COVID-19 Data Analysis with R - Worldwide

Yanchang Zhao
yanchang@RDataMining.com
http://RDataMining.com

26 March 2020

Contents

1 Introduction

This is an analysis report of the Novel Coronavirus (COVID-19) around the world, to demonstrate data processing and visualisation with R, *tidyverse* and *ggplot2*. This report will be updated from time to time, with new data and more analysis. Please find its latest version at http://www.rdatamining.com/docs/Coronavirus-data-analysis-world.pdf.

A similar COVID-19 analysis report for China is available at http://www.rdatamining.com/docs/Coronavirus-data-analysis-china.pdf, if you are particually interested what has happened in China.

1.1 Data Source

The data source used for this analysis is *the 2019 Novel Coronavirus COVID-19 (2019-nCoV) Data Repository*[1] built by the Center for Systems Science and Engineering, Johns Hopkins University.

1.2 R Packages

Blow is a list of R packages used for this analysis. Package *magrittr* is for pipe operations like `%>%` and `%<>%` and *lubridate* for date operations. Package *tidyverse* is a collection of R packages for data science, including *dplyr* and *tidyr* for data processing and *ggplot2* for graphics. Package *gridExtra* is for arranging multiple grid-based plots on a page and *kableExtra* works together with `kable()` from *knitr* to build complex HTML or LaTeX tables.

```
library(magrittr) # pipe operations
library(lubridate) # date operations
library(tidyverse) # ggplot2, tidyr, dplyr...
library(gridExtra) # multiple grid-based plots on a page
library(ggforce) # accelerating ggplot2
library(kableExtra) # complex tables
library(leaflet) # map
```

2 Loading Data

At first, the datasets, which are three CSV files, are downloaded and saved as local files and then are loaded into R.

```
## source data files
filenames <- c('time_series_covid19_confirmed_global.csv',
               'time_series_covid19_deaths_global.csv',
               'time_series_covid19_recovered_global.csv')
url.path <- paste0('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/',
                   'master/csse_covid_19_data/csse_covid_19_time_series/')

## download files to local
download <- function(filename) {
  url <- file.path(url.path, filename)
  dest <- file.path('./data', filename)
  download.file(url, dest)
}
bin <- lapply(filenames, download)

## load data into R
raw.data.confirmed <- read.csv('./data/time_series_covid19_confirmed_global.csv')
raw.data.deaths <- read.csv('./data/time_series_covid19_deaths_global.csv')
raw.data.recovered <- read.csv('./data/time_series_covid19_recovered_global.csv')

dim(raw.data.confirmed)
```

```
## [1] 245  68
```

Each dataset has 245 rows, corresponding to country/region/province/state. It has 68 columns. Starting from column 5, each column corresponds to a single day. Here we have a look at the first 10 rows and the first 10 columns.

[1] https://github.com/CSSEGISandData/COVID-19

```
raw.data.confirmed[1:10, 1:10] %>%
  kable('latex', booktabs=T, caption='Raw Data (Confirmed, First 10 Columns only)') %>%
  kable_styling(font_size=6, latex_options = c('striped', 'hold_position', 'repeat_header'))
```

Table 1: Raw Data (Confirmed, First 10 Columns only)

Province.State	Country.Region	Lat	Long	X1.22.20	X1.23.20	X1.24.20	X1.25.20	X1.26.20	X1.27.20
	Afghanistan	33.0000	65.0000	0	0	0	0	0	0
	Albania	41.1533	20.1683	0	0	0	0	0	0
	Algeria	28.0339	1.6596	0	0	0	0	0	0
	Andorra	42.5063	1.5218	0	0	0	0	0	0
	Angola	-11.2027	17.8739	0	0	0	0	0	0
	Antigua and Barbuda	17.0608	-61.7964	0	0	0	0	0	0
	Argentina	-38.4161	-63.6167	0	0	0	0	0	0
	Armenia	40.0691	45.0382	0	0	0	0	0	0
Australian Capital Territory	Australia	-35.4735	149.0124	0	0	0	0	0	0
New South Wales	Australia	-33.8688	151.2093	0	0	0	0	3	4

Below we check the time frame of the data.

```
n.col <- ncol(raw.data.confirmed)
## get dates from column names
dates <- names(raw.data.confirmed)[5:n.col] %>% substr(2,8) %>% mdy()
range(dates)
```

```
## [1] "2020-01-22" "2020-03-25"
```

```
min.date <- min(dates)
max.date <- max(dates)
min.date.txt <- min.date %>% format('%d %b %Y')
max.date.txt <- max.date %>% format('%d %b %Y')
```

It shows that the data was last updated on 25 Mar 2020 UTC and all the stats and charts in this report are based on that data.

3 Data Preparation

3.1 Data Cleaning

The three datesets are converted from wide to long format and then are aggregated by country. After that, they are merged into one single dataset.

```
## data cleaning and transformation
cleanData <- function(data) {
  ## remove some columns
  data %<>% select(-c(Province.State, Lat, Long)) %>% rename(country=Country.Region)
  ## convert from wide to long format
  data %<>% gather(key=date, value=count, -country)
  ## convert from character to date
  data %<>% mutate(date = date %>% substr(2,8) %>% mdy())
  ## aggregate by country
  data %<>% group_by(country, date) %>% summarise(count=sum(count, na.rm=T)) %>% as.data.frame()
  return(data)
}

## clean the three datasets
data.confirmed <- raw.data.confirmed %>% cleanData() %>% rename(confirmed=count)
data.deaths <- raw.data.deaths %>% cleanData() %>% rename(deaths=count)
```

```r
data.recovered <- raw.data.recovered %>% cleanData() %>% rename(recovered=count)

## merge above 3 datasets into one, by country and date
data <- data.confirmed %>% merge(data.deaths, all.x=T) %>% merge(data.recovered, all.x=T)
# data %<>% mutate(recovered = ifelse(is.na(recovered), lag(recovered, 1), recovered))

## countries/regions with confirmed cases, excl. cruise ships
countries <- data %>% pull(country) %>% setdiff('Cruise Ship')

## first 10 records when it first broke out in China
data %>% filter(country=='China') %>% head(10) %>%
  kable('latex', booktabs=T, caption='Raw Data (with first 10 Columns Only)',
        format.args=list(big.mark=',')) %>%
  kable_styling(latex_options = c('striped', 'hold_position', 'repeat_header'))
```

Table 2: Raw Data (with first 10 Columns Only)

country	date	confirmed	deaths	recovered
China	2020-01-22	548	17	28
China	2020-01-23	643	18	30
China	2020-01-24	920	26	36
China	2020-01-25	1,406	42	39
China	2020-01-26	2,075	56	49
China	2020-01-27	2,877	82	58
China	2020-01-28	5,509	131	101
China	2020-01-29	6,087	133	120
China	2020-01-30	8,141	171	135
China	2020-01-31	9,802	213	214

There are 174 countries with confirmed COVID-19 cases, as of 25 Mar 2020 UTC.

3.2 Worldwide Cases

The raw data provide the daily number of cases in every country. They are aggregated below to derive the daily stats of the whole world.

```r
## counts for the whole world
data.world <- data %>% group_by(date) %>%
  summarise(country='World',
            confirmed = sum(confirmed, na.rm=T),
            deaths = sum(deaths, na.rm=T),
            recovered = sum(recovered, na.rm=T))

data %<>% rbind(data.world)

## current confirmed cases
data %<>% mutate(current.confirmed = confirmed - deaths - recovered)
```

3.3 Daily Increases and Death Rates

After that, the daily increases of death and recovered cases and the death rates are calculated.

`rate.upper` is caculated with the total dead and recovered cases. It is the upper bound of death rate and the reasons are

1) there were much more deaths than recovered cases when the coronavirus broke out and when it was not contained, and
2) the daily number of death will decrease and that of recovered will increase as it becomes contained and more effective measures and treatments are used.

`rate.lower` is caculated with total dead and confirmed cases. It is a lower bound of death rate, because there are and will be new deaths from the current confirmed cases. The final death rate is expected to be in between of the above two rates.

`rate.daily` is caculated with the daily dead and recovered cases and therefore is more volatile than the above two. However, it can give us a clue of the current situlation: whether it is very serious or is getting better.

```
## sort by country and date
data %<>% arrange(country, date)

## daily increases of deaths and recovered cases
## set NA to the increases on day1
n <- nrow(data)
day1 <- min(data$date)
data %<>% mutate(new.confirmed = ifelse(date == day1, NA, confirmed - lag(confirmed, n=1)),
                 new.deaths = ifelse(date == day1, NA, deaths - lag(deaths, n=1)),
                 new.recovered = ifelse(date == day1, NA, recovered - lag(recovered, n=1)))

## change negative number of new cases to zero
data %<>% mutate(new.confirmed = ifelse(new.confirmed < 0, 0, new.confirmed),
                 new.deaths = ifelse(new.deaths < 0, 0, new.deaths),
                 new.recovered = ifelse(new.recovered < 0, 0, new.recovered))

## death rate based on total deaths and recovered cases
data %<>% mutate(rate.upper = (100 * deaths / (deaths + recovered)) %>% round(1))
## lower bound: death rate based on total confirmed cases
data %<>% mutate(rate.lower = (100 * deaths / confirmed) %>% round(1))
## death rate based on the number of death/recovered on every single day
data %<>% mutate(rate.daily = (100 * new.deaths / (new.deaths + new.recovered)) %>% round(1))


## convert from wide to long format, for drawing area plots
data.long <- data %>%
  select(c(country, date, confirmed, current.confirmed, recovered, deaths)) %>%
  gather(key=type, value=count, -c(country, date))
## set factor levels to show them in a desirable order
data.long %<>% mutate(type=recode_factor(type, confirmed='Total Confirmed',
                                         current.confirmed='Current Confirmed',
                                         recovered='Recovered',
                                         deaths='Deaths'))


## convert from wide to long format, for drawing area plots
rates.long <- data %>%
  # filter(country %in% top.countries) %>%
  select(c(country, date, rate.upper, rate.lower, rate.daily)) %>%
  # mutate(country=factor(country, levels=top.countries)) %>%
  gather(key=type, value=count, -c(country, date))
# set factor levels to show them in a desirable order
```

```
rates.long %<>% mutate(type=recode_factor(type, rate.daily='Daily',
                                          rate.lower='Lower bound',
                                          rate.upper='Upper bound'))
```

4 Worldwide Cases

After tidying up the data, we visualise it with various charts.

