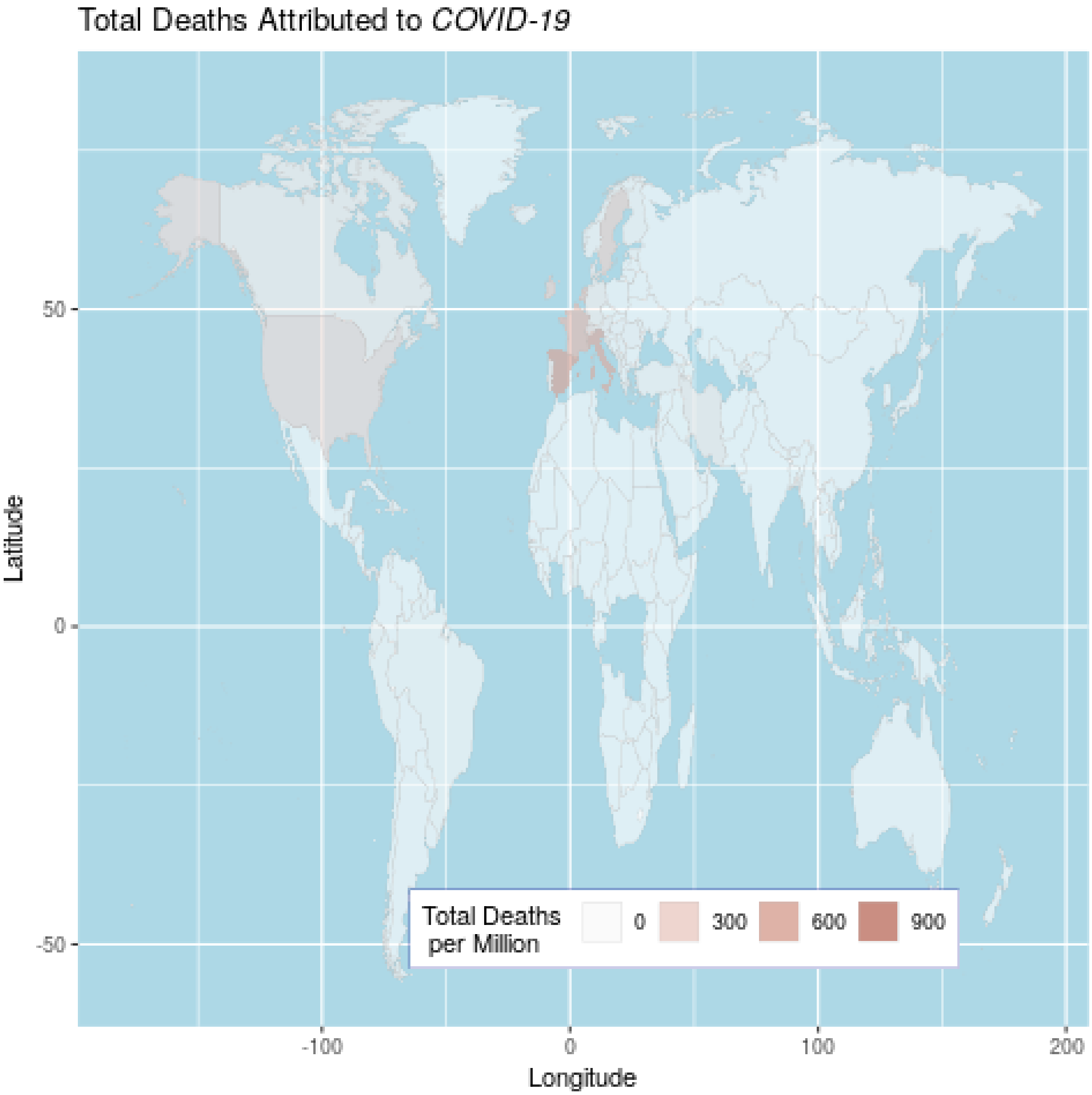

Figure 1: Chloropleth map of total deaths attributed to *COVID-19* (per Million people)

R

```r
# Compute the centroid as the mean longitude and lattitude
# Used as label coordinate for country's names
region_lab_df <- some.eu.maps %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat)) %>%
    full_join(aggregate(total_deaths_per_million ~ region, fatalmap, mean))
# Manually Adjust US to be population Centre
region_lab_df[region_lab_df$region == "USA",]$long <- -92.47
region_lab_df[region_lab_df$region == "USA",]$lat <- 37.37


wmp +
  scale_size_continuous(range = c(1, 9), name = "Total Number \n of Deaths") +
   guides(size = FALSE) +
   geom_point(data = region_lab_df, aes(y = lat, x = long, size = total_deaths_per_million), alpha = 0.5, col = "purple")
```

Listing 8: use ggplot2 to create a chloropleth map from data, output in figure 1

R

```r
## Rename to facilitate joining with map
names(fatalprop) <- c("region", "total_deaths_per_million")

## San Marino will be shown by italy
 fatalprop <- fatalprop %>% filter(region!="San Marino")
```

