COVID-19 Data Analysis with R - Worldwide

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

30 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] 253  72
```

Each dataset has 253 rows, corresponding to country/region/province/state. It has 72 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" "2020-03-29"
```

```
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 29 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=T) %>% merge(data.recovered, all=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 177 countries with confirmed COVID-19 cases, as of 29 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
```

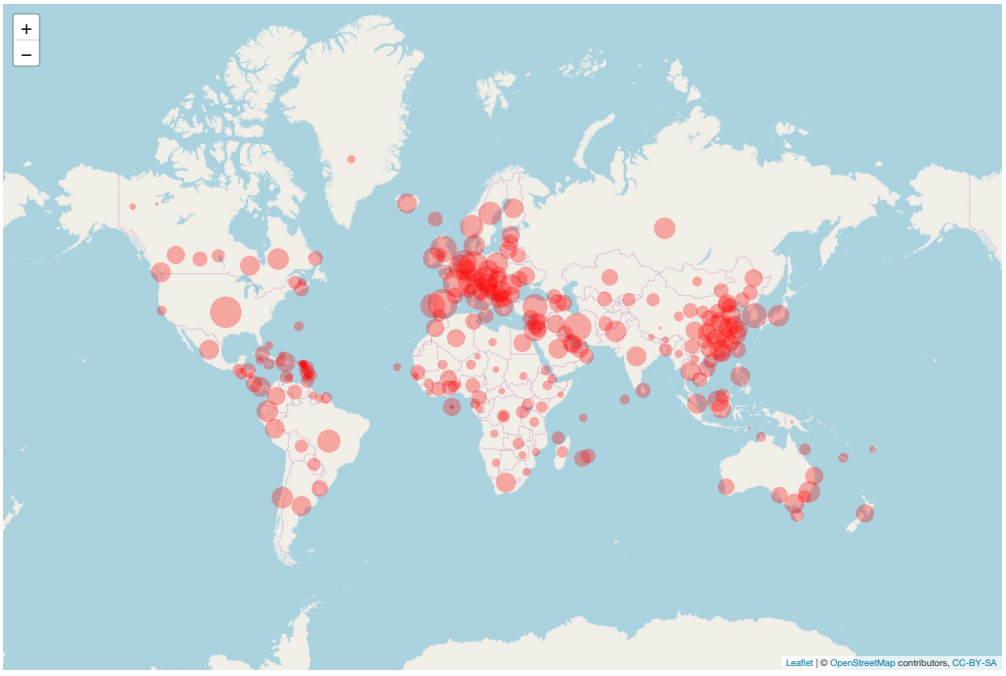


Figure 1: World Map

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)
```

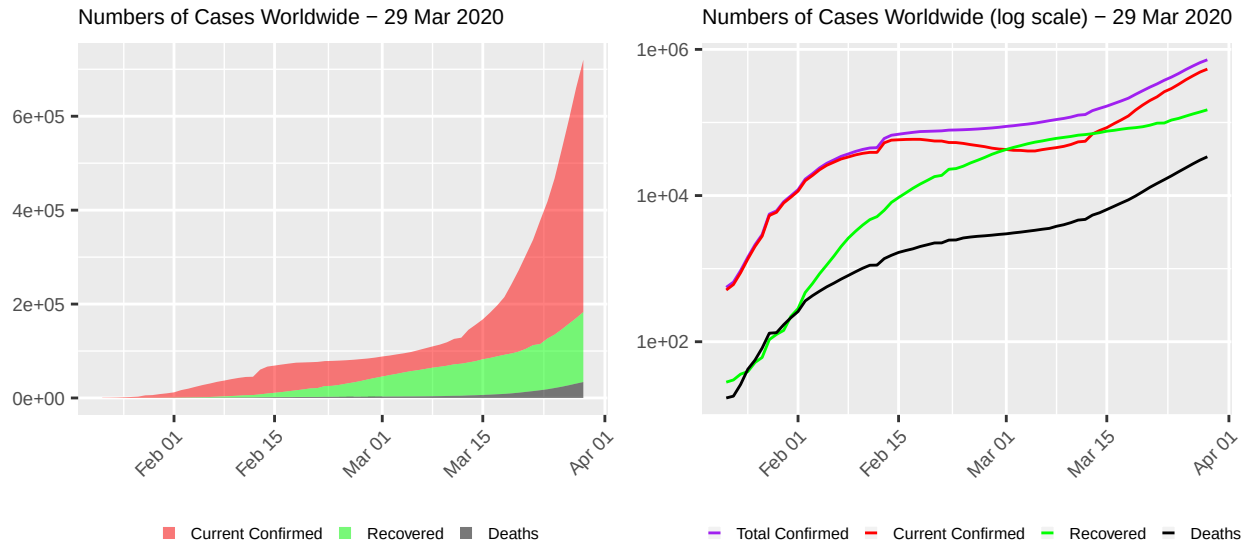


Figure 2: COVID-19 Cases Worldwide

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)
```