4.1 World Map

Below is a world map of vconfirmed cases. An interactive map can be created if running the code in R or RStudio, or knitting it into a HTML file.

```
## select last column, which is the number of latest confirmed cases
x <- raw.data.confirmed
x$confirmed <- x[, ncol(x)]
x %<>% select(c(Country.Region, Province.State, Lat, Long, confirmed)) %>%
  mutate(txt=paste0(Country.Region, ' - ', Province.State, ': ', confirmed))

m <- leaflet(width=1200, height=800) %>% addTiles()
# circle marker (units in pixels)
m %<>% addCircleMarkers(x$Long, x$Lat,
                        radius=2+log2(x$confirmed), stroke=F,
                        color='red', fillOpacity=0.3,
                        popup=x$txt)

# world
m
```

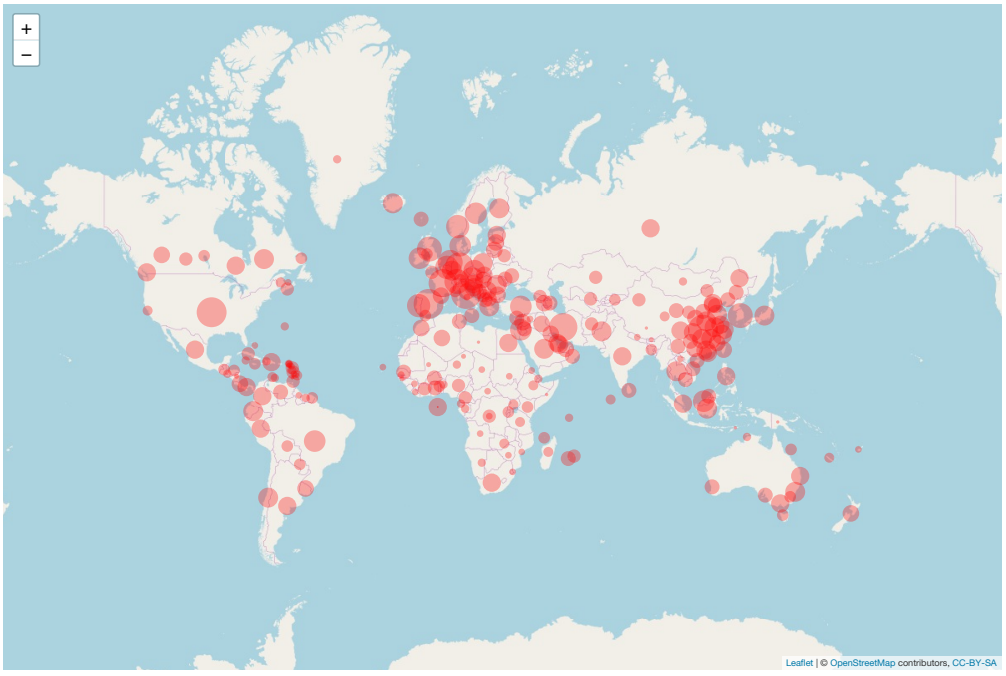


Figure 1: World Map

Views of some specific countries or regions can be produced with the script below.

```
## China
m %>% setView(95, 35, zoom=4)
## Australia and New Zealand
m %>% setView(135, -27, zoom=4)
## US and Canada
m %>% setView(-105, 40, zoom=4)
## Europe
m %>% setView(10, 50, zoom=4)
```

4.2 Number of Cases

In the rest of this section, we will focuse on the cases worldwide. Similar analysis for a single country can be done by filter the data with the corresponding country name.

```
# data %<>% filter(country=='China')
# data %<>% filter(country=='Australia')
world.long <- data.long %>% filter(country == 'World')

## cases - area plot
plot1 <- world.long %>% filter(type != 'Total Confirmed') %>%
  ggplot(aes(x=date, y=count)) +
  geom_area(aes(fill=type), alpha=0.5) +
  labs(title=paste0('Numbers of Cases Worldwide - ', max.date.txt)) +
  scale_fill_manual(values=c('red', 'green', 'black')) +
  theme(legend.title=element_blank(), legend.position='bottom',
        plot.title = element_text(size=8),
        axis.title.x=element_blank(),
        axis.title.y=element_blank(),
        legend.key.size=unit(0.2, 'cm'),
        legend.text=element_text(size=6),
        axis.text=element_text(size=7),
        axis.text.x=element_text(angle=45, hjust=1))

plot2 <- world.long %>%
  ggplot(aes(x=date, y=count)) +
  geom_line(aes(color=type)) +
  labs(title=paste0('Numbers of Cases Worldwide (log scale) - ', max.date.txt)) +
  scale_color_manual(values=c('purple', 'red', 'green', 'black')) +
  theme(legend.title=element_blank(), legend.position='bottom',
        plot.title = element_text(size=8),
        axis.title.x=element_blank(),
        axis.title.y=element_blank(),
        legend.key.size=unit(0.2, 'cm'),
        legend.text=element_text(size=6),
        axis.text=element_text(size=7),
        axis.text.x=element_text(angle=45, hjust=1)) +
  scale_y_continuous(trans='log10')

grid.arrange(plot1, plot2, ncol=2)
```

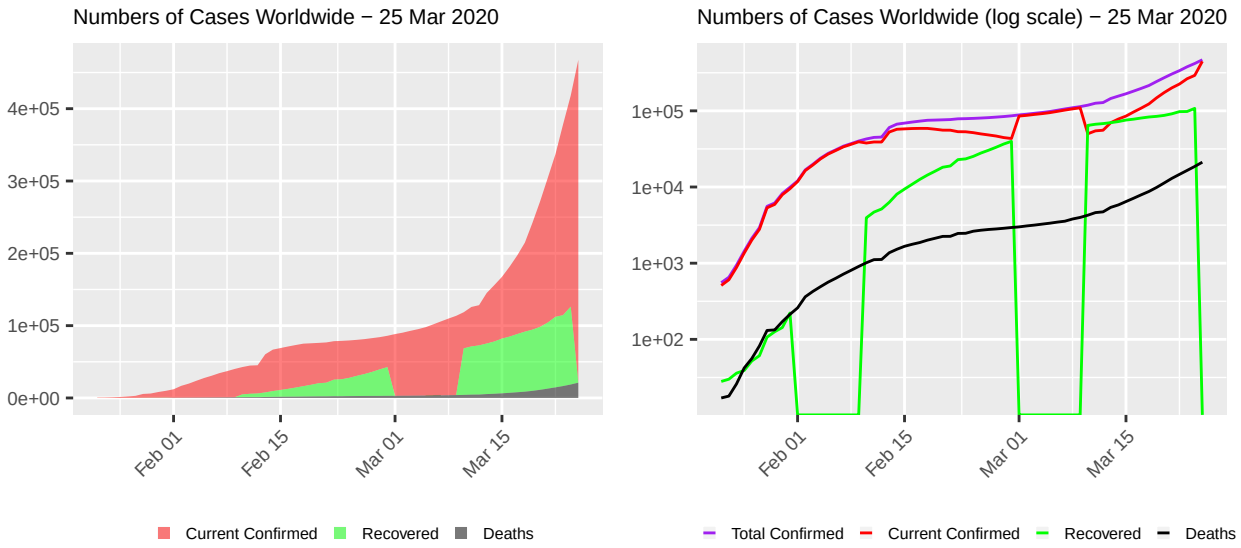


Figure 2: COVID-19 Cases Worldwide

4.3 Current Confirmed Cases

```
data.world <- data %>% filter(country=='World')
n <- nrow(data.world)
```

```
## current confirmed and daily new confirmed
plot1 <- ggplot(data.world, aes(x=date, y=current.confirmed)) +
  geom_point() + geom_smooth() +
  xlab('') + ylab('Count') + labs(title='Current Confirmed Cases') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
plot2 <- ggplot(data.world, aes(x=date, y=new.confirmed)) +
  geom_point() + geom_smooth() +
  xlab('') + ylab('Count') + labs(title='Daily New Confirmed Cases') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
## show two plots side by side
grid.arrange(plot1, plot2, ncol=2)
```

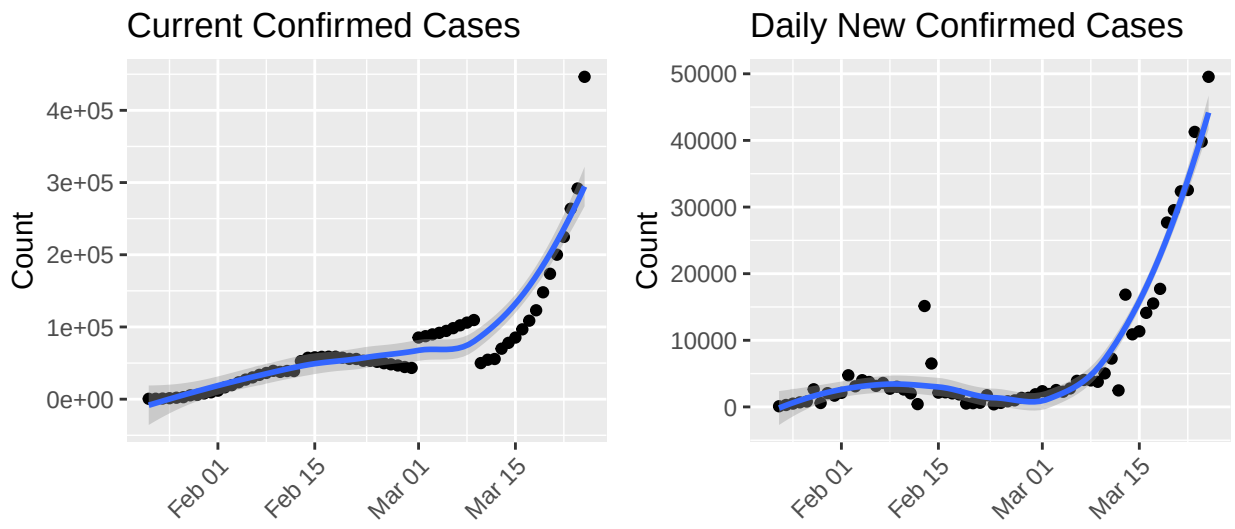


Figure 3: Current Confirmed Cases

Figure 3 shows the numbers of current (see left chart) and new (see right chart) confirmed cases. The blue lines are smoothed conditional means and the grey band around them show the 95% confidence interval.

4.4 Deaths and Recovered Cases

```
## a scatter plot with a smoothed line and vertical x-axis labels
plot1 <- ggplot(data.world, aes(x=date, y=deaths)) +
  geom_point() + geom_smooth() +
  xlab('') + ylab('Count') + labs(title='Accumulative Deaths') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
plot2 <- ggplot(data.world, aes(x=date, y=recovered)) +
  geom_point() + geom_smooth() +
  xlab('') + ylab('Count') + labs(title='Accumulative Recovered Cases') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
plot3 <- ggplot(data.world, aes(x=date, y=new.deaths))  +
  geom_point() + geom_smooth()  +
  xlab('') + ylab('Count') + labs(title='New Deaths') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
plot4 <- ggplot(data.world, aes(x=date, y=new.recovered)) +
  geom_point() + geom_smooth()  +
  xlab('') + ylab('Count') + labs(title='New Recovered Cases') +
  theme(axis.text.x=element_text(angle=45, hjust=1))
## show four plots together, with 2 plots in each row
grid.arrange(plot1, plot2, plot3, plot4, nrow=2)
```