Listing 9: Rename the features of the data and remove San Marino

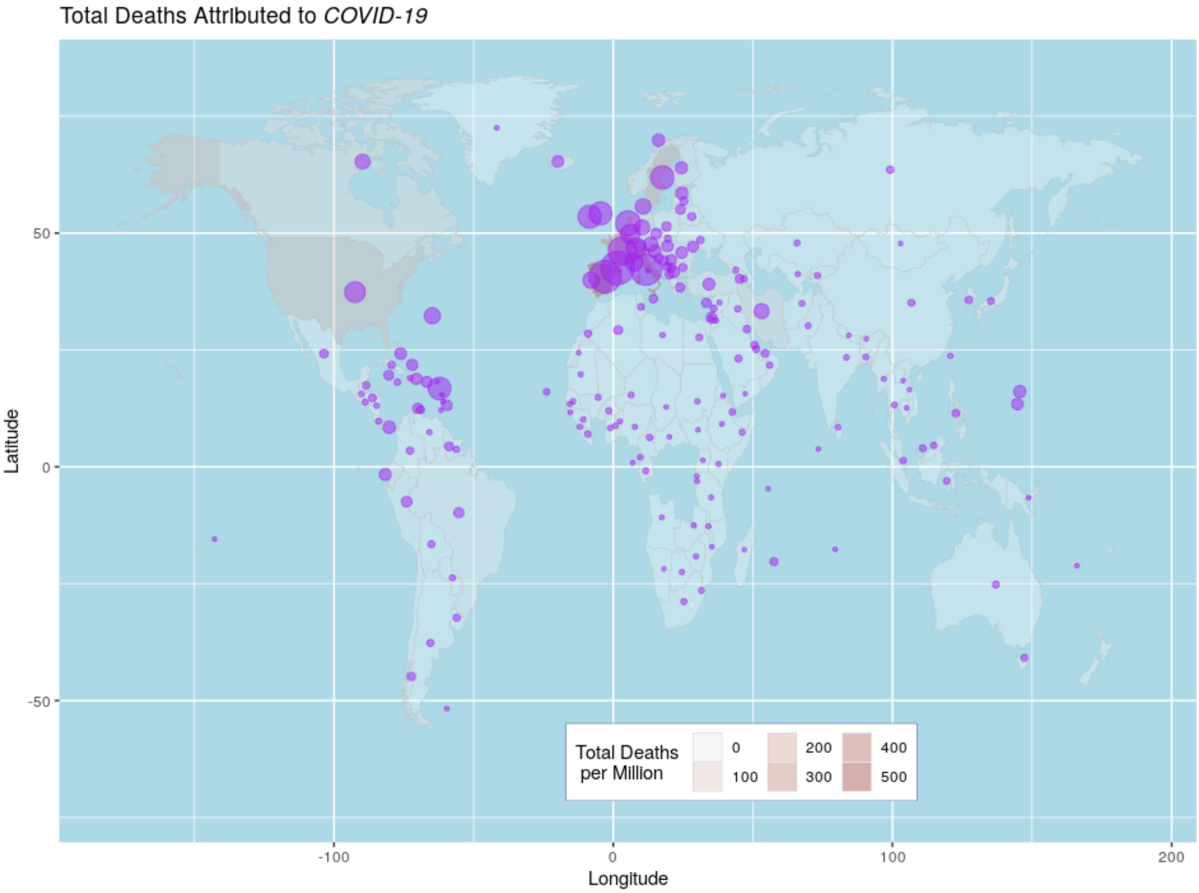

Figure 2: Chloropleth map with bubble overlay to aid in case visualisation

R

```r
fatalmap <- left_join(fatalprop, some.eu.maps, by = "region")

## Filter out only Europe
fatalmap <-  fatalmap %>%
  filter(30 <  lat & lat < 65) %>%
  filter(-30 <  long & long < 35)

## Create Label Data Frame
region_lab_df <- fatalmap %>%
  dplyr::group_by(region) %>%
  dplyr::summarise(long = mean(long), lat = mean(lat)) %>%
   full_join(aggregate(total_deaths_per_million ~ region, fatalmap, mean))
```

Listing 10: use dplyr to reduce the plot size and create a data frame of country labels

R

```r
library(ggrepel)
ggplot(fatalmap, aes(map_id = region, label = region)) +
  geom_map(map = fatalmap,
           aes(fill = total_deaths_per_million),
           color = "white") +
  geom_point(data = region_lab_df, aes(y = lat, x = long, size = total_deaths_per_million), alpha = 0.45, colour = "blue", stroke = 1, fill = "white", shape = 21) +  scale_size_continuous(range = c(1, 25), name = "Total Number \n of Deaths") +
  guides(size = FALSE) +
  expand_limits(x = fatalmap$long, y = fatalmap$lat) +
  scale_fill_viridis_c(option = "C") +
  scale_fill_gradient(high = "darkred", low = "white") +
  guides(fill = guide_legend("Total Deaths \n per Million")) +
  # Change the colors of plot panel background to lightblue
  # and the color of grid lines to white
  theme(
    panel.background = element_rect(
      fill = "lightblue",
      colour = "lightblue",
      size = 0.5,
      linetype = "solid"
    ),
    legend.position = c(0.1, 0.6),
    legend.direction = "vertical",
    legend.background = element_rect(
      fill = "white",
      size =
        1.1,
      colour = "darkblue",
      linetype = "solid"
    )
  ) +
  labs(
    x = "Longitude",
    y = "Latitude",
    title = TeX("Total Deaths Attributed to \\textit{COVID-19}")
  ) +
  geom_text_repel(
    data = region_lab_df,
    aes(y = lat, x = long, label = region),
    size = 2,
    col = "black",
    nudge_y = 0.7,
    nudge_x = -0.5,
    min.segment.length = 0.6,
    force = 2
  )
```