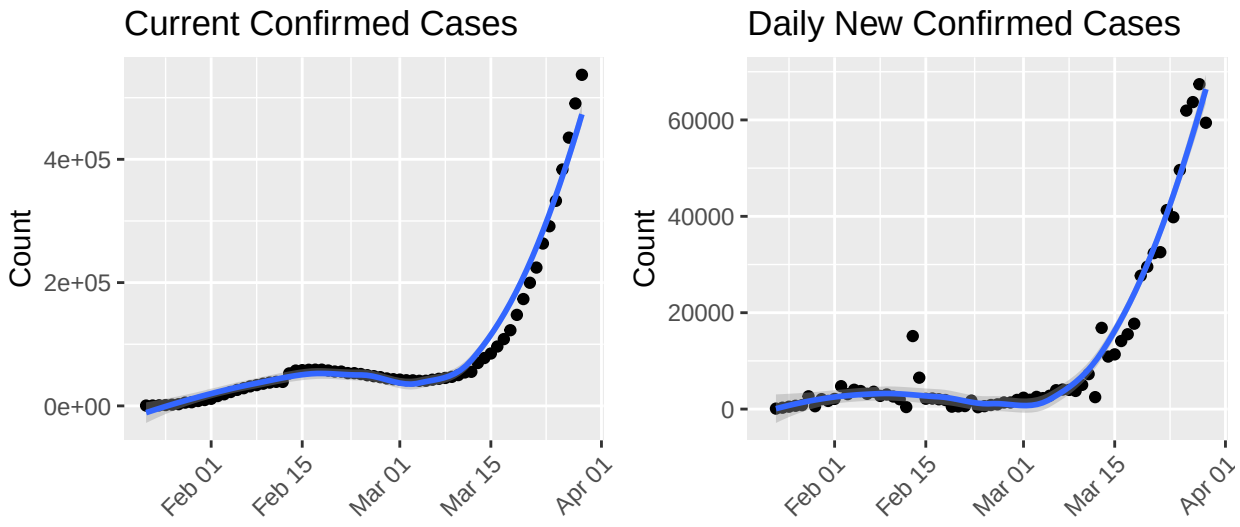


Figure 3: Current Confirmed Cases

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)
```

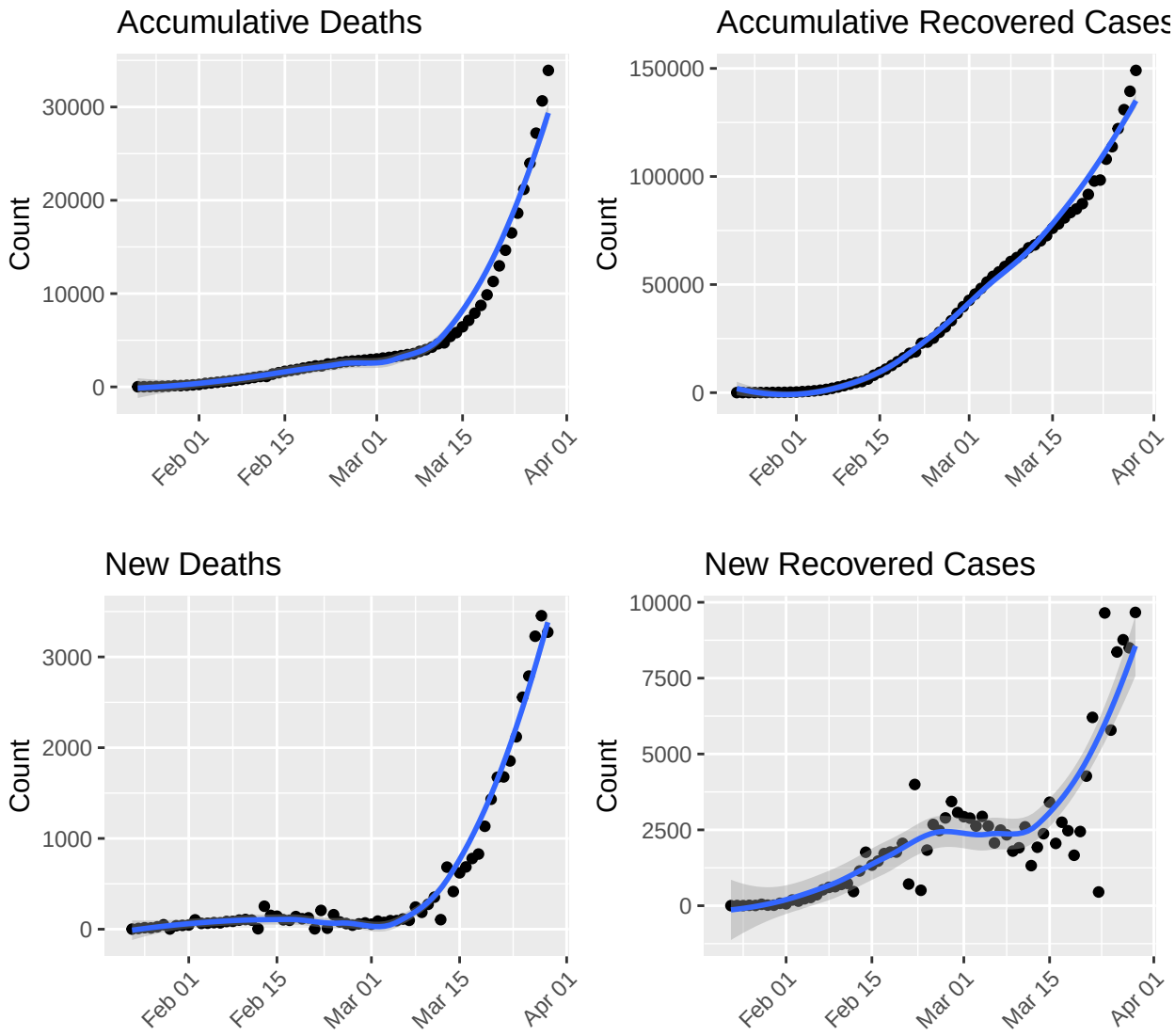


Figure 4: Deaths and Recovered Cases

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 29 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 29 Mar 2020 UTC, it will be between 4.7% and 18.5%.

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)
```