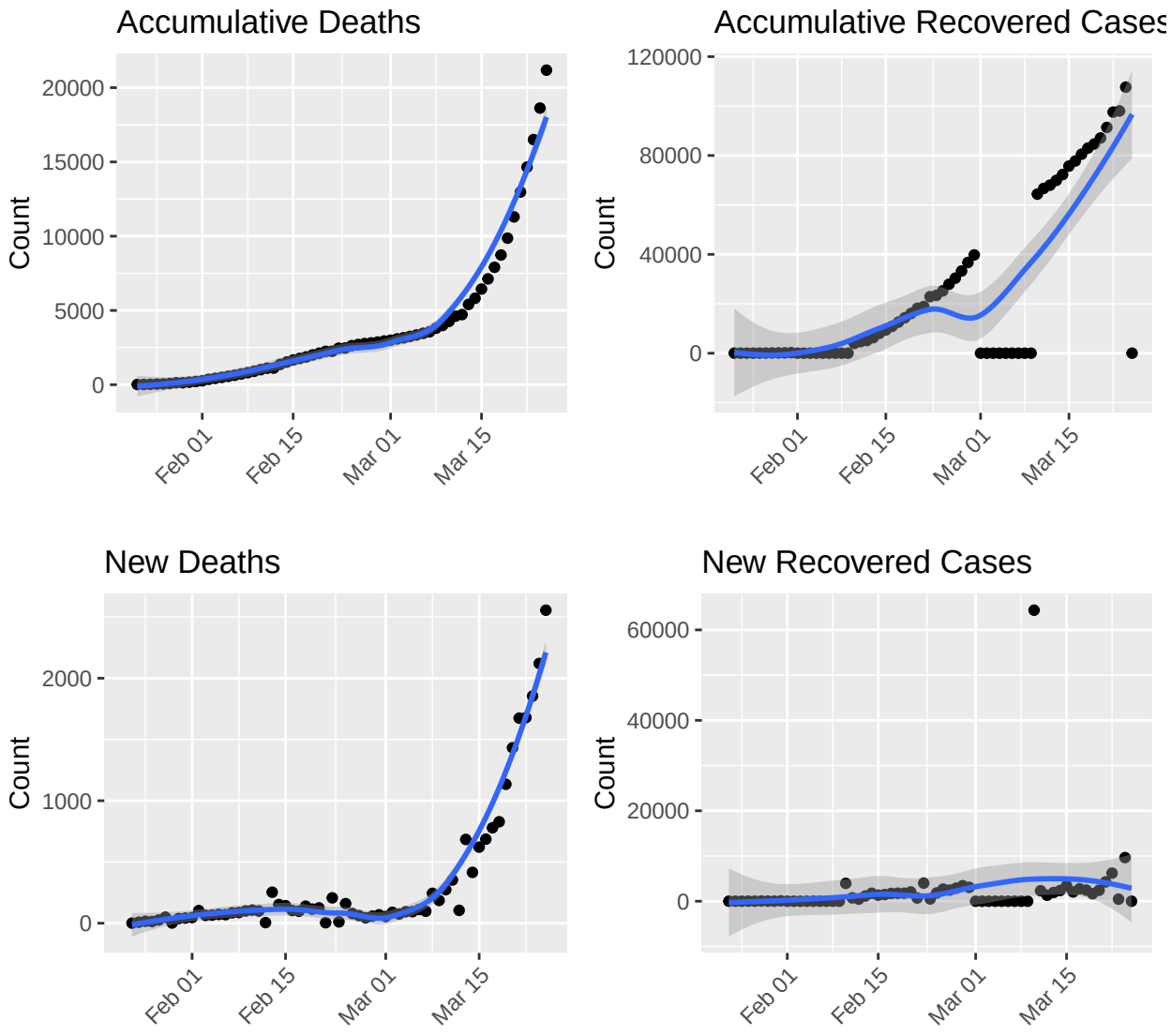


Figure 4: Deaths and Recovered Cases

4.5 Death Rates

Figure 5 shows death rates caculated in three different ways (see Section 3.3 for details). The left chart shows the death rates from 22 Jan 2020 to 25 Mar 2020 and the right one is a zoom-in view of the rates in last two weeks.

In the right chart, the upper bound (in blue) is decreasing, as there will be more recovered cases and fewer dead ones daily as time goes on. However, the lower bound (in green) keeps going up, as there are and will be new deaths from the current confirmed cases. Therefore, the final death rate is expected to be in-between of those two rates, and based on the latest data retrieved as of 25 Mar 2020 UTC, it will be between 4.5% and 100%.

A surge in the daily death rate (in red) suggests that the situlation is changing dramatically (actually, getting worse) and that above lower/upper bounds are likely to increase shortly. A likely reason of that surge is the recent outbreak of coronavirus in Italy, Iran and some other European countries.

```
## three death rates
plot1 <- ggplot(data.world, aes(x=date)) +
```

```
  geom_line(aes(y=rate.upper, colour='Upper bound')) +
  geom_line(aes(y=rate.lower, colour='Lower bound')) +
  geom_line(aes(y=rate.daily, colour='Daily')) +
  xlab('') + ylab('Death Rate (%)') + labs(title='Overall') +
  theme(legend.position='bottom', legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'),
        axis.text.x=element_text(angle=45, hjust=1))
## focusing on last 2 weeks
y.max <- data.world[n-(14:0), ] %>% select(rate.upper, rate.lower, rate.daily) %>% max()
plot2 <- ggplot(data.world[n-(14:0),], aes(x=date)) +
  geom_line(aes(y=rate.upper, colour='Upper bound')) +
  geom_line(aes(y=rate.lower, colour='Lower bound')) +
  geom_line(aes(y=rate.daily, colour='Daily')) +
  xlab('') + ylab('Death Rate (%)') + labs(title='Last two weeks') +
  theme(legend.position='bottom', legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'),
        axis.text.x=element_text(angle=45, hjust=1)) +
  ylim(c(0, y.max))
grid.arrange(plot1, plot2, ncol=2)
```

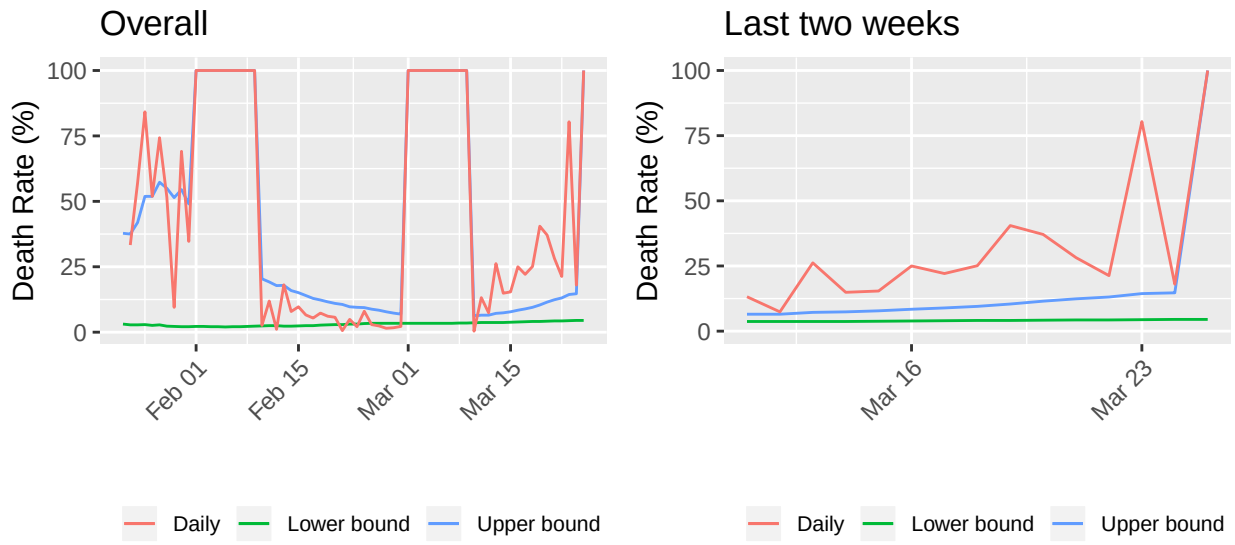


Figure 5: Death Rate

5 Top Twenty Countries

Next, we will have a look at the top 20 countries in total confirmed cases.

```
## ranking by confirmed cases
data.latest.all <- data %>% filter(date == max(date)) %>%
  select(country, date,
         confirmed, new.confirmed, current.confirmed,
         recovered, deaths, new.deaths, death.rate=rate.lower) %>%
  mutate(ranking = dense_rank(desc(confirmed)))
```

```
k <- 20
## top 20 countries: 21 incl. 'World'
top.countries <- data.latest.all %>% filter(ranking <= k + 1) %>%
  arrange(ranking) %>% pull(country) %>% as.character()
top.countries %>% setdiff('World') %>% print()
```

```
##  [1] "China"          "Italy"          "US"             "Spain"
##  [5] "Germany"        "Iran"           "France"         "Switzerland"
##  [9] "United Kingdom" "Korea, South"   "Netherlands"    "Austria"
## [13] "Belgium"        "Canada"         "Norway"         "Portugal"
## [17] "Brazil"         "Sweden"         "Turkey"         "Israel"
```

```
## add 'Others'
top.countries %<>% c('Others')
## put all others in a single group of 'Others'
data.latest <- data.latest.all %>% filter(!is.na(country)) %>%
  mutate(country=ifelse(ranking <= k + 1, as.character(country), 'Others')) %>%
  mutate(country=country %>% factor(levels=c(top.countries)))
data.latest %<>% group_by(country) %>%
  summarise(confirmed=sum(confirmed), new.confirmed=sum(new.confirmed),
            current.confirmed=sum(current.confirmed),
            recovered=sum(recovered), deaths=sum(deaths), new.deaths=sum(new.deaths)) %>%
  mutate(death.rate=(100 * deaths/confirmed) %>% round(1))
data.latest %<>% select(c(country, confirmed, deaths, death.rate,
                 new.confirmed, new.deaths, current.confirmed))
```

```
data.latest %>% mutate(death.rate=death.rate %>% format(nsmall=1) %>% paste0('%')) %>%
  kable('latex', booktabs=T, row.names=T, align=c('l', rep('r', 6)),
        caption=paste0('Cases in Top 20 Countries - ', max.date.txt,
                       '. See a complete list of all infected countries at the end of this report.'),
        format.args=list(big.mark=',')) %>%
  kable_styling(font_size=7, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

```
## convert from wide to long format, for drawing area plots
data.latest.long <- data.latest %>% filter(country!='World') %>%
  gather(key=type, value=count, -country)
## set factor levels to show them with proper text and in a desirable order
data.latest.long %<>% mutate(type=recode_factor(type,
                                    confirmed='Total Confirmed',
                                    deaths='Total Deaths',
                                    death.rate='Death Rate (%)',
                                    new.confirmed='New Confirmed (compared with one day before)',
                                    new.deaths='New Deaths (compared with one day before)',
                                    current.confirmed='Current Confirmed'))
```