Listing 11: Generate a Chloropleth map centred on Europe using ggplot2

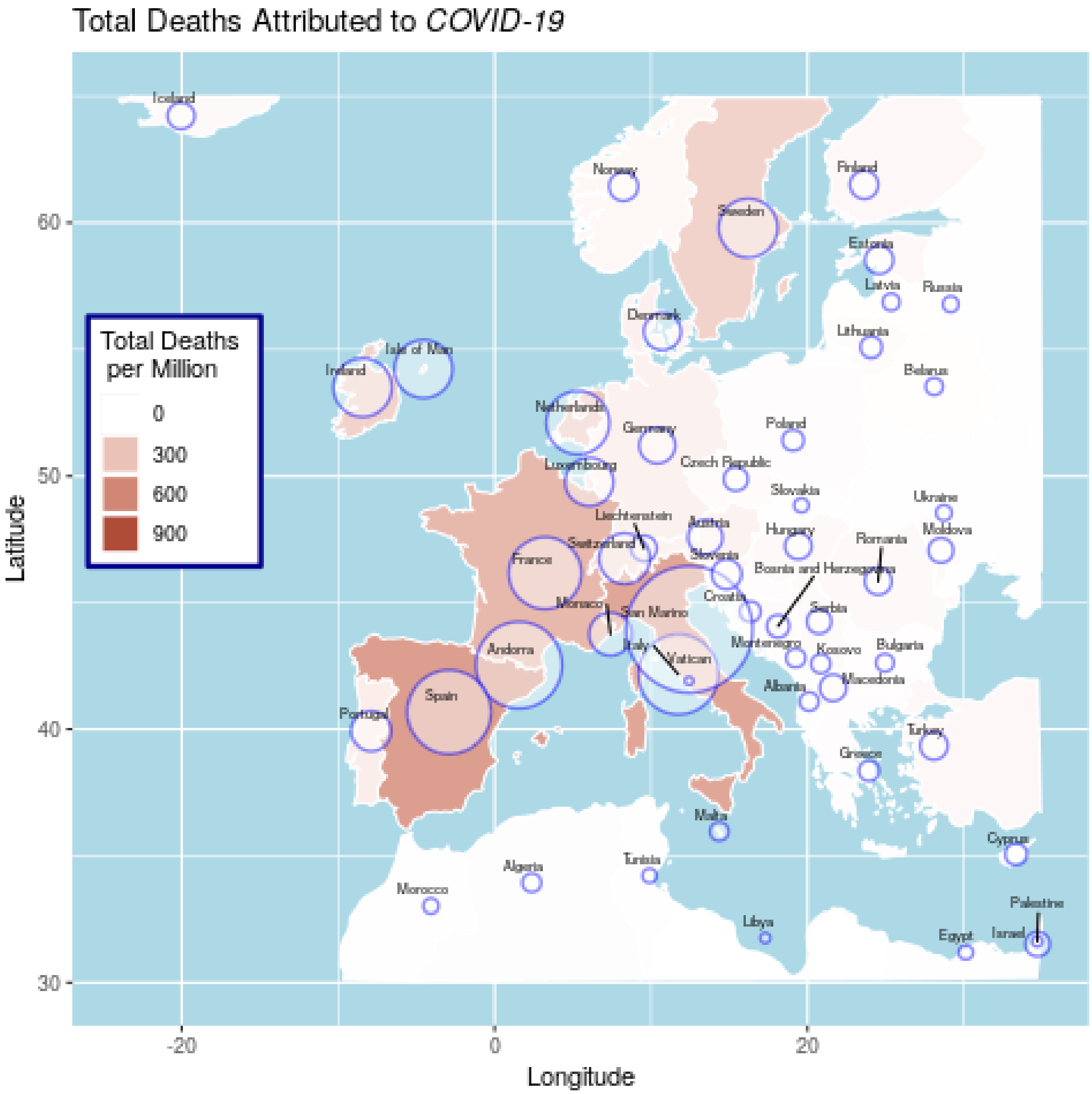

Figure 3: Europe Centred Chloropleth of Deaths Attributed to *COVID-19*

¶ 2 Discussion

2.2 Worldwide

The first plot appears to show a very limited amount of difference in deaths attributable to *COVID-19* across regions other than North America and Europe.

While first-world countries such as New Zealand and Australia are somewhat insulated from the disease by virtue of geography and population density, it's striking that much of Asia and Russia have such low levels of disease incidence.

This could be attributed to the fact that a more power-centric regime such as in China, Russia, North Korea, etc. may have more capacity to:

1. Diminish the spread of the disease by implementing policy decisions,
   (a) whereas countries such as the US and Europe have a much higher expectation of civil liberties and hence much lower tolerance for government intervention.
2. Control the spread of information for want of international reputation.
   (a) In saying that though research suggests that under-reporting has even occured in countries such as the US [17] so such under-reporting could merely be incidental.

A similar disease, *MERS*, emerged in 2012 in Middle-Eastern Regions [24] and a Korean outbreak of the *MERS* disease occured in 2015 [16], these outbreaks likely prepared Korea, the Middle East and other Asian regions for an outbreak which helps explain the dichotomous nature of the deaths attributable to *COVID-19* for those Countries.

2.2 Europe

A closer look at Europe shows that Belgium and Italy have been the most affected by this disease, it isn't very clear why those regions have been impacted so significantly, particularly considering the comparatively permissive borders within the *EU*, but this could be indicative of policy decisions and warrants further research.

¶ 2 Advantages compared to other methods

A Chloropleth map provides a very clear way to visualise the occurence of disease in a geographical sense, in contrast to other methods such as scatter plots, heatmaps and bar charts, the chloropleth map provides a clear way to distinguish the impact of the disease on individual countries.

Chloropleth maps also allow trends across regions to be easily identified, e.g. figure 3 shows how severe the outbreak is in *Europe* relative to other regions, this might be lost in abstraction when using other visualization methods.

The discrete distinction between countries, a fundamental component of a chloropleth map, is desirable because it is consistent with the independent legislatures accross countries, this allows for a comparison of the impact that policy decisions may or may not have on a region.

¶ 2 Disasadvantages

When maps are projected into a 2D plane they are necessarily distorted, this distortion can impact how spread the data appears to be.

A chloropleth map can make it hard to compare metrics between to regions in any specific sense, for this a more appropriate visualization could be for instance a bar chart.

¶ 2 Literature review of related work

The *John Hopkins Coronavirus Dashboard* [2] implemented bubbles to visualise the number of cases, a screenshot of this is provided in the appendix at figure 7, this was a part of the motivation for implementing bubbles in the chloropleth map because the visualization was so much more *striking* and promoted pre-attentive processing of the information.

In his blog, Kenneth Field produced chloropleth and bubble-map charts detailing the spread of *COVID-19*, with however, a focuse on China, [7] these plots were very similar to those produced in this report, however the legend for the bubble plot was very nicely implemented and can be seen in figure 8 of the appendix. He also produced an example illustrating why the use of a heatmap or contour map can make for a poor visualisation of cases due to the difficulty in interpreting the visualization compared to a bubble chart, for this reason a bubble chart was used in this report and a heatmap was not implemented.

A paper in the publication *Environment & Planning A* suggested using a cartogram to visualise the spread of disease, there example is provided in figure 9 of the appendix. [8] Although the cartogram is visually quite appealing and easy to read, it is difficult to interpret quickly, the visualisation does not promote pre-attentive processing, for this reason the visualisation strategy was not implemented.

§ 3 Time Series

¶ 3 Implementation

Time series charts can be an effective way to visualise the behaviour of a value over time, for this dataset however, two modifications will be implemented in order to make the trends more distinct.

3.3 Log Scale

The spread of disease over time can often be described by an exponential model as demonstrated in equations (1) and (2), for this reason the use of a $\log$ -scale will linearise trends and so the use of a $\log$ -scale will make it easier to compare the rates of population change between different countries.

$$\frac{\mathrm{d}p}{\mathrm{d}t} \propto p \implies p = Ce^{kt} \quad \exists k, c \in \mathbb{R} \tag{1}$$

$$\frac{\mathrm{d}p}{\mathrm{d}t} \propto p \wedge \frac{\mathrm{d}p}{\mathrm{d}t} \propto (N-p) \implies p = \frac{ke^{Nt}}{1-ke^{Nt}} \quad \exists k \in \mathbb{R}, N \in \mathbb{R}^+ \tag{2}$$

3.3 Adjust Zero

In addition to a $\log -$ scale, *sliding* the data to be relative to the number of days since the first case can allow the trends of the data to be compared, A similar technique was implemented by *John Hopkins University* in a visualisation published in *The Guardian* [10]. Here the number of cases has been considered from the date of the first case, however, figure 4 shows the trend from the date of the 100th case, while it appears to line the countries up better the loss of information was undesirable.

¶ 3 Technical Details

3.3 Preliminary

In order to log scale the data the `mutate` function from the `dplyr` package was used on data transformed into *wide* format by using the `pivot_wider` function, this is shown in listing 12.

Sliding the date back to the number of cases however was a little more difficult and required the use of a `for` loop to iterate the `lead` function over each column (where each column, after transformation with `dplyr`, represented the value for a country), this is demonstrated in listing 12 with an example of the produced *tidy* data provided in table 1; the code to produce the plot is demonstrated in listing 13, the output of which is provided in figure 4.

Rather than using a line plot or a scatter plot, a `loess` model was placed ontop of semi-opaque points, this is to enhance the continuity of the visualisation. The *Gestalt Laws* provide that continuous shapes are easier for readers to interpret [18] and for this reason the the overlay was implemented, to aid the reader in delineating between the different countries in a plot.

Plots with many colours mapped to categorical variables can be difficult to interpret [23, 15], for this reason less than 10 countries were compared on the same plot.

Table 1: Top few rows of the *tidy* data set created from listing 12.

