COVID-19 Data Analysis with R - Worldwide

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

08 April 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] 263  80
```

Each dataset has 263 rows, corresponding to country/region/province/state. It has 80 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" "2020-04-06"
```

```
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 06 Apr 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 184 countries with confirmed COVID-19 cases, as of 06 Apr 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
```

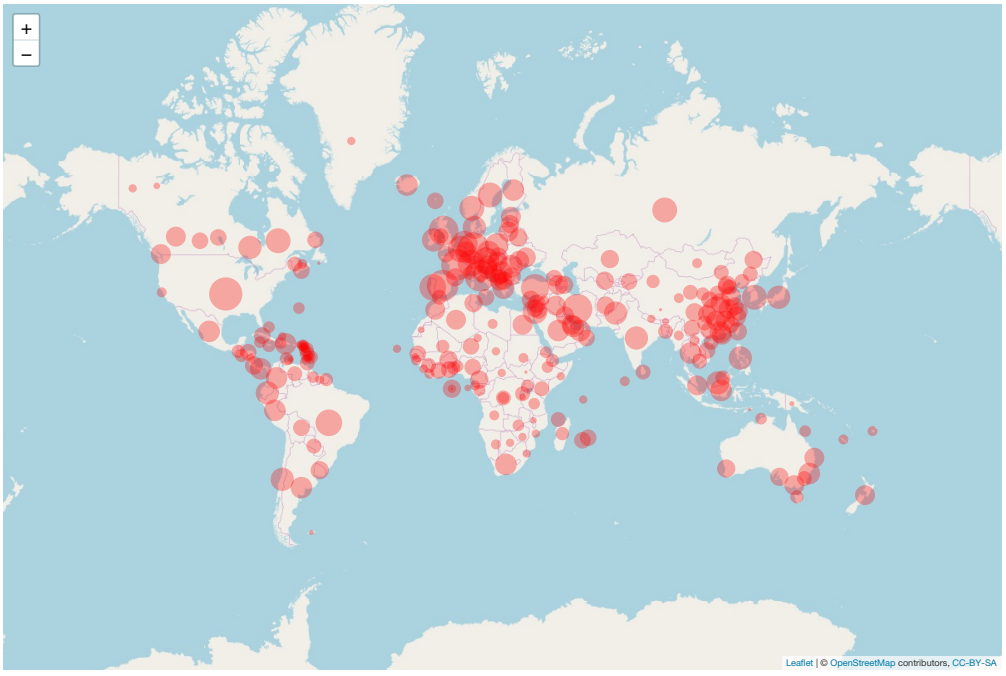


Figure 1: World Map

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)
```

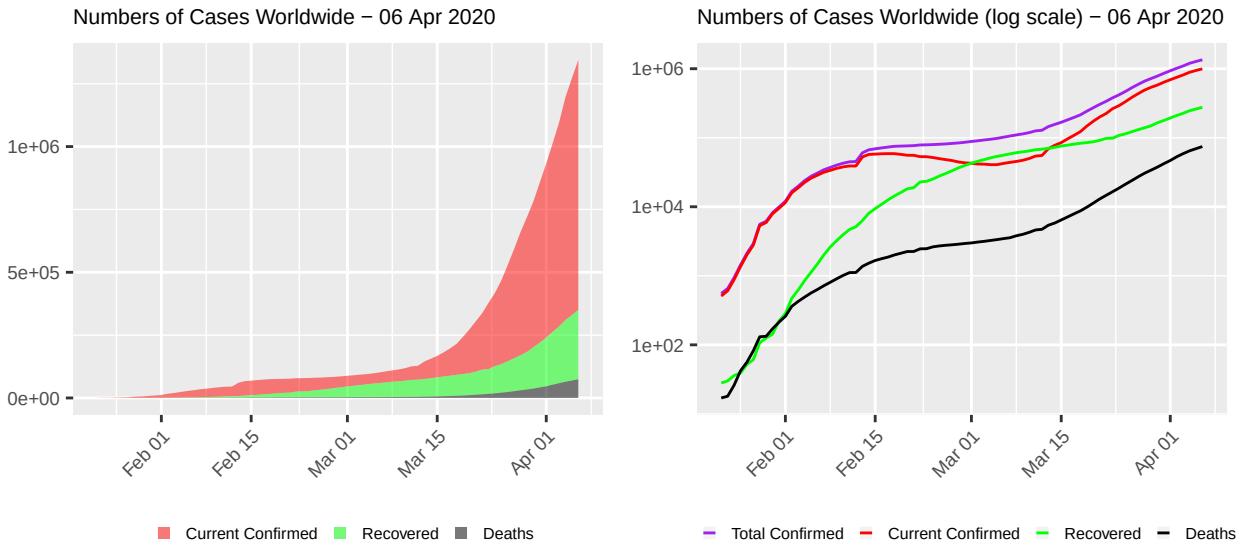


Figure 2: COVID-19 Cases Worldwide

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)
```