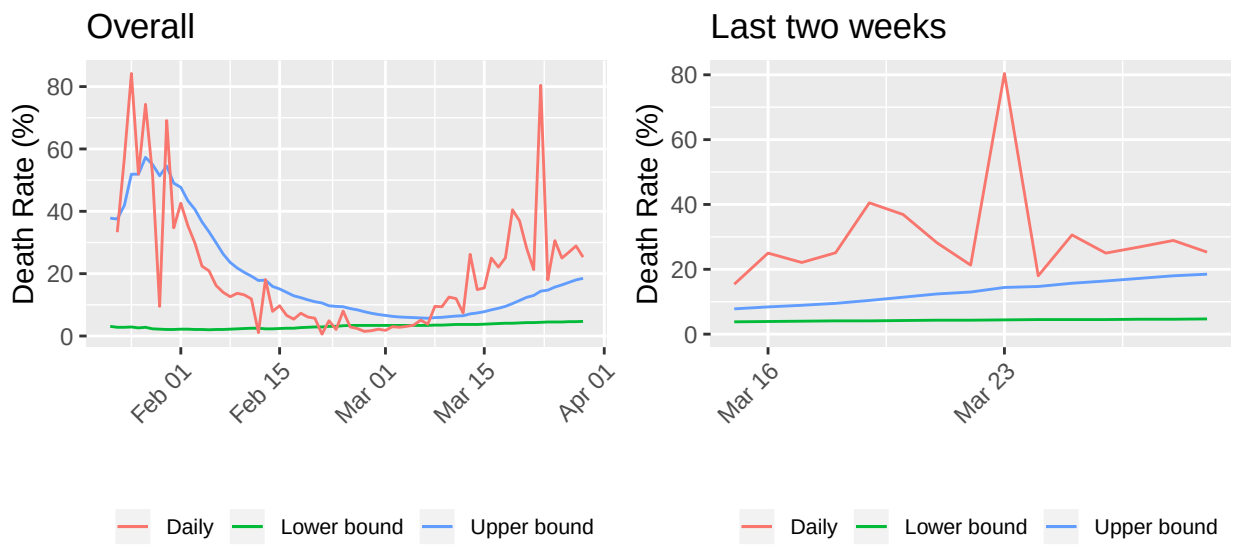


Figure 5: Death Rate

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

```
## 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 - 29 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	720,117	33,925	4.7%	59,411	3,273	537,110
2	US	140,886	2,467	1.8%	19,408	441	135,754
3	Italy	97,689	10,779	11.0%	5,217	756	73,880
4	China	82,122	3,304	4.0%	123	5	3,236
5	Spain	80,110	6,803	8.5%	6,875	821	58,598
6	Germany	62,095	533	0.9%	4,400	100	52,351
7	France	40,708	2,611	6.4%	2,603	294	30,871
8	Iran	38,309	2,640	6.9%	2,901	123	23,278
9	United Kingdom	19,780	1,231	6.2%	2,468	210	18,398
10	Switzerland	14,829	300	2.0%	753	36	12,934
11	Netherlands	10,930	772	7.1%	1,111	132	9,905
12	Belgium	10,836	431	4.0%	1,702	78	9,046