Date	***Location***	***Total Cases Per Million***
1	South Korea	0.193
1	Italy	0.116
1	Australia	0.00860
1	Germany	0.122
1	United Kingdom	0.0976
1	USA	0.00903
1	Russia	0.00303
2	South Korea	0.480
2	Italy	0.339
2	Australia	0.0558

3.3 Facet Grid

This plot however does not show all the data made available, the data set also includes information on the number of tests,cases and deaths resulting from *COVID-19*, in order to visualise this the `fact_grid` layer can be used to create a multi-scatterplot. first it is necessary to create a data frame, this can be implemented by repeating the process in listing 12 for each different metric but it will also be necessary to add a feature corresponding to that metric's description, we will also create non-log scaled data as

R

```r
cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]

# interested_locations <- c("Australia", "USA", "Italy", "Germany", "Belgium", "United Kingdom", "New Zealand", "Japan", "China")
interested_locations <- c("Australia", "USA", "Italy", "Germany", "Russia", "South Korea", "United Kingdom")

cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases_per_million > 1) %>%
  mutate(total_cases_per_million = log10(total_cases_per_million)) %>%
  dplyr::select(date, total_cases_per_million, location) %>%
  pivot_wider(names_from = location, values_from = total_cases_per_million)


for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}

cv <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to = "total_cases_per_million")
```

Listing 12: Use dplyr to transform the data as shown in table 1, this can then be passed to ggplot as shown in listing 13

R

```
ggplot(cv , aes(y = total_cases_per_million, x = date, col = location, group = location)) +
  geom_point(alpha = 0.3)  +
  geom_smooth() +
  theme_bw() +
  labs(y = "Total Number of Cases (Log-10 Scale)", title = "Log Scaled Total COVID-19 Cases per Million", x = TeX("Days since Case \\textit{#100}")) +
  guides(col = guide_legend("Location"))
#  geom_smooth()
```

Listing 13: Use `dplyr` to transform the data before plotting with `ggplot`

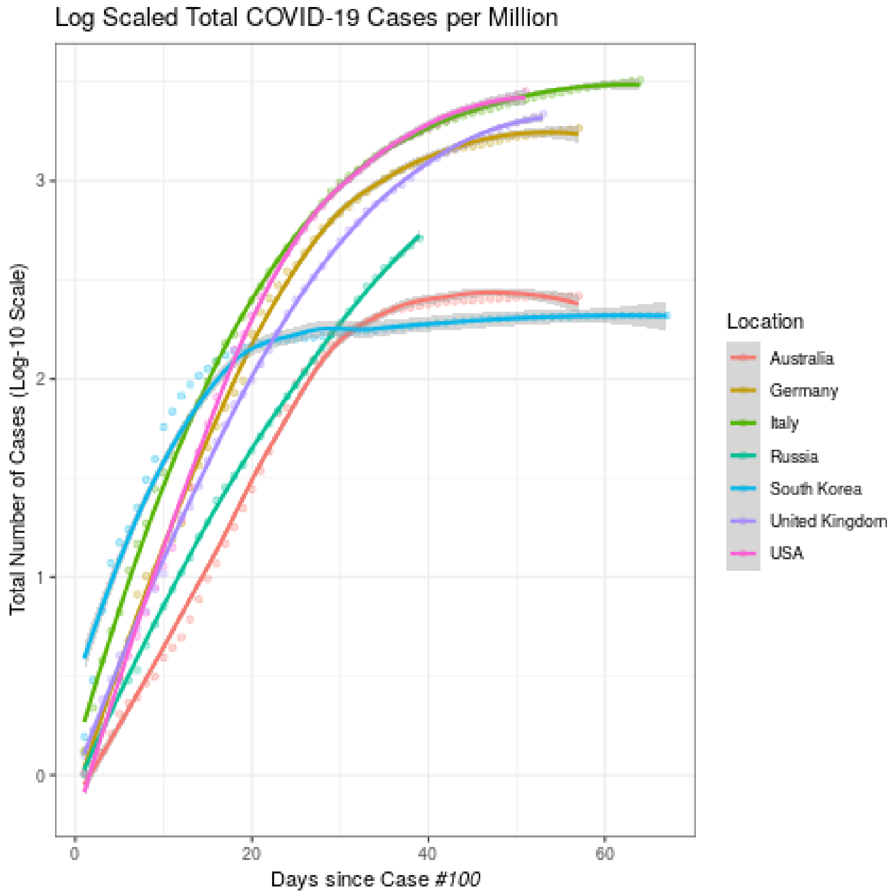

Figure 4: Chloropleth map of total deaths attributed to *COVID-19* (per Million people)

well, this is demonstrated in listings 14 through 19, finally the dataframes are merged in listing 20, the corresponding plot is shown in figure 5.

R

```r
interested_locations <- c("Australia", "USA", "Italy", "Germany", "Russia", "South Korea", "United Kingdom")

###### Number of Cases
cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]

cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
  mutate(total_cases_per_million = log10(total_cases_per_million)) %>%
  dplyr::select(date, total_cases_per_million, location) %>%
  pivot_wider(names_from = location, values_from = total_cases_per_million)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
          is.na() %>%
          sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}

cv_cases_log <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to = "value") %>%
  add_column(subject = "No. of Cases") %>%
  add_column(scale = "Log-10 Scale")
```

Listing 14: Use `dplyr` to create a data frame of log scaled cases

¶ 3 Advantages compared to other methods

- The advantage to a log-scaled plot is that it allows rates of change to be compared between countries

R

```r
### Number of deaths

cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]

cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
   mutate(total_deaths_per_million = log10(total_deaths_per_million))
    ↪  %>%
  dplyr::select(date, total_deaths_per_million, location) %>%
  pivot_wider(names_from = location, values_from =
   ↪  total_deaths_per_million)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}

cv_deaths_log <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to =
  ↪  "value") %>%
  add_column(subject = "No. of Deaths") %>%
  add_column(scale = "Log-10 Scale")
```

Listing 15: Use dplyr to create a data frame of log scaled deaths

R

```r
### Number of Tests
cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]
cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
  mutate(total_tests_per_thousand = log10(total_tests_per_thousand)-3) %>%
  dplyr::select(date, total_tests_per_thousand, location) %>%
  pivot_wider(names_from = location, values_from = total_tests_per_thousand)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}
cv_tests_log <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to = "value") %>%
  add_column(subject = "No. of Tests") %>%
  add_column(scale = "Log-10")

cv <- rbind(cv_cases_log, cv_deaths_log, cv_tests_log)
cv %>%
  filter(subject == "deaths")

p_per_cap <- ggplot(cv , aes(y = value, x = date)) +
  geom_point(alpha = 0.3, aes(col = location))  +
   geom_smooth(aes(col = location), size = 0.5) +
  theme_bw() +
  labs(y = TeX("Count (log_{10} Scale)"), title = TeX("log_{10} Scale; Value of \\textit{COVID-19} Statistics over Time"), x = TeX("Days since Case \\textit{#1}"), subtitle = "Counts Per Million of population") +
  guides(col = guide_legend("Location")) +
  facet_grid(rows = vars(subject), scales = "free_y")
p_per_cap
```