```
## bar chart
data.latest.long %>% ggplot(aes(x=country, y=count, fill=country, group=country)) +
  geom_bar(stat='identity') +
  geom_text(aes(label=count, y=count), size=2, vjust=0) +
  xlab('') + ylab('') +
  labs(title=paste0('Top 20 Countries with Most Confirmed Cases - ', max.date.txt)) +
  scale_fill_discrete(name='Country', labels=aes(count)) +
  theme(legend.title=element_blank(),
        legend.position='none',
        plot.title=element_text(size=11),
```

Table 3: Cases in Top 20 Countries - 25 Mar 2020. See a complete list of all infected countries at the end of this report.

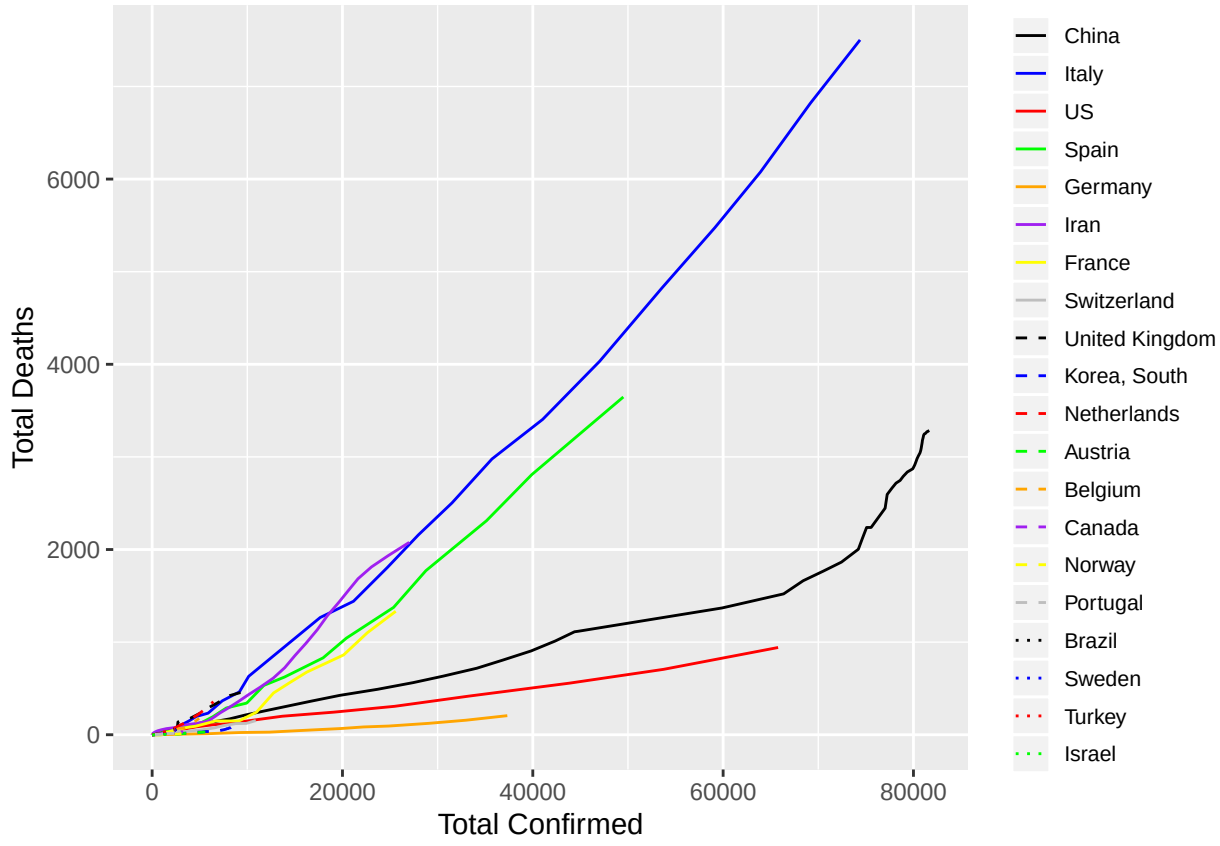
	country	confirmed	deaths	death.rate	new.confirmed	new.deaths	current.confirmed
1	World	467,593	21,180	4.5%	49,548	2,555	446,413
2	China	81,661	3,285	4.0%	70	4	
3	Italy	74,386	7,503	10.1%	5,210	683	
4	US	65,778	942	1.4%	12,038	236	
5	Spain	49,515	3,647	7.4%	9,630	839	
6	Germany	37,323	206	0.6%	4,337	49	
7	Iran	27,017	2,077	7.7%	2,206	143	
8	France	25,600	1,333	5.2%	2,978	231	
9	Switzerland	10,897	153	1.4%	1,020	31	
10	United Kingdom	9,640	466	4.8%	1,476	43	
11	Korea, South	9,137	126	1.4%	100	6	
12	Netherlands	6,438	357	5.5%	858	80	
13	Austria	5,588	30	0.5%	305	2	
14	Belgium	4,937	178	3.6%	668	56	
15	Canada	3,251	30	0.9%	461	4	
16	Norway	3,084	14	0.5%	221	2	
17	Portugal	2,995	43	1.4%	633	10	
18	Brazil	2,554	59	2.3%	307	13	
19	Sweden	2,526	62	2.5%	240	26	
20	Turkey	2,433	59	2.4%	561	15	
21	Israel	2,369	5	0.2%	1,131	2	
22	Others	40,464	605	1.5%	5,158	81	

```
      axis.text=element_text(size=7),
      axis.text.x=element_text(angle=45, hjust=1)) +
facet_wrap(~type, ncol=1, scales='free_y')
```

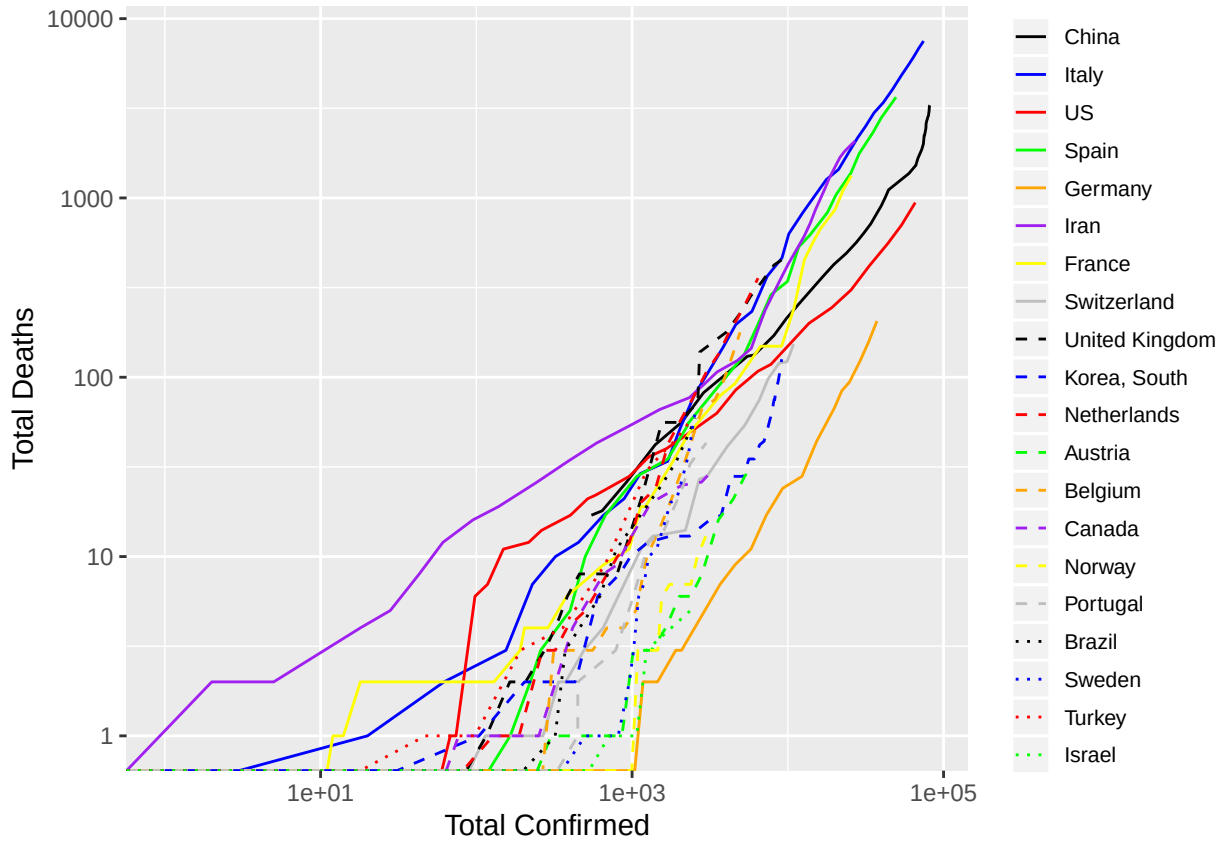
Figure 6: Top 20 Countries with Most Confirmed Cases

5.1 Confirmed vs Deaths

```
# linetypes <- rep(c("dotted", "dashed", "solid"), each=8)
# colors <- rep(c('grey', 'yellow', 'purple', 'orange', 'green', 'red', 'blue', 'black'), 3)
linetypes <- rep(c("solid", "dashed", "dotted"), each=8)
colors <- rep(c('black', 'blue', 'red', 'green', 'orange', 'purple', 'yellow', 'grey'), 3)
df <- data %>% filter(country %in% setdiff(top.countries, c('World', 'Others'))) %>%
  mutate(country=country %>% factor(levels=c(top.countries)))
p <- df %>% ggplot(aes(x=confirmed, y=deaths, group=country)) +
  geom_line(aes(color=country, linetype=country)) +
  xlab('Total Confirmed') + ylab('Total Deaths') +
  scale_linetype_manual(values=linetypes) +
  scale_color_manual(values=colors) +
  theme(legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'))
p
```




```
p + scale_x_log10() + scale_y_log10()
```



The two figures below show the numbers of confirmed cases and deaths of top 20 countries, as well as the death rates up to 25 Mar 2020.

```
df <- data.latest %>% filter(country %in% setdiff(top.countries, 'World'))

plot1 <- df %>% ggplot(aes(x=confirmed, y=deaths, col=death.rate, size=current.confirmed)) +
  scale_size(name='Current Confirmed', trans='log2', breaks=c(1e3, 2e3, 5e3, 1e4, 2e4, 4e4)) +
  geom_text(aes(label=country), size=2.5, check_overlap=T, vjust=-1.6) +
  geom_point() +
  xlab('Total Confirmed') + ylab('Total Deaths') +
  labs(col="Death Rate (%)") +
  scale_color_gradient(low='#56B1F7', high='#132B43') +
  scale_x_log10() + scale_y_log10()

plot2 <- df %>% ggplot(aes(x=new.confirmed, y=new.deaths, col=death.rate, size=current.confirmed)) +
  scale_size(name='Current Confirmed', trans='log2', breaks=c(1e3, 2e3, 5e3, 1e4, 2e4, 4e4)) +
  geom_text(aes(label=country), size=2.5, check_overlap=T, vjust=-1.6) +
  geom_point() +
  xlab('New Confirmed') + ylab('New Deaths') +
  labs(col="Death Rate (%)") +
  scale_color_gradient(low='#56B1F7', high='#132B43') +
  scale_x_log10() + scale_y_log10()