13	Korea, South	9,583	152	1.6%	105	8	4,398
14	Turkey	9,217	131	1.4%	1,815	23	8,981
15	Austria	8,788	86	1.0%	517	18	8,223
16	Canada	6,280	64	1.0%	704	3	5,750
17	Portugal	5,962	119	2.0%	792	19	5,800
18	Norway	4,284	25	0.6%	269	2	4,252
19	Brazil	4,256	136	3.2%	352	25	4,114
20	Israel	4,247	15	0.4%	628	3	4,100
21	Australia	3,984	16	0.4%	344	2	3,724
22	Others	65,222	1,310	2.0%	6,324	174	59,517

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

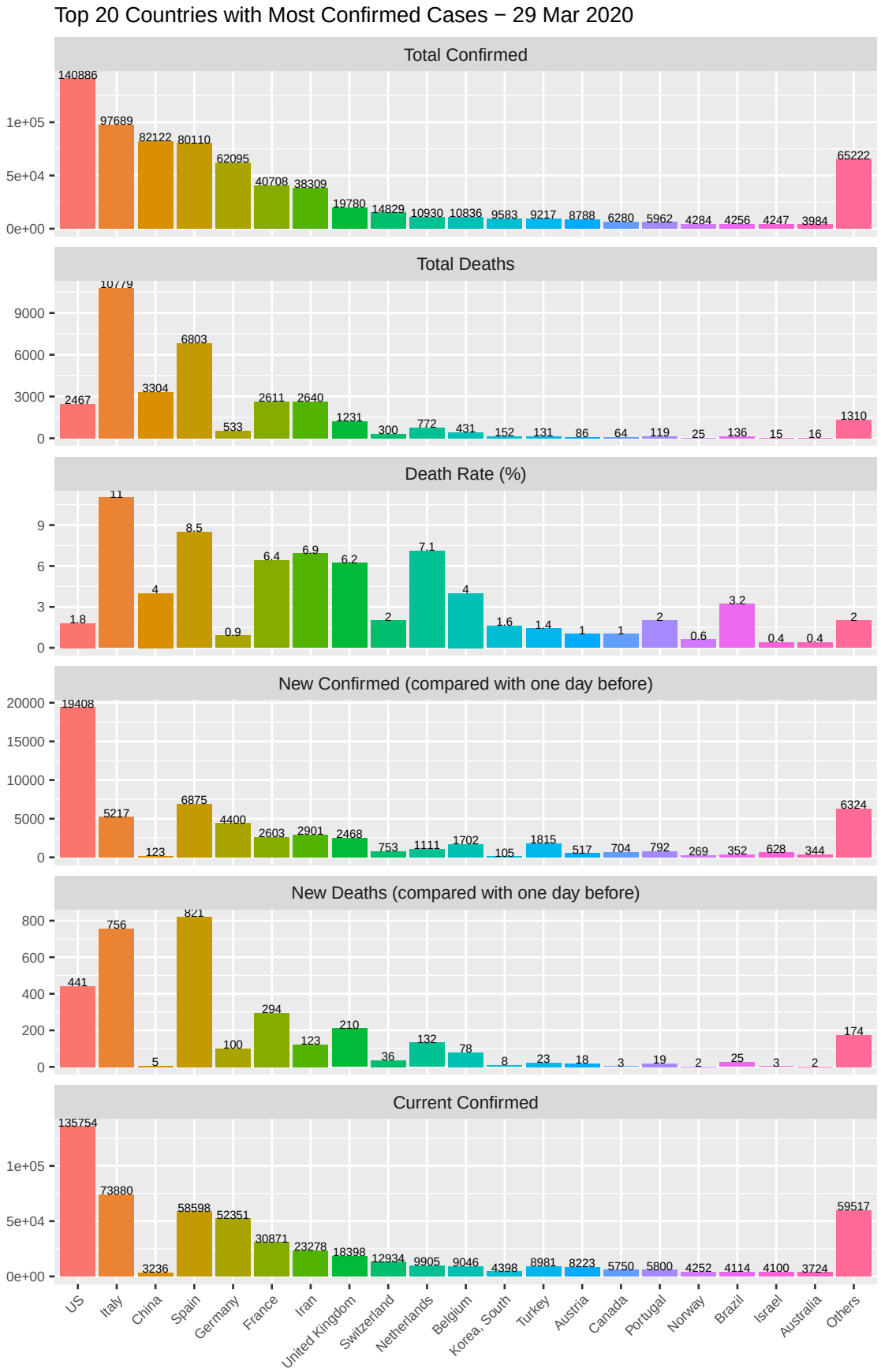
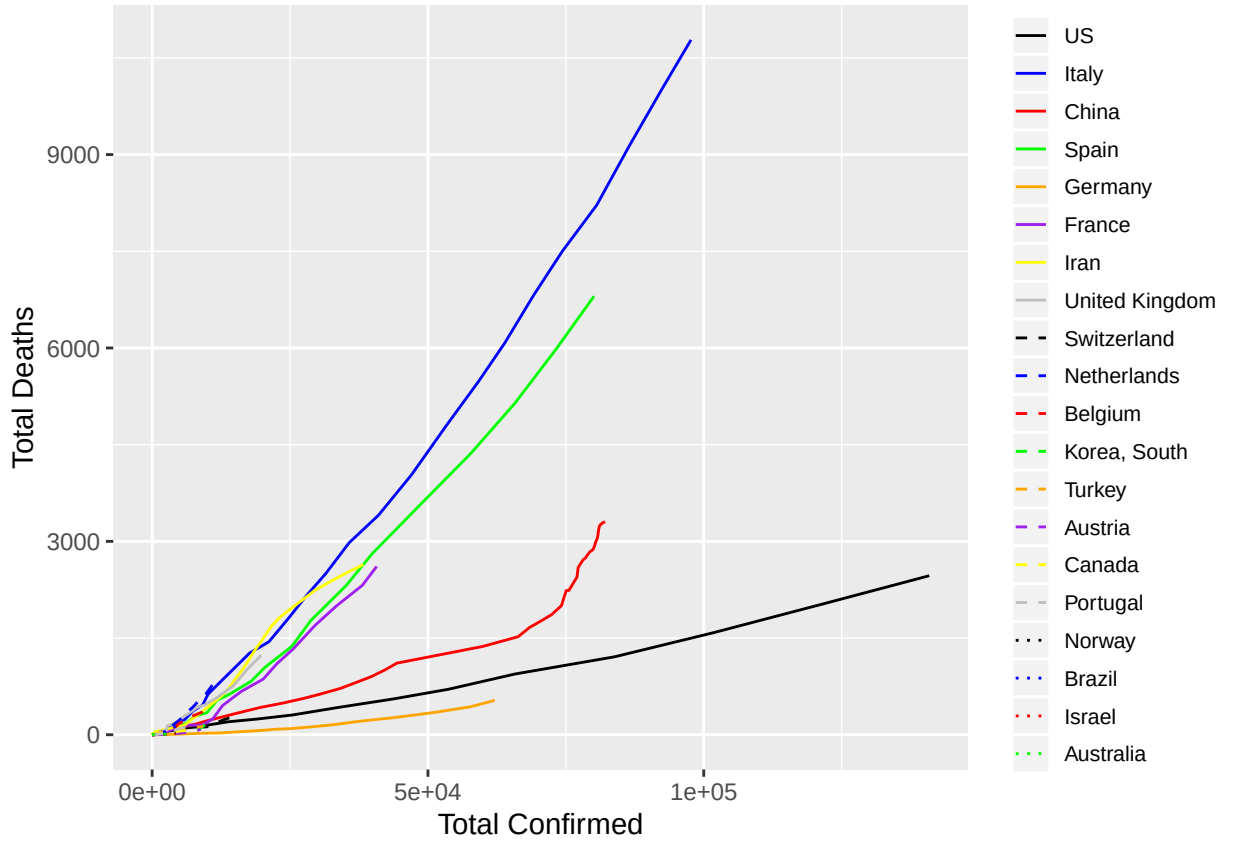


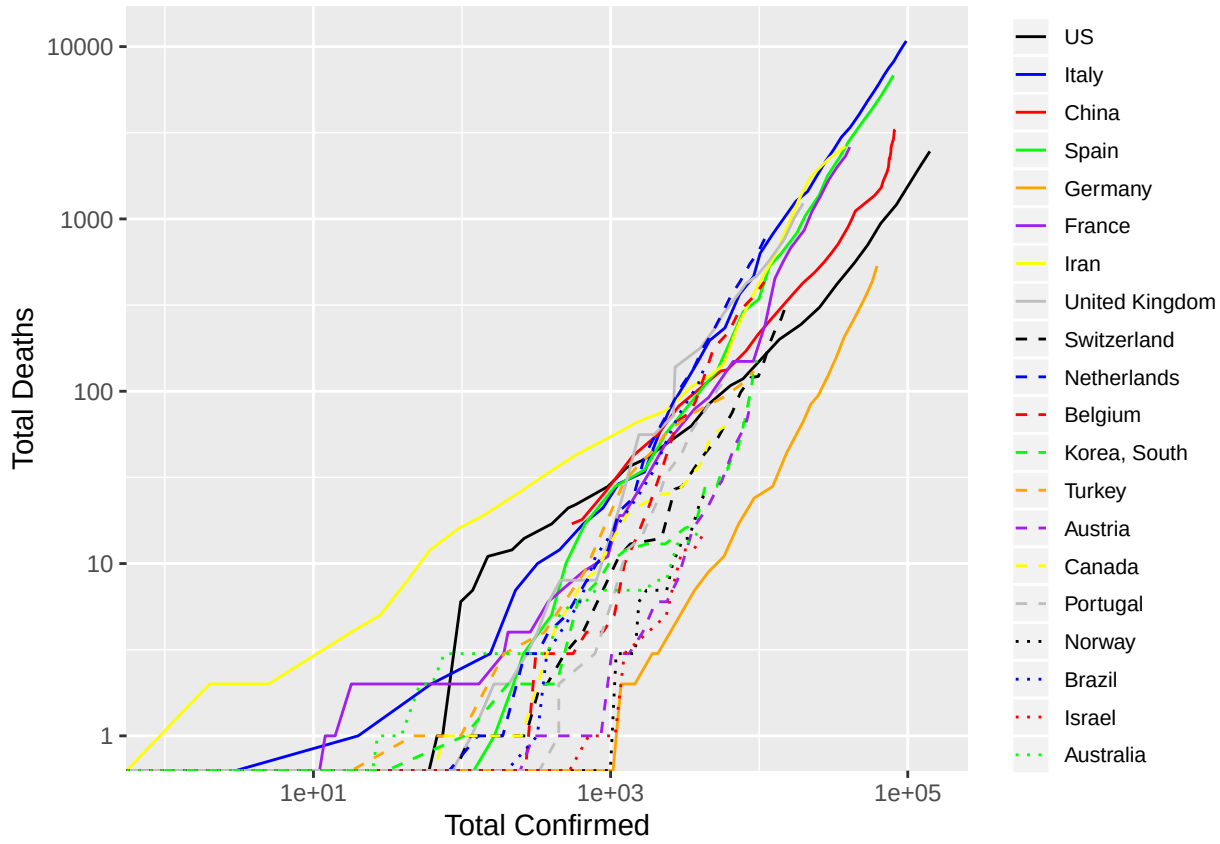
Figure 6: Top 20 Countries with Most Confirmed Cases

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 29 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)
```