Listing 16: Use `dplyr` to create a data frame of log scaled deaths, observe thousands is scaled to millions.

```r
R
interested_locations <- c("Australia", "USA", "Italy", "Germany",
 ↪  "Russia", "South Korea", "United Kingdom")

###### Number of Cases
cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]

cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
# mutate(total_cases = log10(total_cases)) %>%
  dplyr::select(date, total_cases_per_million, location) %>%
  pivot_wider(names_from = location, values_from =
   ↪  total_cases_per_million)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}

cv_cases_raw <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to =
  ↪  "value") %>%
  add_column(subject = "No. of Cases") %>%
  add_column(scale = "Count")
```

Listing 17: use dplyr to create a data frame of non-log scaled cases

R

```r
### Number of deaths

cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]

cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
#  mutate(total_deaths_per_million = log10(total_deaths_per_million_))
↪  %>%
  dplyr::select(date, total_deaths_per_million, location) %>%
  pivot_wider(names_from = location, values_from =
  ↪  total_deaths_per_million)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}

cv_deaths_raw <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to =
 ↪  "value") %>%
  add_column(subject = "No. of Deaths") %>%
  add_column(scale = "Count")
```

Listing 18: use `dplyr` to create a data frame of non-log scaled deaths

R

```r
### Number of Tests
cv <- as_tibble(covid)
cv <- cv %>%
  mutate(date = as.Date(date))
cv <- cv[order(cv$date),]
cv <- cv %>%
  filter(location %in% interested_locations) %>%
  filter(total_cases > 1) %>%
 # mutate(total_tests_per_thousandd = log10(total_tests_per_thousand)) %>%
  mutate(total_tests_per_thousandd = total_tests_per_thousand/1000) %>%
  dplyr::select(date, total_tests_per_thousand, location) %>%
  pivot_wider(names_from = location, values_from = total_tests_per_thousand)

for (i in 2:ncol(cv)) {
  ## Slide the Columns up and put the NA at the end
cv[,i] <-   pull(cv, i) %>%
  lead(cv[,i] %>%
         is.na() %>%
         sum())
 ## Replace the date with the number of days
cv$date <- seq_len(nrow(cv))
}
cv_tests_raw <- cv %>%
 pivot_longer(names(cv)[-1], names_to = "location", values_to = "value") %>%
  add_column(subject = "No. of Tests") %>%
  add_column(scale = "Count")
cv <- rbind(cv_cases_raw, cv_deaths_raw, cv_tests_raw)
cv %>%
  filter(subject == "deaths")

p_total <- ggplot(cv , aes(y = value, x = date)) +
  geom_point(alpha = 0.3, aes(col = location))  +
   geom_smooth(aes(col = location), size = 0.5) +
  theme_bw() +
  labs(y = TeX("Total Count"), title = TeX("Total Count of \\textit{COVID-19} Statistics over Time"), x = TeX("Days since Case \\textit{#1}")) +
  guides(col = guide_legend("Location"), subtitle = "Per Million of Population") +
  facet_grid(rows = vars(subject), scales = "free_y")
p_total
```

Listing 19: use dplyr to create a data frame of non-log scaled tests

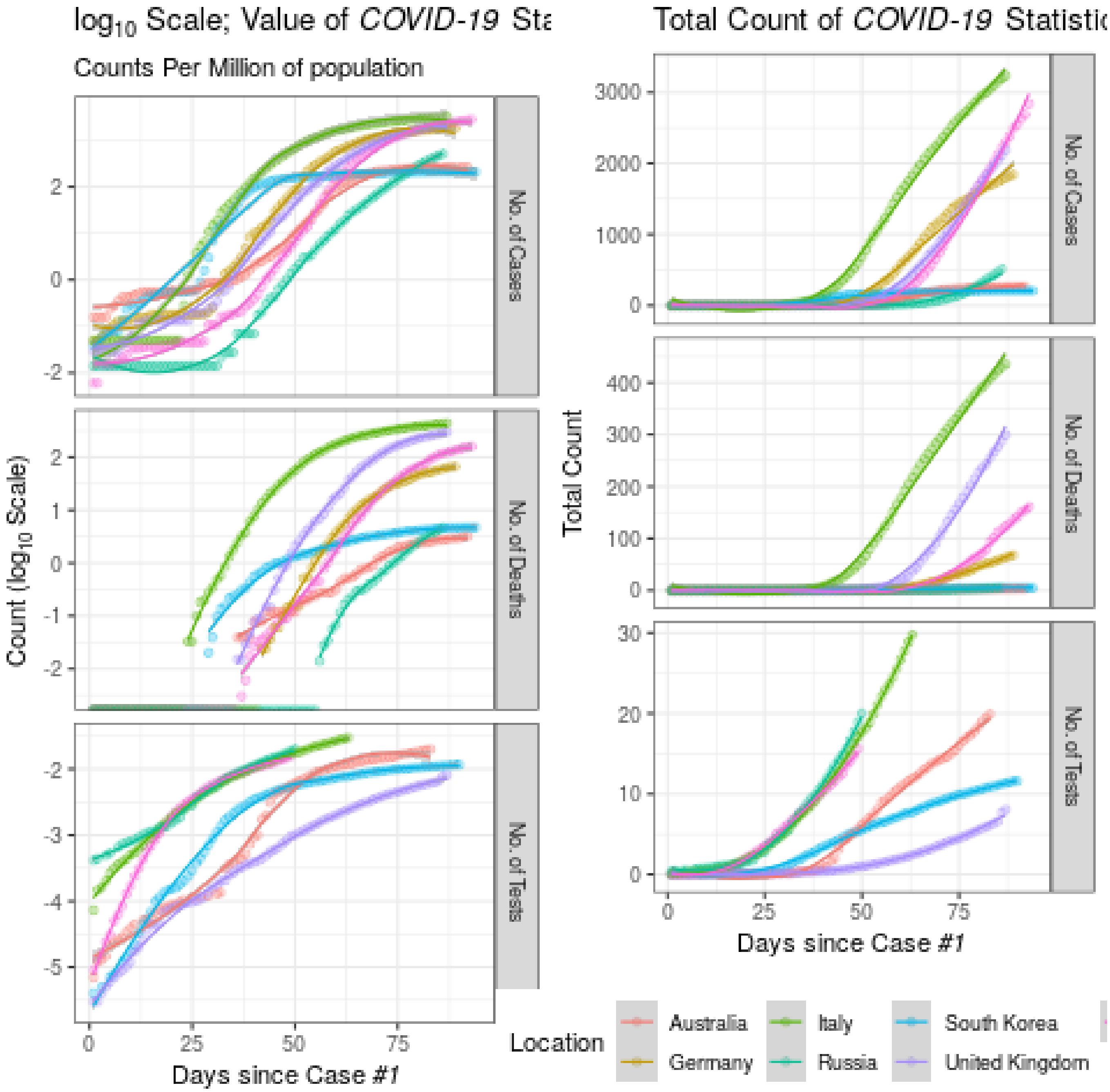

Figure 5: Multi Scatter Plot of *COVID-19* Metrics.

```r
plots <- list(p_per_cap + guides(col = FALSE), p_total+ theme(legend.position="bottom") )
# plots <- list(p_per_cap + theme(legend.position="bottom"), p_total+ theme(legend.position="bottom") )
library(gridExtra)

gridExtra::grid.arrange(grobs = plots, layout_matrix = matrix(1:2, nrow = 1))
```