grid.arrange(plot1, plot2, ncol=1)
```

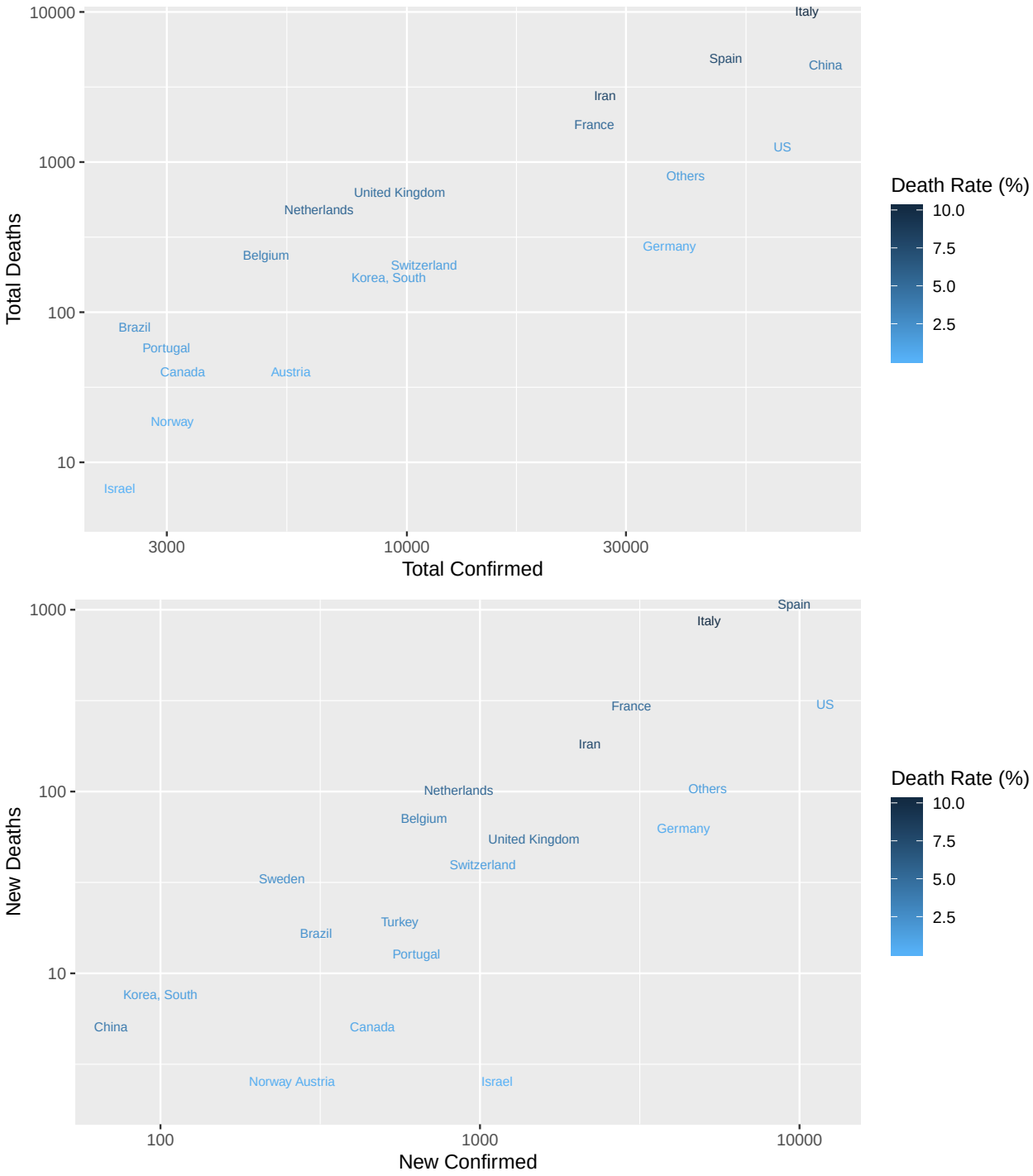


Figure 7: Top 20 Countries

5.2 Comparison across Countries

The area plots blow show the numbers of dead, recovered, total and current confimed cases. Note that, in the area plot, the total number of total confirmed cases is represented by the total areas of current confirmed, recovered and dead.

```r
## plot: cases by type
df <- data.long %>% filter(country %in% top.countries) %<>%
  mutate(country=country %>% factor(levels=c(top.countries)))

p <- df %>% filter(country != 'World') %>%
  ggplot(aes(x=date, y=count)) + xlab('') + ylab('Count')  +
  theme(legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'),
        plot.title=element_text(size=11),
        axis.text.x=element_text(angle=45, hjust=1)) +
  facet_wrap(~type, ncol=2, scales='free_y')

## area plot
plot1 <- p + geom_area(aes(fill=country)) +
  labs(title=paste0('Cases around the World - ', max.date.txt))

## line plot and in log scale
# linetypes <- rep(c("solid", "dashed", "dotted"), each=8)
# colors <- rep(c('black', 'blue', 'red', 'green', 'orange', 'purple', 'yellow', 'grey'), 3)
plot2 <- p + geom_line(aes(color=country, linetype=country)) +
  scale_linetype_manual(values=linetypes) +
  scale_color_manual(values=colors) +
  labs(title=paste0('Cases around the World - Log Scale - ', max.date.txt)) +
  scale_y_continuous(trans='log10')

grid.arrange(plot1, plot2, ncol=1)
```

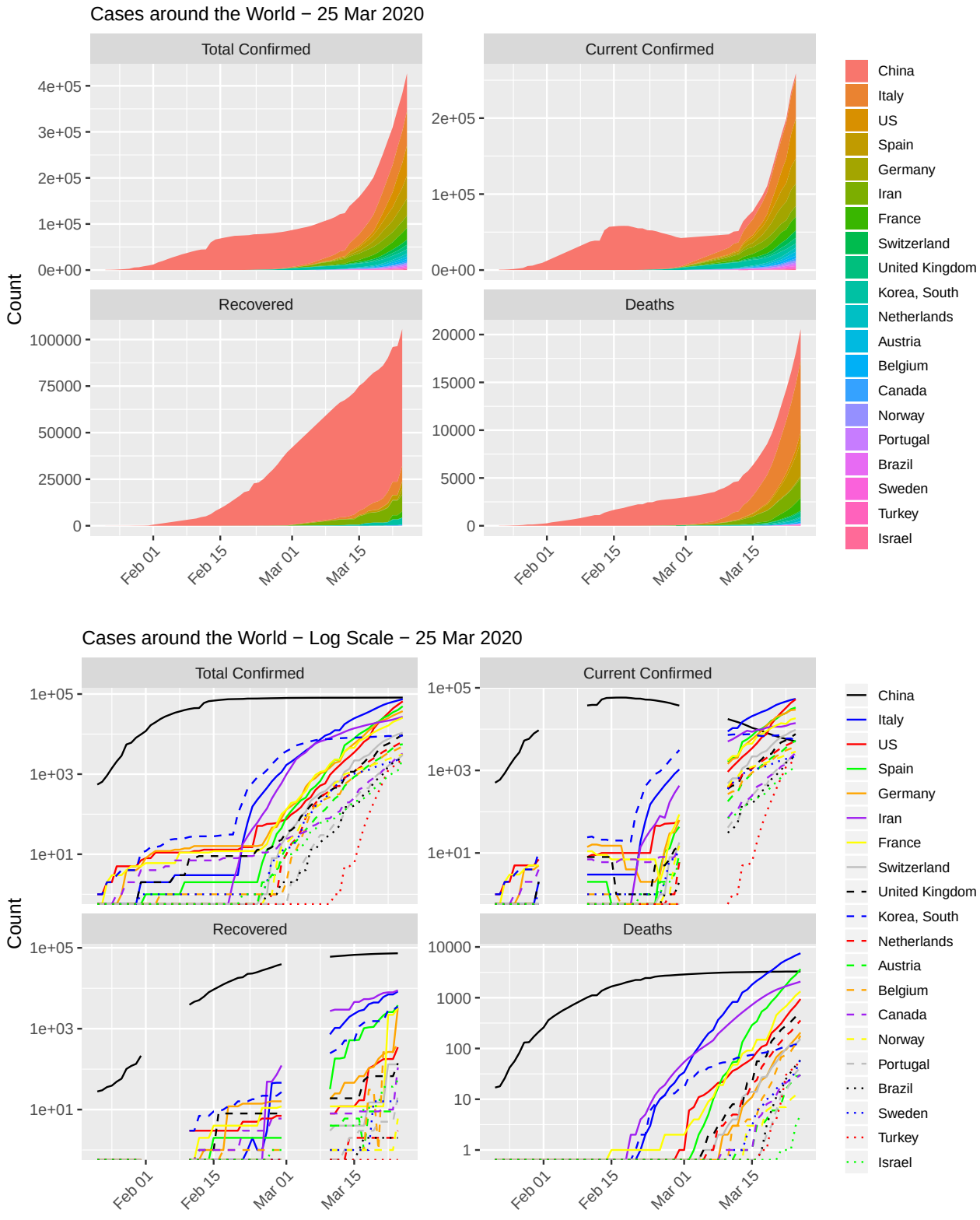


Figure 8: Cases around the World

```
## plot: excluding China
p <- df %>% filter(!(country %in% c('World', 'China'))) %>%
```

```
  ggplot(aes(x=date, y=count)) + xlab('') + ylab('Count') +
  theme(legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'),
        plot.title=element_text(size=11),
        axis.text.x=element_text(angle=45, hjust=1)) +
  facet_wrap(~type, ncol=2, scales='free_y')
p + geom_area(aes(fill=country)) +
  labs(title=paste0('Cases around the World (excl. China) - ', max.date.txt))
```

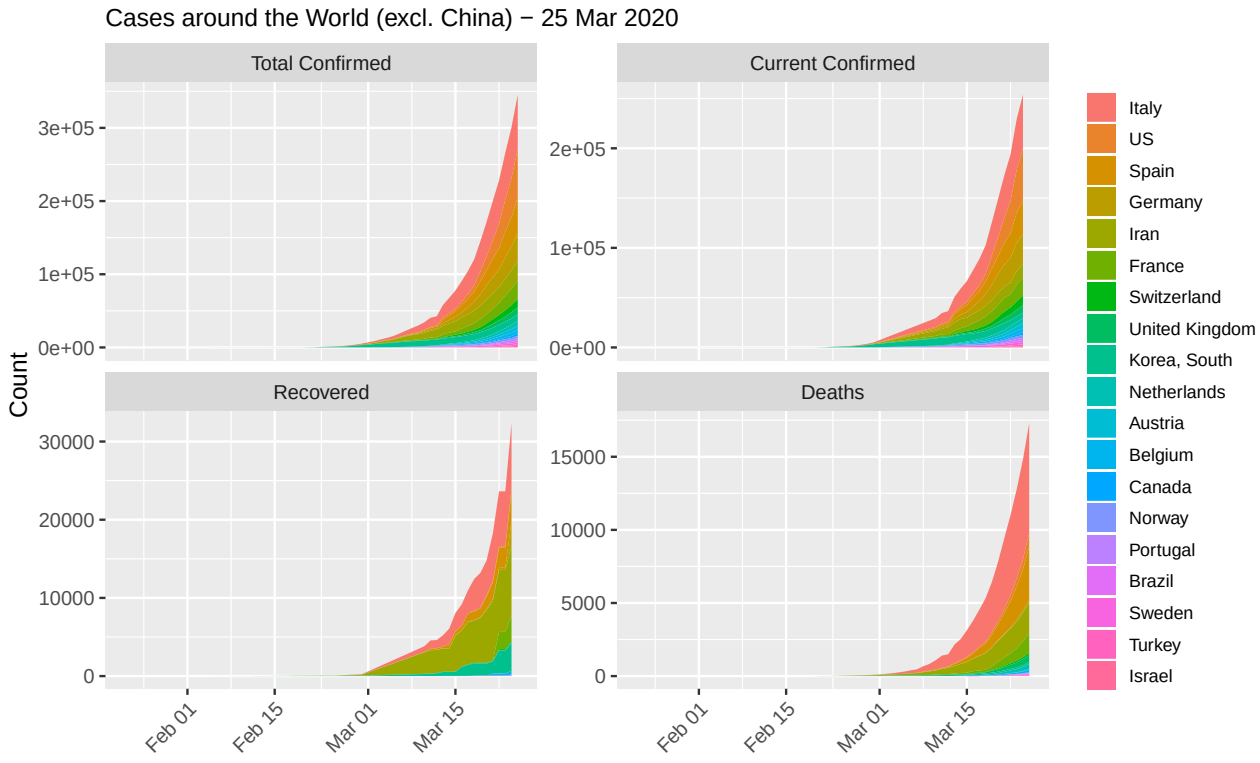


Figure 9: Cases around the World (excl. China)

```
## if Australia in not in top 20, add it in and remove 'Others'
if(!('Australia' %in% top.countries)) {
  top.countries %<>% setdiff('Others') %>% c('Australia')
  df <- data.long %>% filter(country %in% top.countries) %<>%
    mutate(country=country %>% factor(levels=c(top.countries)))
}