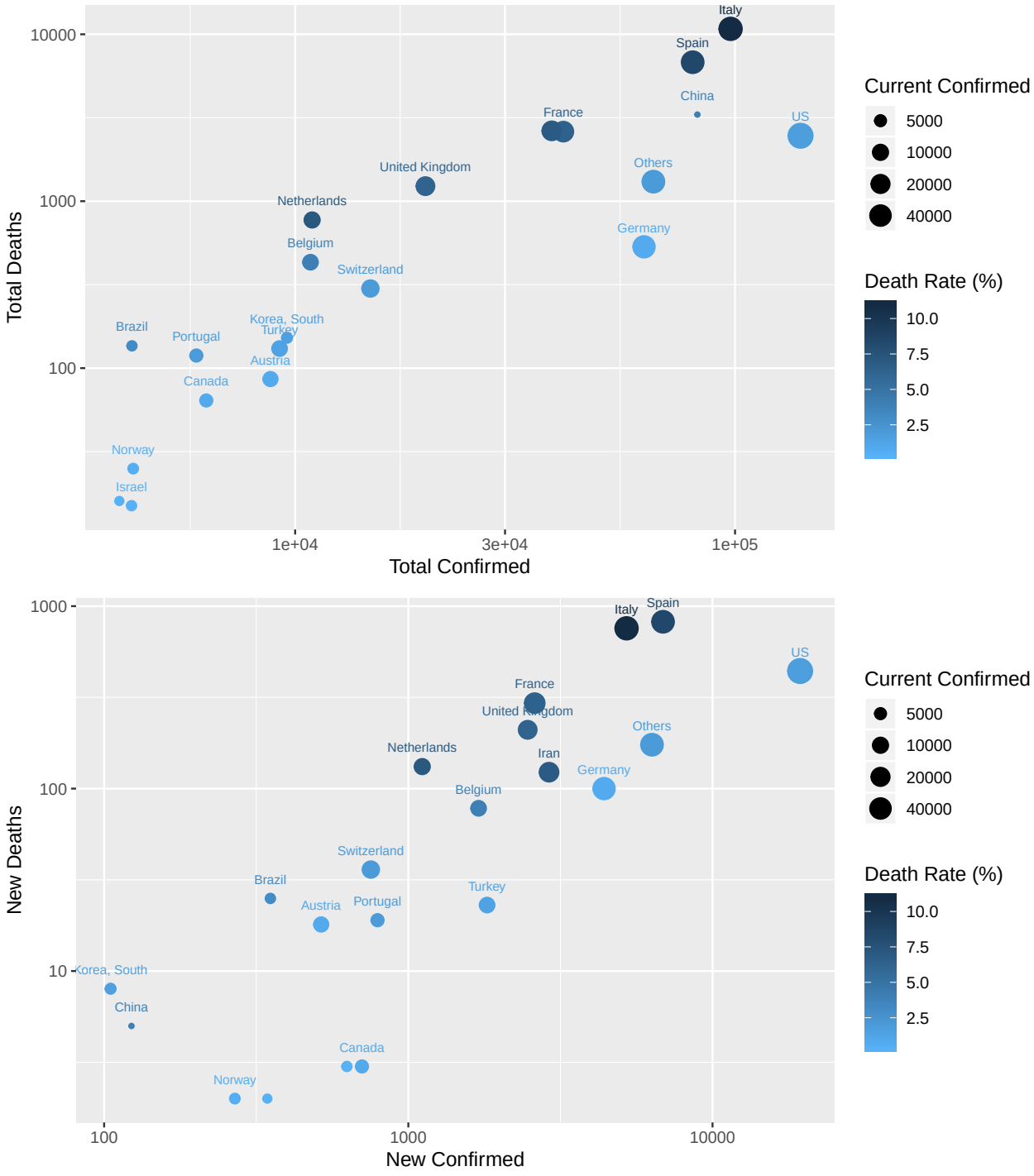


Figure 7: Top 20 Countries

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)
```

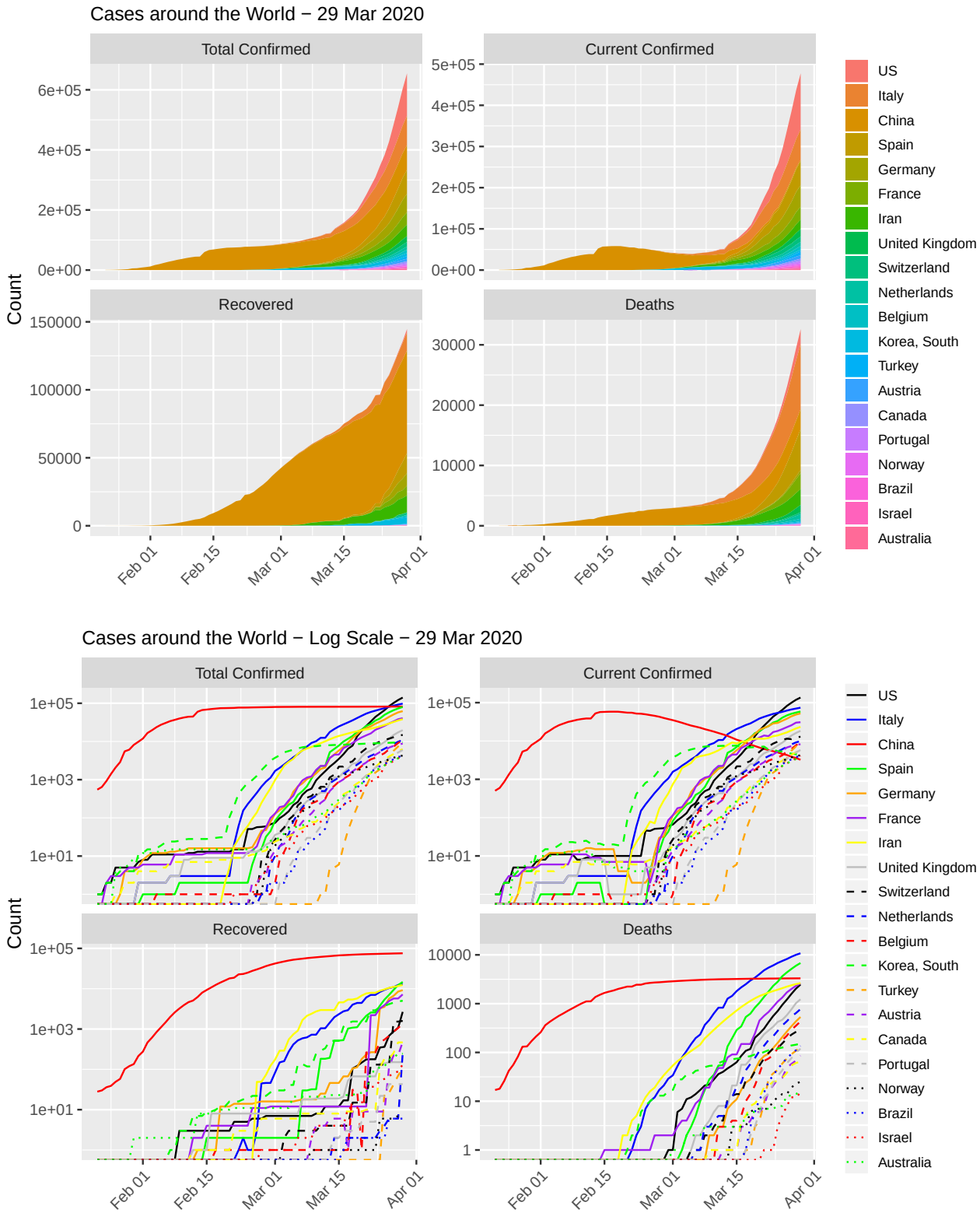


Figure 8: Cases around the World

```
## 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))
```

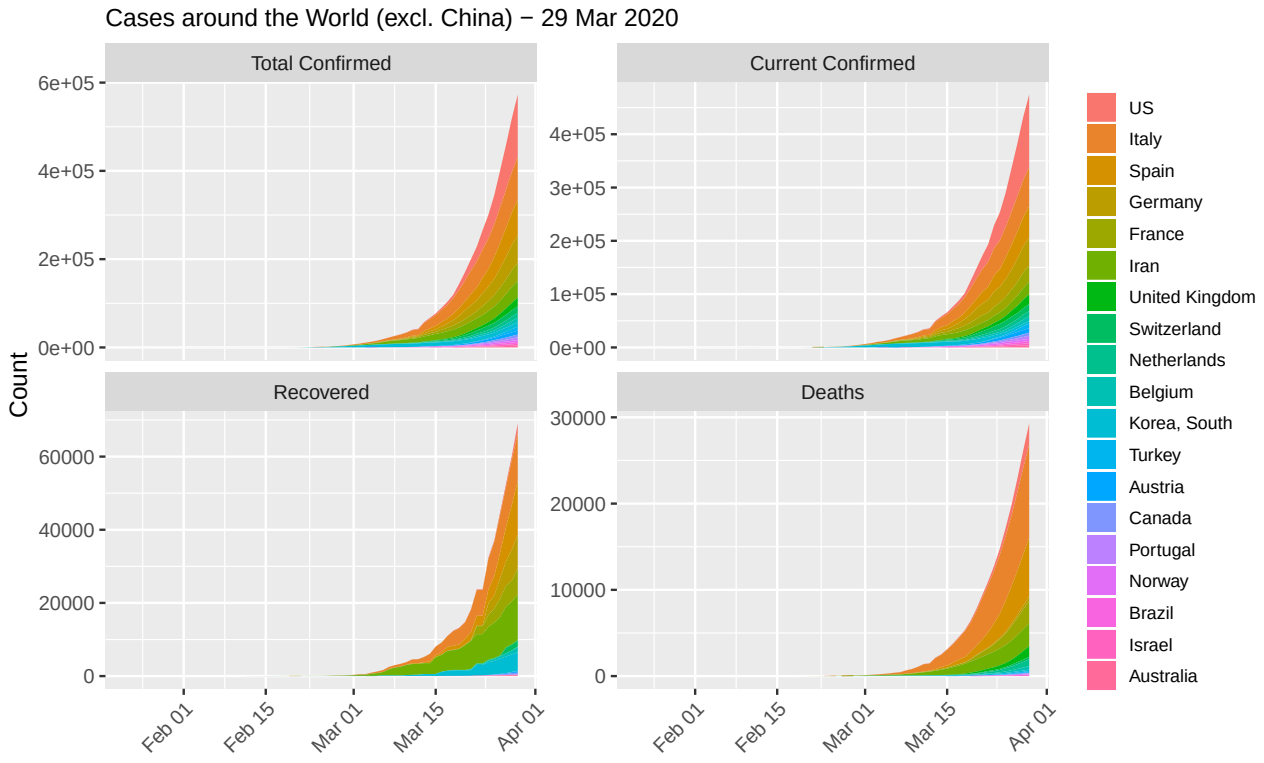


Figure 9: Cases around the World (excl. China)

```
## 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.

```
## 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')
```