Listing 20: Merge the plots in order to create a single visualisation

- Making the Data Relative to the day of the first infection allows individual countries to be compared in terms of there response

¶ 3 Disasadvantages

- A log-scaled plot can be misleading if it is not made clear, this is particularly true for readers who have limited mathematical training.
  - For this reason a plot without log-scaling was included and the axis were labelled accordingly
- Making Data relative to the day of the first infection may not make clear that certain countries had *forewarning* of the disease by virtue of the delay.

¶ 3 Discussion on analysis results

Although the plots have been adjusted to reflect the date that the first cases were observed, it is possible that the disease began spreading before the first official case was reported, this is a belief held by some health officials in Italy.[9]

3.3 Number of Cases

This plot clearly suggests that the spread of the disease was the greatest both in rate and magnitude in Italy, some researchers belive that this is due simply to the fact that Italy has performed more tests. [9]

3.3 Number of Deaths

Italy has had the highest amount of deaths despite it's higher rates of testing, it is not clear why this is the case though. A study by *IQAir* found that 25% of the most air-polluted European countries were located within Italy [11] and such pollution has been found to be correlated with higher rates of death resulting from viral respiratory infection, [1, 3, 26] this could help explain some of the discrepancy but more research into the unique vulnerabilities of Italy is certainly warranted.

The Visualisation suggests that Russia has had the fewest number of deaths related to *COVID-19*, however Moscow's Mayor, Sergei Sobyanin suggested that the official number of infections is likely much lower than reality, a sentiment echoed by Russia's *Doctor's Alliance* (which is essentially a doctors union). [5]

Dispensing with The view that Russia's figures are reliable it is clear that Australia and South Korea have the lowest number of cases overall, while Australia's success can be attributed to it's relative isolation and unique quarantine requirements, [4] combined with lockdown's implemented early in the pandemic (with respect to Australia's first case) [22] the success of South Korea Appears to be related more appropriatley with the aggressive action taken by the country to contact trace the spread of the disease. [20]

It appears that the number of deaths is more closely correlated with the number of cases than the number of tests, however it is not clear what the effect of testing is on the number of new cases.

3.3 Number of Tests

The visualisation also suggests that Italy and the US have undertaken the highest rates of testing, per capita, this however does not appear to have influeced the rates of death or spread of cases significantly, indicating that measuring a countries response to the disease cannot be meaured merely by considering the rate of testing.

¶ 3 Discussion on other Aspects

- A potential improvement to this visualisation would be to plot many countries, say 30 but greyscale those countries and only apply colour to countries of interest, this would provide background information relative to those observations but not overwhelm the reader, this is a suggestion made by Andy Kirk in his *Visualising Data* blog [12].

¶ 3 Literature review of related work

As mentioned in section 3.3 the use of the log-scaled and date-adjusted plot was implemented by *John Hopkins University* in a visualisation published in *The Guardian* newspaper [10].

NSW Health created a visualisation of cases acquired over time using a barchart in a way that resembles a histogram, [13] this plot is very easy to interpret and clearly demonstrates the success of NSW in *flattening the curve*, this visualisation could have been implemented for this data as demonstrated in listing 21 and shown in figure 6 for different countries in a similar fashion, this however was not effective for comparing countries and so was not pursued.

§ 4 Appendix

PROPERTIES: :ID: 84c19d03-8ab7-4793-a86d-e861e1bffe2b

§ 5 References

R

```
#+begin_src
interested_locations <- c("Australia", "USA", "Italy", "Germany",
↪  "Russia", "South Korea", "United Kingdom")
cv <- covid %>%
  dplyr::filter(location %in% interested_locations)

ggplot(fortify(cv), aes(x = as.Date(date), y = new_cases_per_million,
↪  fill = location)) +
  geom_col(col = "grey") +
  labs(x = "Date", y = "New Cases Per Million") +
   theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  theme_bw()
```

Listing 21: Use ggplot to create a bar chart

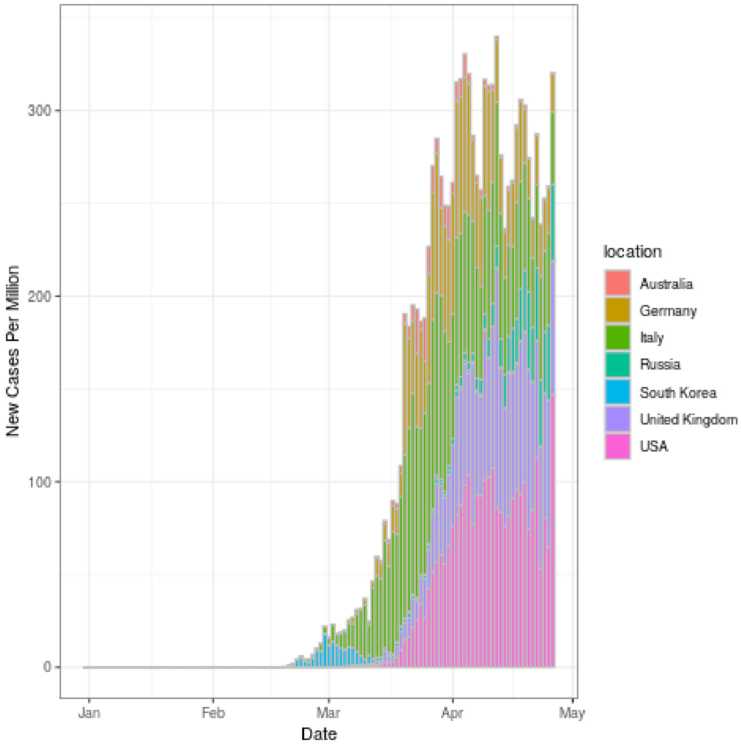

Figure 6: Bar Chart of cases over time for various locations

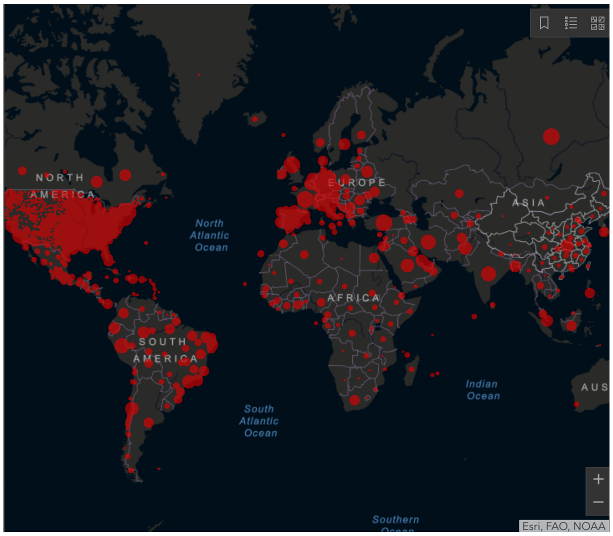

Figure 7: John Hopkins Bubble Chart [2]