## cases by country - area plot
df %>% filter(country != 'World' & type != 'Total Confirmed') %>%
  ggplot(aes(x=date, y=count, fill=type)) +
  geom_area(alpha=0.5) +
  # xlab('') + ylab('') +
  labs(title=paste0('Numbers of COVID-19 Cases in Top 20 Countries - ',
                    max.date.txt)) +
  scale_fill_manual(values=c('red', 'green', 'black')) +
  theme(legend.title=element_blank(), legend.position='bottom',
```

```
      plot.title = element_text(size=12),
      axis.title.x=element_blank(),
      axis.title.y=element_blank(),
      legend.key.size=unit(0.4, 'cm'),
      # legend.text=element_text(size=7),
      strip.text.x=element_text(size=7),
      axis.text=element_text(size=7),
      axis.text.x=element_text(angle=45, hjust=1)) +
facet_wrap(~country, ncol=4, scales='free_y')
```

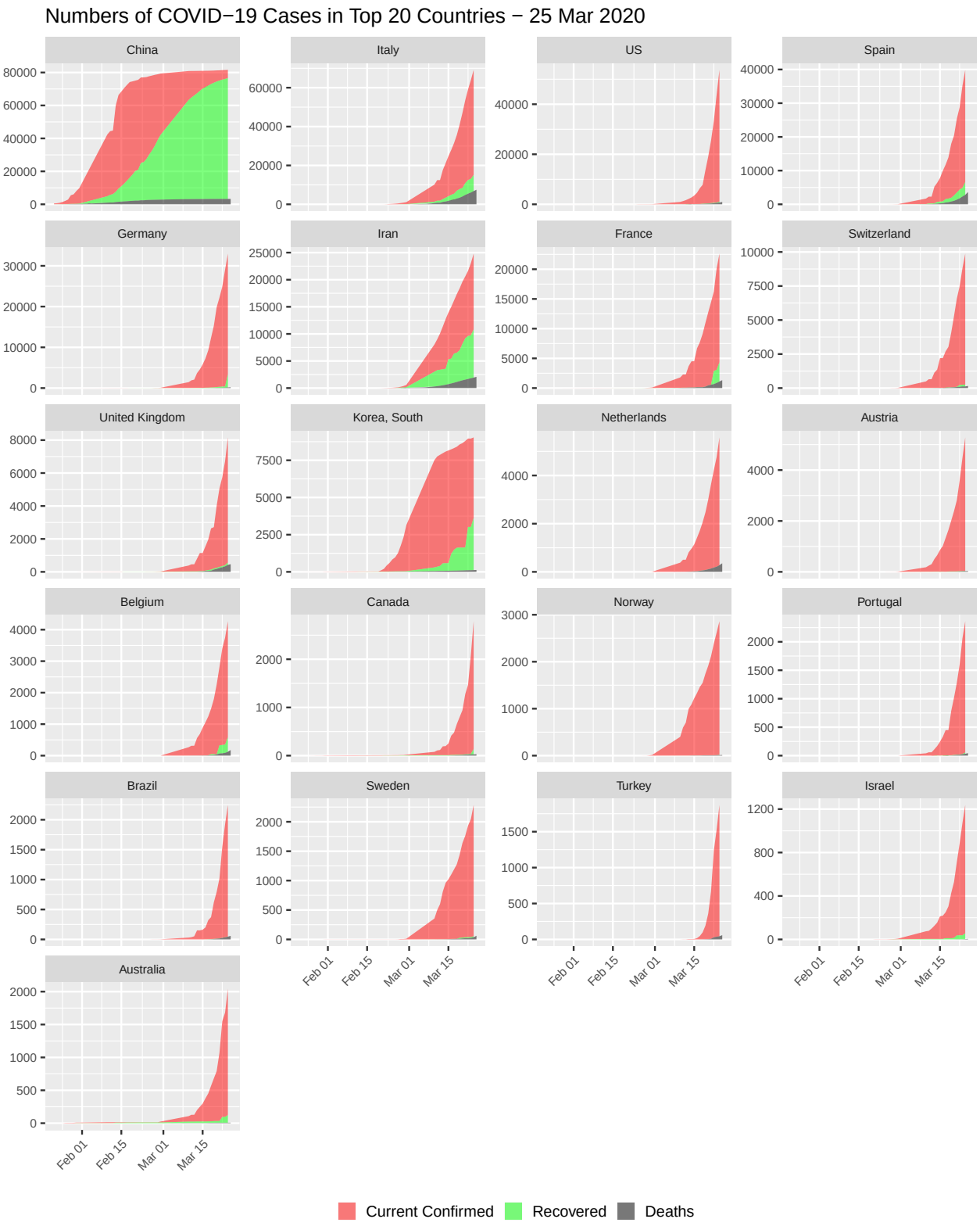


Figure 10: COVID-19 Cases in Top 20 Countries. Ordered descendingly by number of confirmed cases.

```r
## cases by country - line plot - log scale
p <- df %>% filter(country != 'World') %>%
  ggplot(aes(x=date, y=count, color=type)) +
  geom_line() +
  labs(title=paste0('Numbers of COVID-19 Cases in Top 20 Countries (log scale) - ',
                    max.date.txt)) +
  scale_color_manual(values=c('purple', 'red', 'green', 'black')) +
  theme(legend.title=element_blank(), legend.position='bottom',
        plot.title = element_text(size=12),
        axis.title.x=element_blank(),
        axis.title.y=element_blank(),
        legend.key.size=unit(0.4, 'cm'),
        # legend.text=element_text(size=7),
        strip.text.x=element_text(size=7),
        axis.text=element_text(size=7),
        axis.text.x=element_text(angle=45, hjust=1)) +
  scale_y_continuous(trans='log10')
p + facet_wrap(~country, ncol=4, scales='free_y')
```

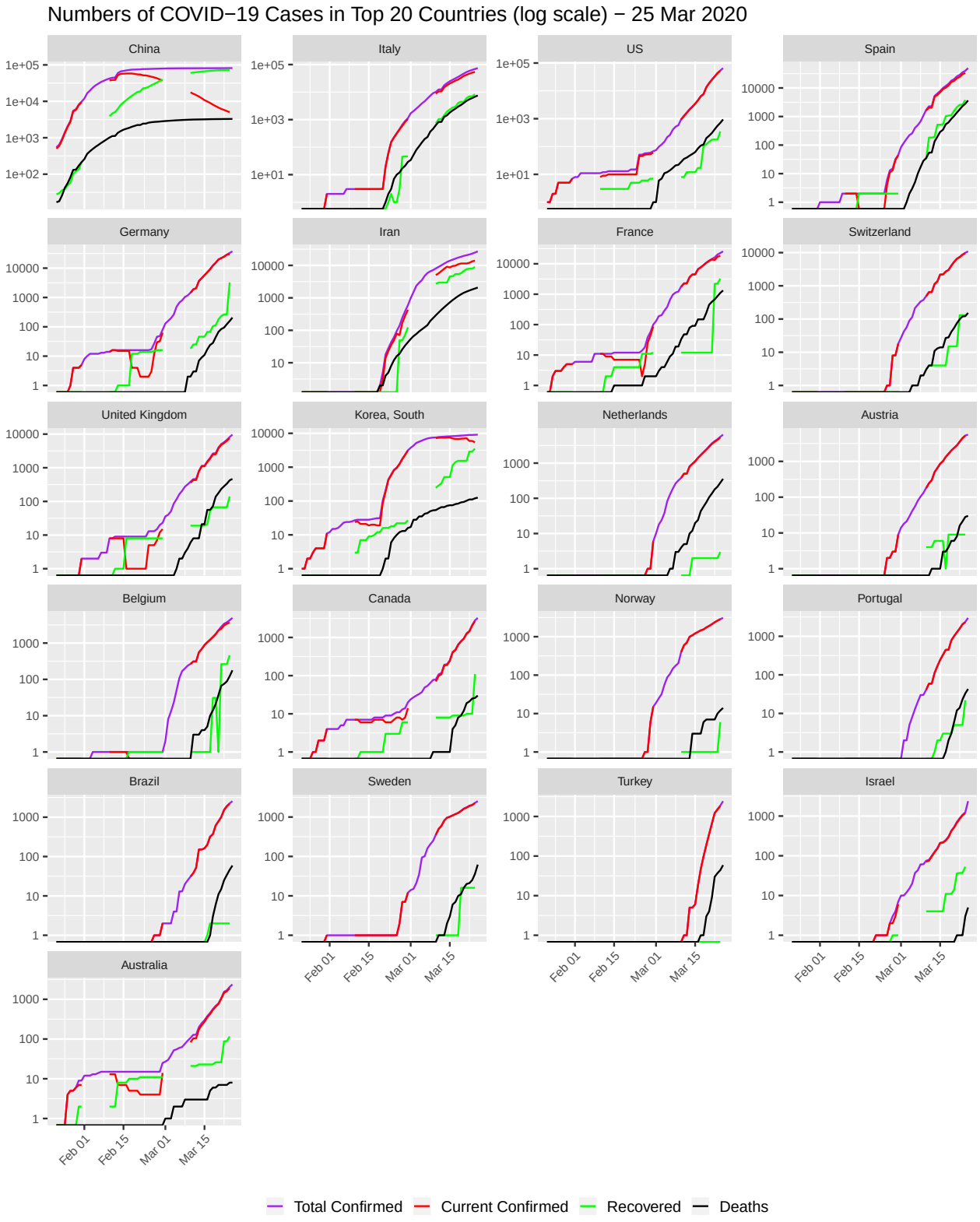


Figure 11: COVID-19 Cases Top 20 Countries (log scale). Ordered descendingly by number of confirmed cases.

```
## plot over multiple pages
# p + facet_wrap_paginate(~country, nrow=4, ncol=3, page=1, scales='free_y')
# p + facet_wrap_paginate(~country, nrow=4, ncol=3, page=2, scales='free_y')
```

Figures 10 and 11 show that the coronavirus seems to be under control in China, with an increase of recovered cases (in green) every day and a shrinking of the current confrimed cases (in red). However, in the rest of the world (i.e., outside of China), the confirmed cases are surging up in many other countries, which suggests that the virus has broken out there.

5.3 Death Rates