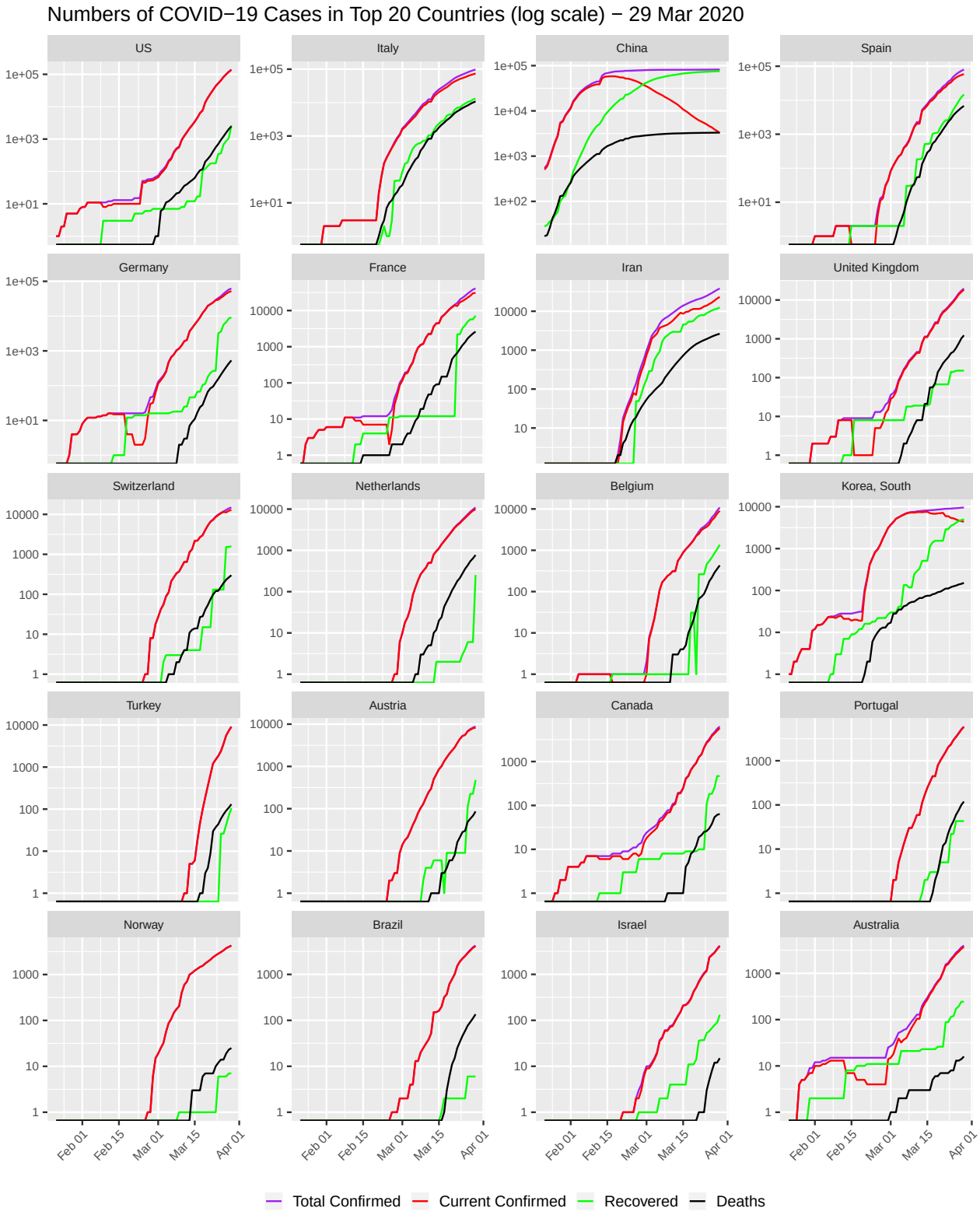


Figure 11: COVID-19 Cases Top 20 Countries (log scale). Ordered descendingly by number of confirmed cases.

```
## 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)
```