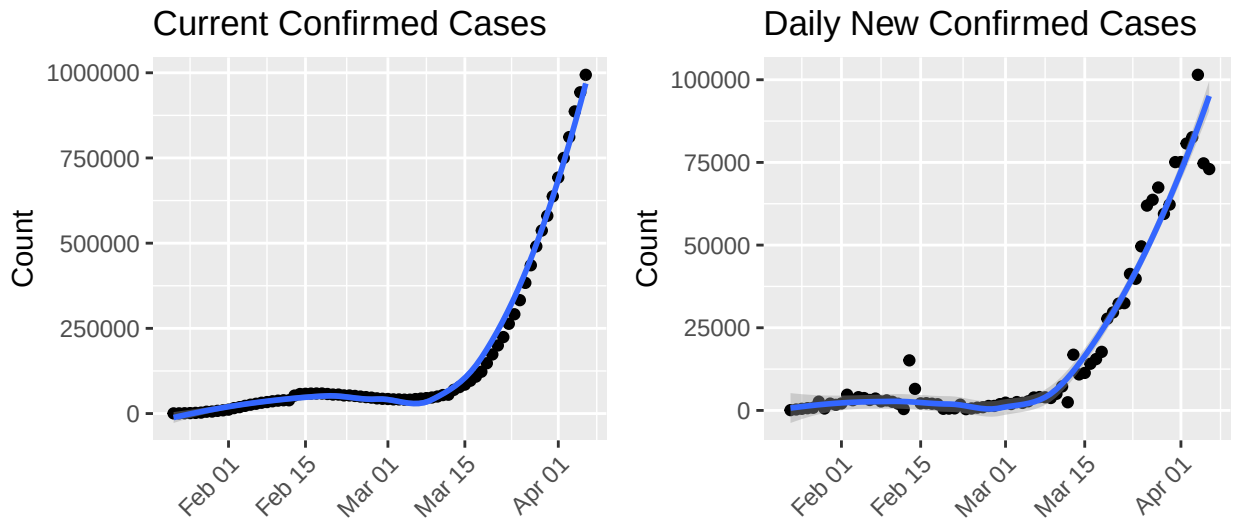


Figure 3: Current Confirmed Cases

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)
```

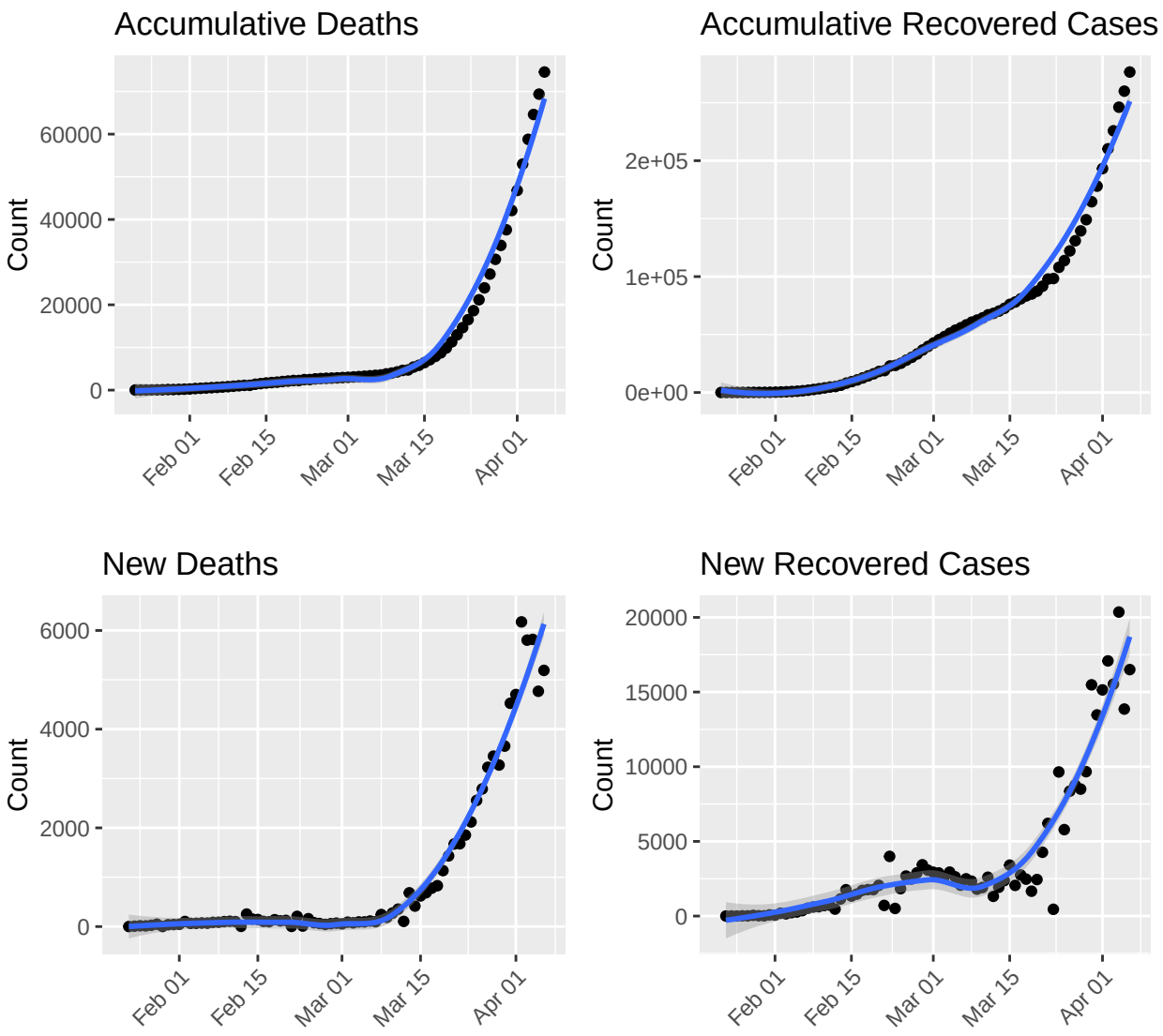


Figure 4: Deaths and Recovered Cases

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 06 Apr 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 06 Apr 2020 UTC, it will be between 5.5% and 21.2%.

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)
```