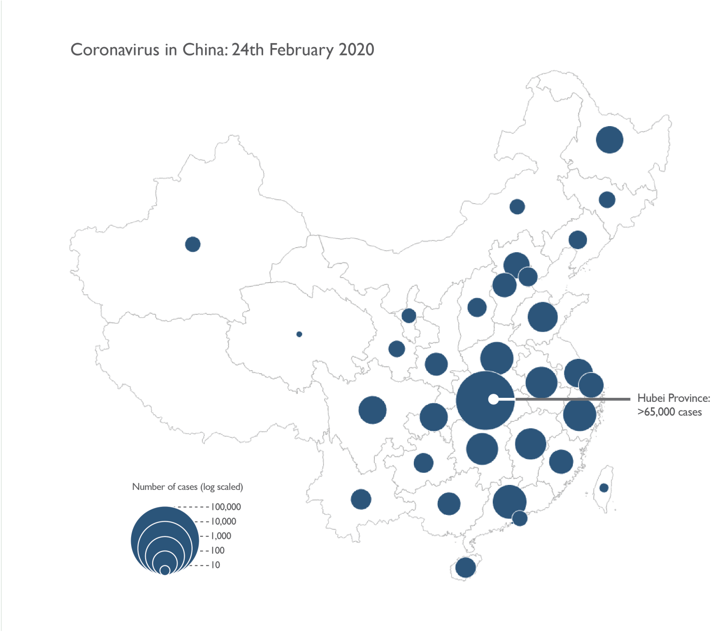
Coronavirus in China: 24th February 2020
Hubei Province:
>65,000 cases
Number of cases (log scaled)
100,000
10,000
1,000
100
10

Figure 8: Bubble Plot Chart produced by Field in his blog [7]

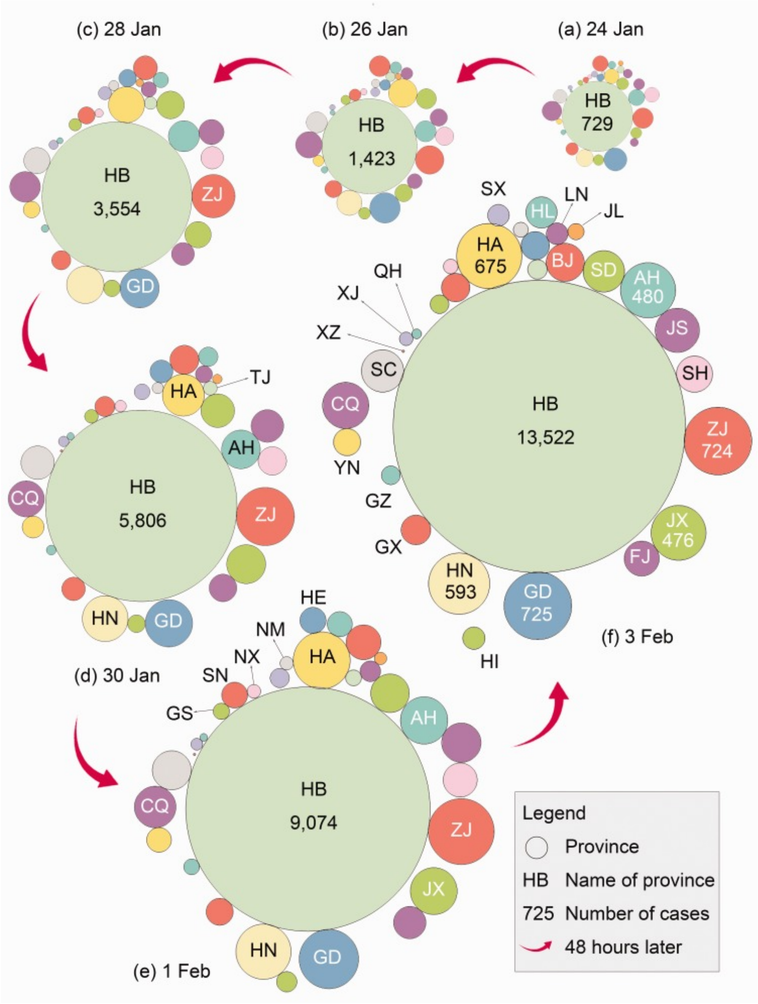

Figure 9: Cartogram of *COVID-19* spread [8]

R

```
citation()
citation("ggplot2")


##  To cite R in publications use:
##
##    R Core Team (2020). R: A language and environment for statistical
##    computing. R Foundation for Statistical Computing, Vienna, Austria.
##    URL https://www.R-project.org/.
##
##  A BibTeX entry for LaTeX users is
##
##    @Manual{,
##      title = {R: A Language and Environment for Statistical Computing},
##      author = {{R Core Team}},
##      organization = {R Foundation for Statistical Computing},
##      address = {Vienna, Austria},
##      year = {2020},
##      url = {https://www.R-project.org/},
##    }
##
##  We have invested a lot of time and effort in creating R, please cite it
##  when using it for data analysis. See also citation("pkgname") for
##  citing R packages.
##
##  To cite ggplot2 in publications, please use:
##
##    H. Wickham. ggplot2: Elegant Graphics for Data Analysis.
##    Springer-Verlag New York, 2016.
##
##  A BibTeX entry for LaTeX users is
##
##    @Book{,
##      author = {Hadley Wickham},
##      title = {ggplot2: Elegant Graphics for Data Analysis},
##      publisher = {Springer-Verlag New York},
##      year = {2016},
##      isbn = {978-3-319-24277-4},
##      url = {https://ggplot2.tidyverse.org},
##    }
```

Listing 22: Generate Citation for ***R*** programming Language

References

[1] Jonathan Ciencewicki and Ilona Jaspers. “Air Pollution and Respiratory Viral Infection”. eng. In: *Inhalation Toxicology* 19.14 (Nov. 2007), pp. 1135–1146. ISSN: 1091-7691. DOI: 10.1080/08958370701665434 (cit. on p. 22).

[2] *COVID-19 Map*. en. May 2020. URL: https://coronavirus.jhu.edu/map.html (visited on 05/21/2020) (cit. on pp. 11, 25).