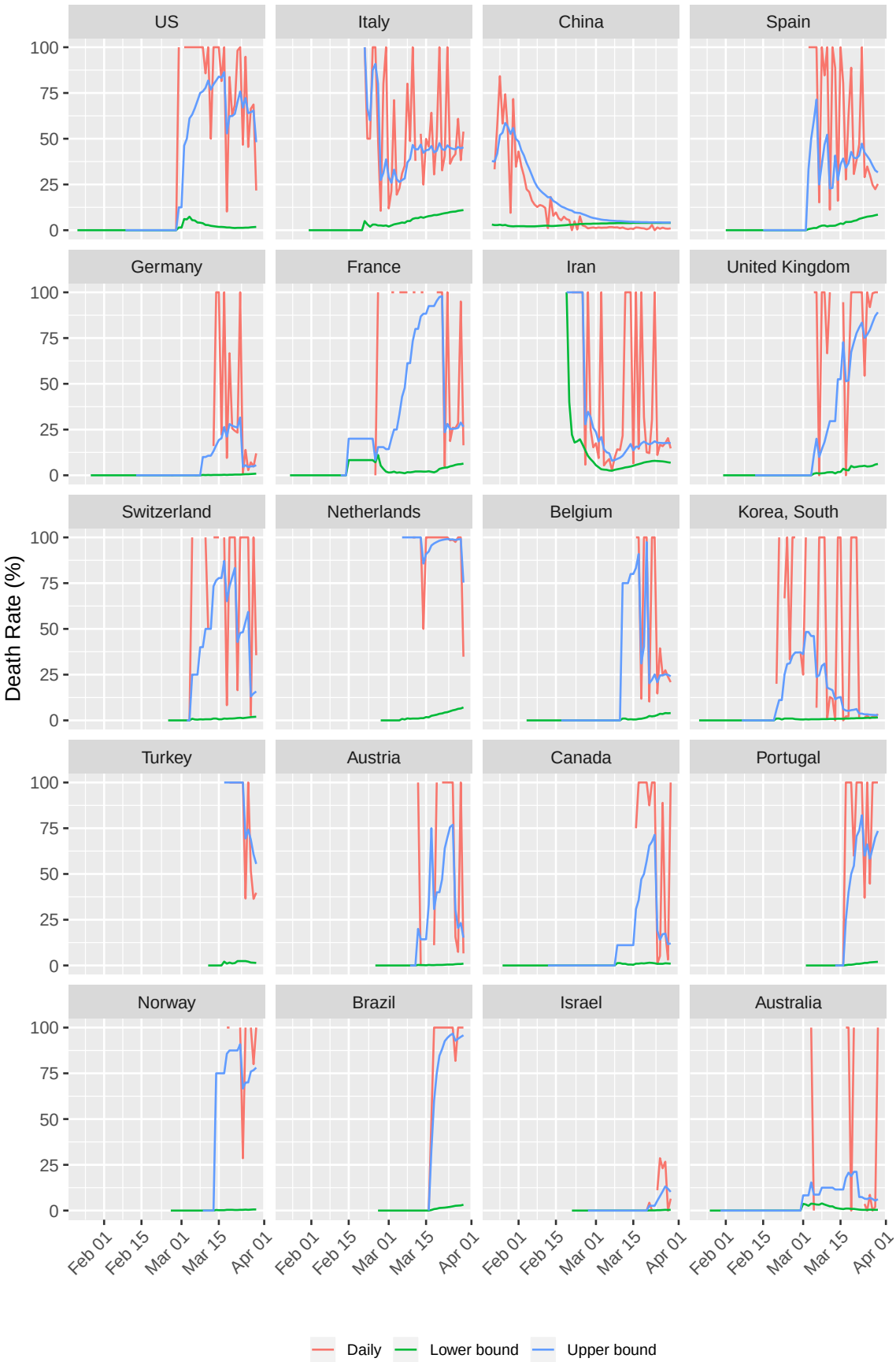


Figure 12: Death Rates

5.4 Countries with Highest Death Rates

Below are a list of top 20 countries with the highest death rates out of countires having 1000+ confirmed cases.

```
## sort the latest data by death rate, and if tie, by confirmed
df <- data %>% filter(date == max(date) & country != 'World' & confirmed >= 1000) %>%
  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 - 29 Mar 2020

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	Italy	97,689	5,217	73,880	13,030	10,779	756	11.0%
2	Indonesia	1,285	130	1,107	64	114	12	8.9%
3	Spain	80,110	6,875	58,598	14,709	6,803	821	8.5%
4	Netherlands	10,930	1,111	9,905	253	772	132	7.1%
5	Iran	38,309	2,901	23,278	12,391	2,640	123	6.9%
6	France	40,708	2,603	30,871	7,226	2,611	294	6.4%
7	United Kingdom	19,780	2,468	18,398	151	1,231	210	6.2%
8	Philippines	1,418	343	1,305	42	71	3	5.0%
9	Belgium	10,836	1,702	9,046	1,359	431	78	4.0%
10	China	82,122	123	3,236	75,582	3,304	5	4.0%
11	Greece	1,156	95	1,066	52	38	6	3.3%
12	Brazil	4,256	352	4,114	6	136	25	3.2%
13	Ecuador	1,924	101	1,863	3	58	10	3.0%
14	Sweden	3,700	253	3,574	16	110	5	3.0%
15	Japan	1,866	173	1,388	424	54	2	2.9%
16	Denmark	2,564	198	2,419	73	72	7	2.8%
17	India	1,024	37	902	95	27	3	2.6%
18	Romania	1,815	363	1,566	206	43	6	2.4%
19	Portugal	5,962	792	5,800	43	119	19	2.0%
20	Switzerland	14,829	753	12,934	1,595	300	36	2.0%

6 Conclusions

As of 29 Mar 2020, there are 177 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 4.7% and 18.5%, 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
2020-03-29	720,117	33,925	149,082	537,110	59,411	3,273	9,667	4.7	18.5	25.3
2020-03-28	660,706	30,652	139,415	490,639	67,415	3,454	8,500	4.6	18.0	28.9
2020-03-27	593,291	27,198	130,915	435,178	63,700	3,228	8,765	4.6	17.2	26.9
2020-03-26	529,591	23,970	122,150	383,471	61,938	2,789	8,363	4.5	16.4	25.0
2020-03-25	467,653	21,181	113,787	332,685	49,608	2,556	5,787	4.5	15.7	30.6
2020-03-24	418,045	18,625	108,000	291,420	39,810	2,120	9,649	4.5	14.7	18.0
2020-03-23	378,235	16,505	98,351	263,379	41,282	1,854	452	4.4	14.4	80.4
2020-03-22	336,953	14,651	97,899	224,403	32,557	1,678	6,207	4.3	13.0	21.3
2020-03-21	304,396	12,973	91,692	199,731	32,361	1,674	4,272	4.3	12.4	28.2
2020-03-20	272,035	11,299	87,420	173,316	29,535	1,432	2,445	4.2	11.4	36.9
2020-03-19	242,500	9,867	84,975	147,658	27,679	1,134	1,663	4.1	10.4	40.5
2020-03-18	214,821	8,733	83,312	122,776	17,719	828	2,472	4.1	9.5	25.1
2020-03-17	197,102	7,905	80,840	108,357	15,528	779	2,752	4.0	8.9	22.1
2020-03-16	181,574	7,126	78,088	96,360	14,120	686	2,054	3.9	8.4	25.0
2020-03-15	167,454	6,440	76,034	84,980	11,353	621	3,410	3.8	7.8	15.4
2020-03-14	156,101	5,819	72,624	77,658	10,896	415	2,373	3.7	7.4	14.9
2020-03-13	145,205	5,404	70,251	69,550	16,853	684	1,927	3.7	7.1	26.2
2020-03-12	128,352	4,720	68,324	55,308	2,477	105	1,321	3.7	6.5	7.4
2020-03-11	125,875	4,615	67,003	54,257	7,255	353	2,599	3.7	6.4	12.0
2020-03-10	118,620	4,262	64,404	49,954	5,030	274	1,910	3.6	6.2	12.5
2020-03-09	113,590	3,988	62,494	47,108	3,769	186	1,800	3.5	6.0	9.4
2020-03-08	109,821	3,802	60,694	45,325	3,974	244	2,336	3.5	5.9	9.5
2020-03-07	105,847	3,558	58,358	43,931	4,046	98	2,493	3.4	5.7	3.8
2020-03-06	101,801	3,460	55,865	42,476	3,915	112	2,069	3.4	5.8	5.1
2020-03-05	97,886	3,348	53,796	40,742	2,766	94	2,626	3.4	5.9	3.5
2020-03-04	95,120	3,254	51,170	40,696	2,280	94	2,942	3.4	6.0	3.1
2020-03-03	92,840	3,160	48,228	41,452	2,534	75	2,626	3.4	6.1	2.8
2020-03-02	90,306	3,085	45,602	41,619	1,937	89	2,886	3.4	6.3	3.0
2020-03-01	88,369	2,996	42,716	42,657	2,358	55	2,934	3.4	6.6	1.8
2020-02-29	86,011	2,941	39,782	43,288	1,899	69	3,071	3.4	6.9	2.2
2020-02-28	84,112	2,872	36,711	44,529	1,366	58	3,434	3.4	7.3	1.7
2020-02-27	82,746	2,814	33,277	46,655	1,358	44	2,893	3.4	7.8	1.5
2020-02-26	81,388	2,770	30,384	48,234	982	62	2,479	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	508	3.1	9.5	2.1
2020-02-22	78,572	2,458	22,886	53,228	1,753	207	3,996	3.1	9.7	4.9
2020-02-21	76,819	2,251	18,890	55,678	622	4	713	2.9	10.6	0.6
2020-02-20	76,197	2,247	18,177	55,773	558	125	2,056	2.9	11.0	5.7
2020-02-19	75,639	2,122	16,121	57,396	503	115	1,769	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	702	2.4	20.4	13.2
2020-02-09	40,150	906	3,244	36,000	3,030	100	628	2.3	21.8	13.7
2020-02-08	37,120	806	2,616	33,698	2,729	87	605	2.2	23.6	12.6
2020-02-07	34,391	719	2,011	31,661	3,597	85	524	2.1	26.3	14.0
2020-02-06	30,794	634	1,487	28,673	3,159	70	363	2.1	29.9	16.2
2020-02-05	27,635	564	1,124	25,947	3,743	72	272	2.0	33.4	20.9
2020-02-04	23,892	492	852	22,548	4,011	66	229	2.1	36.6	22.4
2020-02-03	19,881	426	623	18,832	3,094	64	151	2.1	40.6	29.8
2020-02-02	16,787	362	472	15,953	4,749	103	188	2.2	43.4	35.4
2020-02-01	12,038	259	284	11,495	2,111	46	62	2.2	47.7	42.6
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 1000+ confirmed cases
data.latest.all %>% arrange(desc(confirmed)) %>% select(-c(date, ranking)) %>%
  mutate(death.rate = cell_spec(death.rate, "latex",
                                color = ifelse(confirmed >= 1000 & death.rate >= 5, "red", "black"),
                                bold = ifelse(confirmed >= 1000 & 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 (29 Mar 2020)

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	World	720,117	59,411	537,110	149,082	33,925	3,273	4.7
2	US	140,886	19,408	135,754	2,665	2,467	441	1.8
3	Italy	97,689	5,217	73,880	13,030	10,779	756	**11**
4	China	82,122	123	3,236	75,582	3,304	5	4
5	Spain	80,110	6,875	58,598	14,709	6,803	821	**8.5**
6	Germany	62,095	4,400	52,351	9,211	533	100	0.9
7	France	40,708	2,603	30,871	7,226	2,611	294	**6.4**
8	Iran	38,309	2,901	23,278	12,391	2,640	123	**6.9**
9	United Kingdom	19,780	2,468	18,398	151	1,231	210	**6.2**
10	Switzerland	14,829	753	12,934	1,595	300	36	2
11	Netherlands	10,930	1,111	9,905	253	772	132	**7.1**