```
## three death rates
rate.max <- rates.long$count %>% max(na.rm=T)
df <- rates.long %>% filter(country %in% setdiff(top.countries, 'World')) %>%
  mutate(country=factor(country, levels=top.countries))
df %>% ggplot(aes(x=date, y=count, color=type)) +
  geom_line() +
  xlab('') + ylab('Death Rate (%)') +
  theme(legend.position='bottom', legend.title=element_blank(),
        legend.text=element_text(size=8),
        legend.key.size=unit(0.5, 'cm'),
        axis.text.x=element_text(angle=45, hjust=1)) +
  ylim(c(0, 100)) +
  facet_wrap(~country, ncol=4)
```

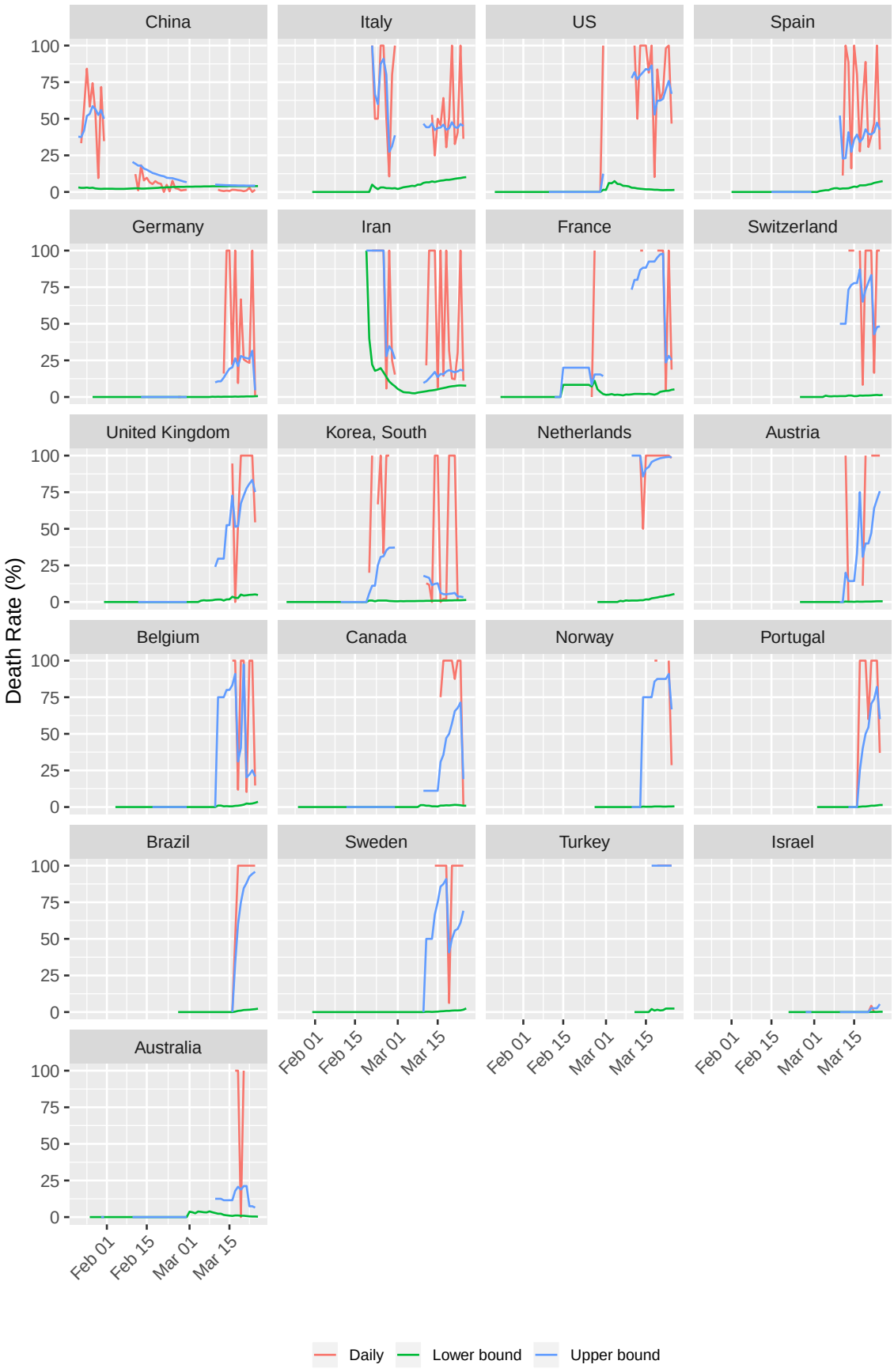


Figure 12: Death Rates

5.4 Countries with Highest Death Rates

Below are a list of top 20 countries of high death rates and with 100+ confirmed cases each.

```
## sort the latest data by death rate, and if tie, by confirmed
df <- data %>% filter(date == max(date) & country != 'World' & confirmed >= 100) %>%
  select(country, confirmed, new.confirmed, current.confirmed,
         recovered, deaths, new.deaths, death.rate=rate.lower) %>%
  arrange(desc(death.rate, confirmed))

df %>% head(20) %>%
  mutate(death.rate=death.rate %>% format(nsmall=1) %>% paste0('%')) %>%
  kable('latex', booktabs=T, row.names=T, align=c('l', rep('r', 7)),
        caption=paste0('Top 20 Countries with Highest Death Rates - ', max.date.txt),
        format.args=list(big.mark=',')) %>%
  kable_styling(font_size=7, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

Table 4: Top 20 Countries with Highest Death Rates - 25 Mar 2020

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	Italy	74,386	5,210			7,503	683	10.1%
2	San Marino	208	21			21	0	10.1%
3	Iraq	346	30			29	2	8.4%
4	Iran	27,017	2,206			2,077	143	7.7%
5	Spain	49,515	9,630			3,647	839	7.4%
6	Indonesia	790	104			58	3	7.3%
7	Algeria	302	38			21	2	7.0%
8	Philippines	636	84			38	3	6.0%
9	Netherlands	6,438	858			357	80	5.5%
10	France	25,600	2,978			1,333	231	5.2%
11	United Kingdom	9,640	1,476			466	43	4.8%
12	Egypt	456	54			21	1	4.6%
13	Hungary	226	39			10	1	4.4%
14	China	81,661	70			3,285	4	4.0%
15	Belgium	4,937	668			178	56	3.6%
16	Albania	146	23			5	0	3.4%
17	Japan	1,307	114			45	2	3.4%
18	Ukraine	145	48			5	2	3.4%
19	Tunisia	173	59			5	1	2.9%
20	Burkina Faso	146	32			4	0	2.7%

6 Conclusions

As of 25 Mar 2020, there are 174 countries with confirmed COVID-19 cases. It seems to be contained in China, but starts to break out in rest of the world. The current death rate is in between 4.5% and 100%, but it is likely to change dramatically with the breakout in many countries, such as European countries.

Appendix A. Processed Data

Blow is the processed data for this analysis.

Appendix A.1 COVID-19 Cases Worldwide

```
## sort by date descendingly and re-order columns
data.world %<>% arrange(desc(date)) %>%
  select(c(date, confirmed, deaths, recovered, current.confirmed,
```

```
          new.confirmed, new.deaths, new.recovered, rate.lower, rate.upper, rate.daily))
## output as a table
data.world %>% kable('latex', booktabs=T, longtable=T, caption='Cases in the Whole World',
               format.args=list(big.mark=',')) %>%
    kable_styling(font_size=4, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

Table 5: Cases in the Whole World

date	confirmed	deaths	recovered	current.confirmed	new.confirmed	new.deaths	new.recovered	rate.lower	rate.upper	rate.daily
2020-03-25	467,593	21,180	0	446,413	49,548	2,555	0	4.5	100.0	100.0
2020-03-24	418,045	18,625	107,659	291,761	39,810	2,120	9,650	4.5	14.7	18.0
2020-03-23	378,235	16,505	98,009	263,721	41,282	1,854	452	4.4	14.4	80.4
2020-03-22	336,953	14,651	97,557	224,745	32,557	1,678	6,207	4.3	13.1	21.3
2020-03-21	304,396	12,973	91,350	200,073	32,361	1,674	4,272	4.3	12.4	28.2
2020-03-20	272,035	11,299	87,078	173,658	29,535	1,432	2,428	4.2	11.5	37.1
2020-03-19	242,500	9,867	84,650	147,983	27,679	1,134	1,663	4.1	10.4	40.5
2020-03-18	214,821	8,733	82,987	123,101	17,719	828	2,472	4.1	9.5	25.1
2020-03-17	197,102	7,905	80,515	108,682	15,528	779	2,752	4.0	8.9	22.1
2020-03-16	181,574	7,126	77,763	96,685	14,120	686	2,054	3.9	8.4	25.0
2020-03-15	167,454	6,440	75,709	85,305	11,353	621	3,410	3.8	7.8	15.4
2020-03-14	156,101	5,819	72,299	77,983	10,896	415	2,373	3.7	7.4	14.9
2020-03-13	145,205	5,404	69,926	69,875	16,853	684	1,927	3.7	7.2	26.2
2020-03-12	128,352	4,720	67,999	55,633	2,477	105	1,321	3.7	6.5	7.4
2020-03-11	125,875	4,615	66,678	54,582	7,255	353	2,314	3.7	6.5	13.2
2020-03-10	118,620	4,262	64,364	49,994	5,030	274	64,364	3.6	6.2	0.4
2020-03-09	113,590	3,988	0	109,602	3,769	186	0	3.5	100.0	100.0
2020-03-08	109,821	3,802	0	106,019	3,974	244	0	3.5	100.0	100.0
2020-03-07	105,847	3,558	0	102,289	4,046	98	0	3.4	100.0	100.0
2020-03-06	101,801	3,460	0	98,341	3,915	112	0	3.4	100.0	100.0
2020-03-05	97,886	3,348	0	94,538	2,766	94	0	3.4	100.0	100.0
2020-03-04	95,120	3,254	0	91,866	2,280	94	0	3.4	100.0	100.0
2020-03-03	92,840	3,160	0	89,680	2,534	75	0	3.4	100.0	100.0
2020-03-02	90,306	3,085	0	87,221	1,937	89	0	3.4	100.0	100.0
2020-03-01	88,369	2,996	0	85,373	2,358	55	0	3.4	100.0	100.0
2020-02-29	86,011	2,941	39,772	43,298	1,899	69	3,071	3.4	6.9	2.2
2020-02-28	84,112	2,872	36,701	44,539	1,366	58	3,434	3.4	7.3	1.7
2020-02-27	82,746	2,814	33,267	46,665	1,358	44	2,893	3.4	7.8	1.5
2020-02-26	81,388	2,770	30,374	48,244	982	62	2,469	3.4	8.4	2.4
2020-02-25	80,406	2,708	27,905	49,793	845	79	2,678	3.4	8.8	2.9
2020-02-24	79,561	2,629	25,227	51,705	603	160	1,833	3.3	9.4	8.0
2020-02-23	78,958	2,469	23,394	53,095	386	11	509	3.1	9.5	2.1
2020-02-22	78,572	2,458	22,885	53,229	1,753	207	3,996	3.1	9.7	4.9
2020-02-21	76,819	2,251	18,889	55,679	622	4	713	2.9	10.6	0.6
2020-02-20	76,197	2,247	18,176	55,774	558	125	2,056	2.9	11.0	5.7
2020-02-19	75,639	2,122	16,120	57,397	503	115	1,768	2.8	11.6	6.1
2020-02-18	75,136	2,007	14,352	58,777	1,878	139	1,769	2.7	12.3	7.3
2020-02-17	73,258	1,868	12,583	58,807	2,034	98	1,718	2.5	12.9	5.4
2020-02-16	71,224	1,770	10,865	58,589	2,194	104	1,470	2.5	14.0	6.6
2020-02-15	69,030	1,666	9,395	57,969	2,145	143	1,337	2.4	15.1	9.7
2020-02-14	66,885	1,523	8,058	57,304	6,517	152	1,763	2.3	15.9	7.9
2020-02-13	60,368	1,371	6,295	52,702	15,147	253	1,145	2.3	17.9	18.1
2020-02-12	45,221	1,118	5,150	38,953	419	5	467	2.5	17.8	1.1
2020-02-11	44,802	1,113	4,683	39,006	2,040	100	737	2.5	19.2	11.9
2020-02-10	42,762	1,013	3,946	37,803	2,612	107	3,946	2.4	20.4	2.6
2020-02-09	40,150	906	0	39,244	3,030	100	0	2.3	100.0	100.0
2020-02-08	37,120	806	0	36,314	2,729	87	0	2.2	100.0	100.0
2020-02-07	34,391	719	0	33,672	3,597	85	0	2.1	100.0	100.0
2020-02-06	30,794	634	0	30,160	3,159	70	0	2.1	100.0	100.0
2020-02-05	27,635	564	0	27,071	3,743	72	0	2.0	100.0	100.0
2020-02-04	23,892	492	0	23,400	4,011	66	0	2.1	100.0	100.0
2020-02-03	19,881	426	0	19,455	3,094	64	0	2.1	100.0	100.0
2020-02-02	16,787	362	0	16,425	4,749	103	0	2.2	100.0	100.0
2020-02-01	12,038	259	0	11,779	2,111	46	0	2.2	100.0	100.0
2020-01-31	9,927	213	222	9,492	1,693	42	79	2.1	49.0	34.7
2020-01-30	8,234	171	143	7,920	2,068	38	17	2.1	54.5	69.1
2020-01-29	6,166	133	126	5,907	588	2	19	2.2	51.4	9.5
2020-01-28	5,578	131	107	5,340	2,651	49	46	2.3	55.0	51.6
2020-01-27	2,927	82	61	2,784	809	26	9	2.8	57.3	74.3
2020-01-26	2,118	56	52	2,010	684	14	13	2.6	51.9	51.9
2020-01-25	1,434	42	39	1,353	493	16	3	2.9	51.9	84.2
2020-01-24	941	26	36	879	287	8	6	2.8	41.9	57.1
2020-01-23	654	18	30	606	99	1	2	2.8	37.5	33.3
2020-01-22	555	17	28	510				3.1	37.8	

Appendix A.2 Latest Cases by Country