[3] Daniel P. Croft et al. “The Association between Respiratory Infection and Air Pollution in the Setting of Air Quality Policy and Economic Change”. In: *Annals of the American Thoracic Society* 16.3 (Mar. 2019), pp. 321–330. ISSN: 2329-6933. DOI: 10.1513/AnnalsATS.201810-691OC. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6394122/ (visited on 05/24/2020) (cit. on p. 22).

[4] Department of Agrigulture, Water and the Environment, Australia. *Fact Sheet - Travelling or Returning to Australia - Department of Agriculture*. Aug. 2019. URL: https://www.agriculture.gov.au/travelling/travel-agent-resources/factsheet-travelling-returning (visited on 05/24/2020) (cit. on p. 23).

[5] Dole. *If Coronavirus Explodes in Russia, Putin's Political Survival Could Be on the Line*. en-AU. Apr. 2020. URL: https://www.abc.net.au/news/2020-04-05/coronavirus-is-russia-lying-about-its-infection-rate/12118056 (visited on 05/24/2020) (cit. on p. 23).

[6] *Europe :: Italy The World Factbook - Central Intelligence Agency*. May 2020. URL: https://www.cia.gov/library/publications/the-world-factbook/geos/it.html (visited on 05/18/2020) (cit. on p. 1).

[7] Kenneth Field. *Mapping Coronavirus, Responsibly*. en-US. Feb. 2020. URL: https://www.esri.com/arcgis-blog/products/product/mapping/mapping-coronavirus-responsibly/ (visited on 05/21/2020) (cit. on pp. 11, 25).

[8] Peichao Gao et al. “Visualising the Expansion and Spread of Coronavirus Disease 2019 by Cartograms”. In: *Environment & Planning a* (Feb. 2020). ISSN: 0308-518X. DOI: 10.1177/0308518X20910162. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7140974/ (visited on 05/21/2020) (cit. on pp. 11, 26).

[9] Mellissa Godin. *Why Is the Coronavirus Outbreak So Bad in Italy?* en. Mar. 2020. URL: https://time.com/5799586/italy-coronavirus-outbreak/ (visited on 05/24/2020) (cit. on p. 22).

[10] Pablo Gutiérrez. “Coronavirus World Map: Which Countries Have the Most Cases and Deaths?” en-GB. In: *The Guardian* (May 2020). ISSN: 0261-3077. URL: https://www.theguardian.com/world/2020/may/20/coronavirus-world-map-which-countries-have-the-most-cases-and-deaths (visited on 05/21/2020) (cit. on pp. 12, 23).

[11] IQAir. *World's Most Polluted Cities in 2019 - PM2.5 Ranking | AirVisual*. en. 2019. URL: https://www.airvisual.com/world-most-polluted-cities (visited on 05/24/2020) (cit. on p. 22).

[12] Andy Kirk. *Make Grey Your Best Friend*. Jan. 2015. URL: https://www.visualisingdata.com/2015/01/make-grey-best-friend/ (visited on 05/21/2020) (cit. on p. 23).

[13] NSW Health. *NSW COVID-19 Case Statistics - COVID-19 (Coronavirus)*. May 2020. URL: https://www.health.nsw.gov.au/Infectious/covid-19/Pages/stats-nsw.aspx (visited on 05/21/2020) (cit. on p. 23).

[14] Hannah Ritchie. *Coronavirus Source Data*. May 2020. URL: https://ourworldindata.org/coronavirus-source-data (visited on 05/21/2020) (cit. on p. 1).

[15] Lisa Rost. *Choosing Colors for Data Visualization*. en-US. Aug. 2018. URL: https://www.dataquest.io/blog/what-to-consider-when-choosing-colors-for-data-visualization/ (visited on 05/21/2020) (cit. on p. 12).

[16] Ruel Serrano. *Intensified Public Health Measures Help Control MERS-CoV Outbreak in the Republic of Korea*. en. World Health Organization. July 2015. URL: https://www.who.int/westernpacific/news/detail/28-07-2015-intensified-public-health-measures-help-control-mers-cov-outbreak-in-the-republic-of-korea (visited on 05/18/2020) (cit. on p. 10).

[17] Neeraj Sood et al. "Seroprevalence of SARS-CoV-2Specific Antibodies Among Adults in Los Angeles County, California, on April 10-11, 2020". en. In: *JAMA* (May 2020). DOI: 10.1001/jama.2020.8279. URL: https://jamanetwork.com/journals/jama/fullarticle/2766367 (visited on 05/19/2020) (cit. on p. 10).

[18] Markus R. Staudinger et al. "Gestalt Perception and the Decline of Global Precedence in Older Subjects". en. In: *Cortex* 47.7 (July 2011), pp. 854–862. ISSN: 0010-9452. DOI: 10.1016/j.cortex.2010.08.001. URL: http://www.sciencedirect.com/science/article/pii/S0010945210002170 (visited on 05/21/2020) (cit. on p. 12).

[19] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2020. URL: http://www.R-project.org/ (cit. on p. 1).

[20] Derek Thompson. *What's Behind South Korea's COVID-19 Exceptionalism?* en-US. May 2020. URL: https://www.theatlantic.com/ideas/archive/2020/05/whats-south-koreas-secret/611215/ (visited on 05/24/2020) (cit. on p. 23).

[21] Hadley Wickham. *Ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag, 2016. ISBN: 978-3-319-24277-4. URL: https://ggplot2.tidyverse.org (cit. on p. 1).

[22] Olivia Willis. *The Coronavirus Emergency Plan Has Been Activated. Here's What That Means for You*. en-AU. Feb. 2020. URL: https://www.abc.net.au/news/health/2020-02-28/what-coronavirus-emergency-plan-means-for-you/12010056 (visited on 05/24/2020) (cit. on p. 23).

[23] Alan Wilson. *The Power of The Palette: Why Color Is Key in Data Visualization and How to Use It*. en-US. Feb. 2017. URL: https://theblog.adobe.com/the-power-of-the-palette-why-color-is-key-in-data-visualization-and-how-to-use-it/ (visited on 05/21/2020) (cit. on p. 12).

[24] Matt Woodley. "How Does Coronavirus Compare with Previous Global Outbreaks?" In: *Royal Australian College of General Practitioners* (Feb. 2020) (cit. on p. 10).

[25] World Health Organization. *Coronavirus Disease (COVID-19) - Events as They Happen*. en. May 2020. URL: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/events-as-they-happen (visited on 05/21/2020) (cit. on p. 1).

[26] Dandan Zhang et al. "The Relationship between Air Quality and Respiratory Pathogens among Children in Suzhou City". In: *Italian Journal of Pediatrics* 45.1 (Sept. 2019), p. 123. ISSN: 1824-7288. DOI: 10.1186/s13052-019-0702-2. URL: https://doi.org/10.1186/s13052-019-0702-2 (visited on 05/24/2020) (cit. on p. 22).