12	Belgium	10,836	1,702	9,046	1,359	431	78	4
13	Korea, South	9,583	105	4,398	5,033	152	8	1.6
14	Turkey	9,217	1,815	8,981	105	131	23	1.4
15	Austria	8,788	517	8,223	479	86	18	1
16	Canada	6,280	704	5,750	466	64	3	1
17	Portugal	5,962	792	5,800	43	119	19	2
18	Norway	4,284	269	4,252	7	25	2	0.6
19	Brazil	4,256	352	4,114	6	136	25	3.2
20	Israel	4,247	628	4,100	132	15	3	0.4
21	Australia	3,984	344	3,724	244	16	2	0.4
22	Sweden	3,700	253	3,574	16	110	5	3
23	Czechia	2,817	186	2,790	11	16	5	0.6
24	Ireland	2,615	200	2,564	5	46	10	1.8
25	Denmark	2,564	198	2,419	73	72	7	2.8
26	Malaysia	2,470	150	2,047	388	35	8	1.4
27	Chile	2,139	230	2,057	75	7	1	0.3
28	Luxembourg	1,950	119	1,889	40	21	3	1.1
29	Ecuador	1,924	101	1,863	3	58	10	3
30	Japan	1,866	173	1,388	424	54	2	2.9
31	Poland	1,862	224	1,833	7	22	4	1.2
32	Romania	1,815	363	1,566	206	43	6	2.4
33	Pakistan	1,597	102	1,554	29	14	2	0.9
34	Russia	1,534	270	1,462	64	8	4	0.5
35	Philippines	1,418	343	1,305	42	71	3	**5**
36	Thailand	1,388	143	1,284	97	7	1	0.5
37	Saudi Arabia	1,299	96	1,225	66	8	4	0.6
38	Indonesia	1,285	130	1,107	64	114	12	**8.9**
39	South Africa	1,280	93	1,247	31	2	1	0.2
40	Finland	1,240	73	1,219	10	11	2	0.9
41	Greece	1,156	95	1,066	52	38	6	3.3
42	India	1,024	37	902	95	27	3	2.6
43	Iceland	1,020	57	883	135	2	0	0.2
44	Panama	901	115	880	4	17	3	1.9
45	Dominican Republic	859	140	817	3	39	11	4.5
46	Peru	852	181	818	16	18	2	2.1
47	Mexico	848	131	828	4	16	4	1.9
48	Singapore	844	42	629	212	3	1	0.4
49	Argentina	745	55	654	72	19	1	2.6
50	Serbia	741	82	728	0	13	3	1.8
51	Slovenia	730	46	709	10	11	2	1.5
52	Croatia	713	56	655	52	6	1	0.8

Table 6: Cases by Country (29 Mar 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
53	Diamond Princess	712	0	99	603	10	0	1.4
54	Colombia	702	94	682	10	10	4	1.4
55	Estonia	679	34	656	20	3	2	0.4
56	Qatar	634	44	585	48	1	0	0.2
57	Egypt	609	33	437	132	40	4	6.6
58	United Arab Emirates	570	102	509	58	3	1	0.5
59	Iraq	547	41	362	143	42	0	7.7
60	New Zealand	514	63	457	56	1	1	0.2
61	Algeria	511	57	449	31	31	2	6.1
62	Bahrain	499	23	223	272	4	0	0.8
63	Morocco	479	77	440	13	26	1	5.4
64	Ukraine	475	119	459	6	10	1	2.1
65	Lithuania	460	66	452	1	7	0	1.5
66	Lebanon	438	26	398	30	10	2	2.3
67	Armenia	424	17	391	30	3	2	0.7
68	Hungary	408	65	361	34	13	2	3.2
69	Latvia	347	42	346	1	0	0	0
70	Bulgaria	346	15	324	14	8	1	2.3
71	Andorra	334	26	327	1	6	3	1.8
72	Bosnia and Herzegovina	323	65	309	8	6	1	1.9
73	Costa Rica	314	19	309	3	2	0	0.6
74	Slovakia	314	22	312	2	0	0	0
75	Tunisia	312	34	302	2	8	0	2.6
76	Uruguay	304	30	303	0	1	1	0.3
77	Taiwan*	298	15	266	30	2	0	0.7
78	Kazakhstan	284	56	263	20	1	0	0.4
79	Moldova	263	32	259	2	2	0	0.8
80	Jordan	259	13	238	18	3	2	1.2
81	North Macedonia	259	18	250	3	6	2	2.3
82	Kuwait	255	20	188	67	0	0	0
83	San Marino	224	0	196	6	22	0	9.8
84	Burkina Faso	222	15	187	23	12	1	5.4
85	Cyprus	214	35	194	15	5	0	2.3
86	Albania	212	15	169	33	10	0	4.7
87	Azerbaijan	209	27	190	15	4	0	1.9
88	Vietnam	188	14	163	25	0	0	0
89	Oman	167	15	144	23	0	0	0
90	Cote d'Ivoire	165	64	160	4	1	1	0.6
91	Ghana	152	11	145	2	5	0	3.3
92	Malta	151	2	149	2	0	0	0
93	Uzbekistan	144	40	135	7	2	0	1.4
94	Senegal	142	12	115	27	0	0	0
95	Cameroon	139	48	128	5	6	4	4.3
96	Cuba	139	20	132	4	3	0	2.2
97	Brunei	126	6	91	34	1	0	0.8
98	Afghanistan	120	10	114	2	4	0	3.3
99	Venezuela	119	0	78	39	2	0	1.7
100	Sri Lanka	117	4	105	11	1	0	0.9
101	Nigeria	111	22	107	3	1	0	0.9
102	Honduras	110	15	104	3	3	2	2.7
103	West Bank and Gaza	109	11	90	18	1	0	0.9
104	Mauritius	107	5	104	0	3	1	2.8
105	Cambodia	103	4	82	21	0	0	0
106	Belarus	94	0	62	32	0	0	0
107	Kosovo	94	3	92	1	1	0	1.1
108	Georgia	91	1	73	18	0	0	0
109	Montenegro	85	1	84	0	1	0	1.2
110	Kyrgyzstan	84	26	84	0	0	0	0
111	Bolivia	81	7	80	0	1	1	1.2
112	Trinidad and Tobago	78	4	74	1	3	0	3.8
113	Rwanda	70	10	70	0	0	0	0
114	Congo (Kinshasa)	65	0	57	2	6	0	9.2
115	Paraguay	59	3	55	1	3	0	5.1
116	Liechtenstein	56	0	56	0	0	0	0
117	Bangladesh	48	0	28	15	5	0	10.4
118	Monaco	46	4	44	1	1	1	2.2
119	Kenya	42	4	40	1	1	0	2.4
120	Madagascar	39	13	39	0	0	0	0

Table 6: Cases by Country (29 Mar 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
121	Guatemala	34	0	23	10	1	0	2.9
122	Barbados	33	7	33	0	0	0	0
123	Uganda	33	3	33	0	0	0	0
124	Jamaica	32	2	29	2	1	0	3.1
125	Zambia	29	1	29	0	0	0	0
126	Togo	25	0	23	1	1	0	4
127	El Salvador	24	5	24	0	0	0	0
128	Ethiopia	21	5	20	1	0	0	0
129	Congo (Brazzaville)	19	15	19	0	0	0	0
130	Djibouti	18	4	18	0	0	0	0
131	Mali	18	0	17	0	1	1	5.6
132	Niger	18	8	17	0	1	0	5.6
133	Maldives	17	1	4	13	0	0	0
134	Guinea	16	8	16	0	0	0	0
135	Haiti	15	7	14	1	0	0	0
136	Tanzania	14	0	13	1	0	0	0
137	Equatorial Guinea	12	0	12	0	0	0	0
138	Eritrea	12	6	12	0	0	0	0
139	Mongolia	12	0	12	0	0	0	0
140	Bahamas	11	1	10	1	0	0	0
141	Dominica	11	0	11	0	0	0	0
142	Namibia	11	3	9	2	0	0	0
143	Burma	10	2	10	0	0	0	0
144	Eswatini	9	0	9	0	0	0	0
145	Grenada	9	2	9	0	0	0	0
146	Saint Lucia	9	6	8	1	0	0	0
147	Syria	9	4	8	0	1	1	11.1
148	Guyana	8	0	7	0	1	0	12.5
149	Laos	8	0	8	0	0	0	0
150	Libya	8	5	8	0	0	0	0
151	Mozambique	8	0	8	0	0	0	0
152	Seychelles	8	0	8	0	0	0	0
153	Suriname	8	0	8	0	0	0	0
154	Angola	7	2	5	0	2	2	28.6
155	Antigua and Barbuda	7	0	7	0	0	0	0
156	Gabon	7	0	6	0	1	0	14.3
157	Zimbabwe	7	0	6	0	1	0	14.3
158	Benin	6	0	6	0	0	0	0
159	Cabo Verde	6	1	5	0	1	0	16.7
160	Holy See	6	0	6	0	0	0	0
161	Sudan	6	1	5	0	1	0	16.7
162	Fiji	5	0	5	0	0	0	0
163	Mauritania	5	0	3	2	0	0	0
164	Nepal	5	0	4	1	0	0	0
165	Bhutan	4	1	4	0	0	0	0
166	Gambia	4	1	3	0	1	0	25
167	Nicaragua	4	0	3	0	1	0	25
168	Central African Republic	3	0	3	0	0	0	0
169	Chad	3	0	3	0	0	0	0
170	Liberia	3	0	3	0	0	0	0
171	Somalia	3	0	3	0	0	0	0
172	Belize	2	0	2	0	0	0	0
173	Guinea-Bissau	2	0	2	0	0	0	0
174	MS Zaandam	2	0	2	0	0	0	0
175	Saint Kitts and Nevis	2	0	2	0	0	0	0
176	Papua New Guinea	1	0	1	0	0	0	0
177	Saint Vincent and the Grenadines	1	0	0	1	0	0	0
178	Timor-Leste	1	0	1	0	0	0	0

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!