```
## hightlight high death rates (if >= 5%) for those countries with 100+ confirmed cases
data.latest.all %>% arrange(desc(confirmed)) %>% select(-c(date, ranking)) %>%
  mutate(death.rate = cell_spec(death.rate, "latex",
                                color = ifelse(confirmed >= 100 & death.rate >= 5, "red", "black"),
```

```
                              bold = ifelse(confirmed >= 100 & death.rate >= 5, T, F))) %>%
  kable(format='latex', escape=F, booktabs=T, longtable=T, row.names=T,
        caption=paste0('Cases by Country (', max.date.txt, ')'),
        format.args=list(big.mark=',')) %>%
  kable_styling(font_size=6, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

Table 6: Cases by Country (25 Mar 2020)

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	World	467,593	49,548	446,413	0	21,180	2,555	4.5
2	China	81,661	70			3,285	4	4
3	Italy	74,386	5,210			7,503	683	**10.1**
4	US	65,778	12,038			942	236	1.4
5	Spain	49,515	9,630			3,647	839	**7.4**
6	Germany	37,323	4,337			206	49	0.6
7	Iran	27,017	2,206			2,077	143	**7.7**
8	France	25,600	2,978			1,333	231	**5.2**
9	Switzerland	10,897	1,020			153	31	1.4
10	United Kingdom	9,640	1,476			466	43	4.8
11	Korea, South	9,137	100			126	6	1.4
12	Netherlands	6,438	858			357	80	**5.5**
13	Austria	5,588	305			30	2	0.5
14	Belgium	4,937	668			178	56	3.6
15	Canada	3,251	461			30	4	0.9
16	Norway	3,084	221			14	2	0.5
17	Portugal	2,995	633			43	10	1.4
18	Brazil	2,554	307			59	13	2.3
19	Sweden	2,526	240			62	26	2.5
20	Turkey	2,433	561			59	15	2.4
21	Israel	2,369	1,131			5	2	0.2
22	Australia	2,364	320			8	0	0.3
23	Denmark	1,862	144			34	2	1.8
24	Malaysia	1,796	172			20	4	1.1
25	Czechia	1,654	260			6	3	0.4
26	Ireland	1,564	235			9	2	0.6
27	Luxembourg	1,333	234			8	0	0.6
28	Japan	1,307	114			45	2	3.4
29	Ecuador	1,173	91			28	1	2.4
30	Chile	1,142	220			3	1	0.3
31	Pakistan	1,063	91			8	1	0.8
32	Poland	1,051	150			14	4	1.3
33	Thailand	934	107			4	0	0.4
34	Romania	906	112			17	6	1.9
35	Saudi Arabia	900	133			2	1	0.2
36	Finland	880	88			3	2	0.3
37	Greece	821	78			22	2	2.7
38	Indonesia	790	104			58	3	**7.3**
39	Iceland	737	89			2	0	0.3
40	Diamond Princess	712	0			10	0	1.4
41	South Africa	709	155			0	0	0
42	Russia	658	163			3	2	0.5
43	India	657	121			12	2	1.8
44	Philippines	636	84			38	3	**6**
45	Singapore	631	73			2	0	0.3
46	Qatar	537	11			0	0	0
47	Slovenia	528	48			5	1	0.9
48	Peru	480	64			9	2	1.9
49	Colombia	470	92			4	1	0.9
50	Egypt	456	54			21	1	4.6
51	Panama	443	98			8	2	1.8
52	Croatia	442	60			1	0	0.2
53	Bahrain	419	27			4	1	1
54	Mexico	405	38			5	1	1.2
55	Estonia	404	35			1	1	0.2
56	Dominican Republic	392	80			10	4	2.6
57	Argentina	387	0			8	2	2.1
58	Serbia	384	81			4	1	1
59	Iraq	346	30			29	2	**8.4**
60	Lebanon	333	15			6	2	1.8

Table 6: Cases by Country (25 Mar 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
61	United Arab Emirates	333	85			2	0	0.6
62	Algeria	302	38			21	2	**7**
63	Lithuania	274	65			4	2	1.5
64	Armenia	265	16			0	0	0
65	Bulgaria	242	24			3	0	1.2
66	Taiwan*	235	20			2	0	0.9
67	Hungary	226	39			10	1	4.4
68	Morocco	225	55			6	1	2.7
69	Latvia	221	24			0	0	0
70	Slovakia	216	12			0	0	0
71	San Marino	208	21			21	0	**10.1**
72	New Zealand	205	50			0	0	0
73	Costa Rica	201	24			2	0	1
74	Kuwait	195	4			0	0	0
75	Uruguay	189	27			0	0	0
76	Andorra	188	24			1	0	0.5
77	North Macedonia	177	29			3	1	1.7
78	Bosnia and Herzegovina	176	10			3	0	1.7
79	Tunisia	173	59			5	1	2.9
80	Jordan	172	18			0	0	0
81	Moldova	149	24			1	0	0.7
82	Albania	146	23			5	0	3.4
83	Burkina Faso	146	32			4	0	2.7
84	Ukraine	145	48			5	2	3.4
85	Vietnam	141	7			0	0	0
86	Cyprus	132	8			3	0	2.3
87	Malta	129	19			0	0	0
88	Brunei	109	5			0	0	0
89	Sri Lanka	102	0			0	0	0
90	Oman	99	15			0	0	0
91	Senegal	99	13			0	0	0
92	Cambodia	96	5			0	0	0
93	Azerbaijan	93	6			2	1	2.2
94	Ghana	93	40			4	2	4.3
95	Venezuela	91	7			0	0	0
96	Belarus	86	5			0	0	0
97	Afghanistan	84	10			2	1	2.4
98	Kazakhstan	81	9			0	0	0
99	Cote d'Ivoire	80	7			0	0	0
100	Cameroon	75	9			1	1	1.3
101	Georgia	75	5			0	0	0
102	Trinidad and Tobago	60	3			1	1	1.7
103	Uzbekistan	60	10			0	0	0
104	Cuba	57	9			1	0	1.8
105	Montenegro	52	5			1	0	1.9
106	Liechtenstein	51	0			0	0	0
107	Nigeria	51	7			1	0	2
108	Congo (Kinshasa)	48	3			2	0	4.2
109	Mauritius	48	6			2	0	4.2
110	Kyrgyzstan	44	2			0	0	0
111	Rwanda	41	1			0	0	0
112	Bangladesh	39	0			5	1	12.8
113	Paraguay	37	10			3	1	8.1
114	Honduras	36	6			0	0	0
115	Bolivia	32	3			0	0	0
116	Monaco	31	8			0	0	0
117	Kenya	28	3			0	0	0
118	Jamaica	26	5			1	0	3.8
119	Guatemala	24	3			1	0	4.2
120	Togo	23	3			0	0	0
121	Madagascar	19	2			0	0	0
122	Barbados	18	0			0	0	0
123	Uganda	14	5			0	0	0
124	Maldives	13	0			0	0	0
125	Ethiopia	12	0			0	0	0
126	Tanzania	12	0			0	0	0
127	Zambia	12	9			0	0	0
128	Djibouti	11	8			0	0	0

Table 6: Cases by Country (25 Mar 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
129	Mongolia	10	0			0	0	0
130	El Salvador	9	4			0	0	0
131	Equatorial Guinea	9	0			0	0	0
132	Haiti	8	1			0	0	0
133	Suriname	8	1			0	0	0
134	Dominica	7	5			0	0	0
135	Namibia	7	0			0	0	0
136	Niger	7	4			1	1	14.3
137	Seychelles	7	0			0	0	0
138	Benin	6	0			0	0	0
139	Gabon	6	0			1	0	16.7
140	Bahamas	5	0			0	0	0
141	Fiji	5	1			0	0	0
142	Guyana	5	0			1	0	20
143	Mozambique	5	2			0	0	0
144	Syria	5	4			0	0	0
145	Cabo Verde	4	1			1	0	25
146	Congo (Brazzaville)	4	0			0	0	0
147	Eritrea	4	3			0	0	0
148	Eswatini	4	0			0	0	0
149	Guinea	4	0			0	0	0
150	Holy See	4	0			0	0	0
151	Angola	3	0			0	0	0
152	Antigua and Barbuda	3	0			0	0	0
153	Central African Republic	3	0			0	0	0
154	Chad	3	0			0	0	0
155	Gambia	3	0			1	0	33.3
156	Laos	3	1			0	0	0
157	Liberia	3	0			0	0	0
158	Nepal	3	1			0	0	0
159	Saint Lucia	3	0			0	0	0
160	Sudan	3	0			1	0	33.3
161	Zimbabwe	3	0			1	0	33.3
162	Belize	2	1			0	0	0
163	Bhutan	2	0			0	0	0
164	Guinea-Bissau	2	2			0	0	0
165	Mali	2	2			0	0	0
166	Mauritania	2	0			0	0	0
167	Nicaragua	2	0			0	0	0
168	Saint Kitts and Nevis	2	2			0	0	0
169	Grenada	1	0			0	0	0
170	Libya	1	0			0	0	0
171	Papua New Guinea	1	0			0	0	0
172	Saint Vincent and the Grenadines	1	0			0	0	0
173	Somalia	1	0			0	0	0
174	Timor-Leste	1	0			0	0	0
175	West Bank and Gaza	-1	0			-1	0	100

Appendix B. How to Cite This Work

Citation

Yanchang Zhao, COVID-19 Data Analysis with R – Worldwide. RDataMining.com, 2020. URL: http://www.rdatamining.com/docs/Coronavirus-data-analysis-world.pdf.

BibTex

@techreport{Zhao2020Covid19world,
Author = {Yanchang Zhao},
Institution = {RDataMining.com},
Title = {COVID-19 Data Analysis with R – Worldwide},
Url = {http://www.rdatamining.com/docs/Coronavirus-data-analysis-world.pdf},
Year = {2020}}

Appendix C. Contact

Contact:

Dr. Yanchang Zhao

Email: yanchang@RDataMining.com

Twitter: @RDataMining

LinkedIn: http://group.rdatamining.com

Comments and suggestions and welcome. Thanks!