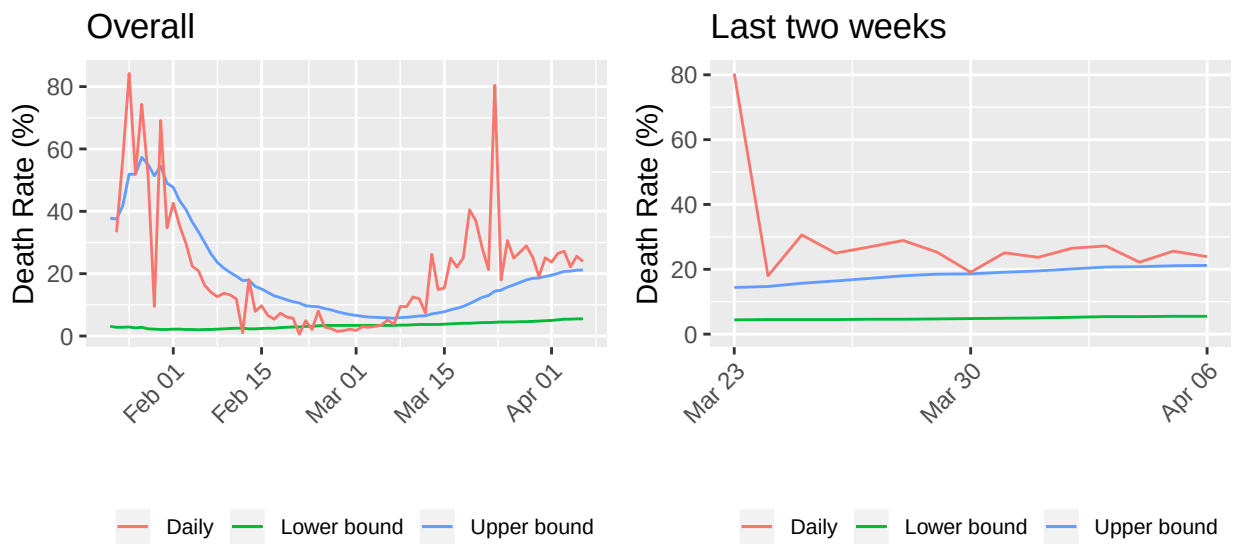


Figure 5: Death Rate

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)))
```

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

```r
## 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))
```

```r
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'))
```

```r
## 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'))
```

```r
## 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 - 06 Apr 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	1,345,101	74,565	5.5%	72,986	5,191	994,021
2	US	366,667	10,783	2.9%	29,595	1,164	336,303
3	Spain	136,675	13,341	9.8%	5,029	700	82,897
4	Italy	132,547	16,523	12.5%	3,599	636	93,187
5	Germany	103,374	1,810	1.8%	3,251	226	72,864
6	France	98,963	8,926	9.0%	5,190	833	72,609
7	China	82,665	3,335	4.0%	63	2	2,020
8	Iran	60,500	3,739	6.2%	2,274	136	32,525
9	United Kingdom	52,279	5,385	10.3%	3,843	442	46,607
10	Turkey	30,217	649	2.1%	3,148	75	28,242
11	Switzerland	21,657	765	3.5%	557	50	12,836
12	Belgium	20,814	1,632	7.8%	1,123	185	15,196
13	Netherlands	18,926	1,874	9.9%	973	103	16,794
14	Canada	16,563	339	2.0%	807	80	12,968
15	Austria	12,297	220	1.8%	246	16	8,614
16	Brazil	12,161	564	4.6%	1,031	78	11,470
17	Portugal	11,730	311	2.7%	452	16	11,279
18	Korea, South	10,284	186	1.8%	47	3	3,500
19	Israel	8,904	57	0.6%	474	8	8,262
20	Sweden	7,206	477	6.6%	376	76	6,524
21	Russia	6,343	47	0.7%	954	2	5,890
22	Others	134,329	3,602	2.7%	9,954	361	113,434

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

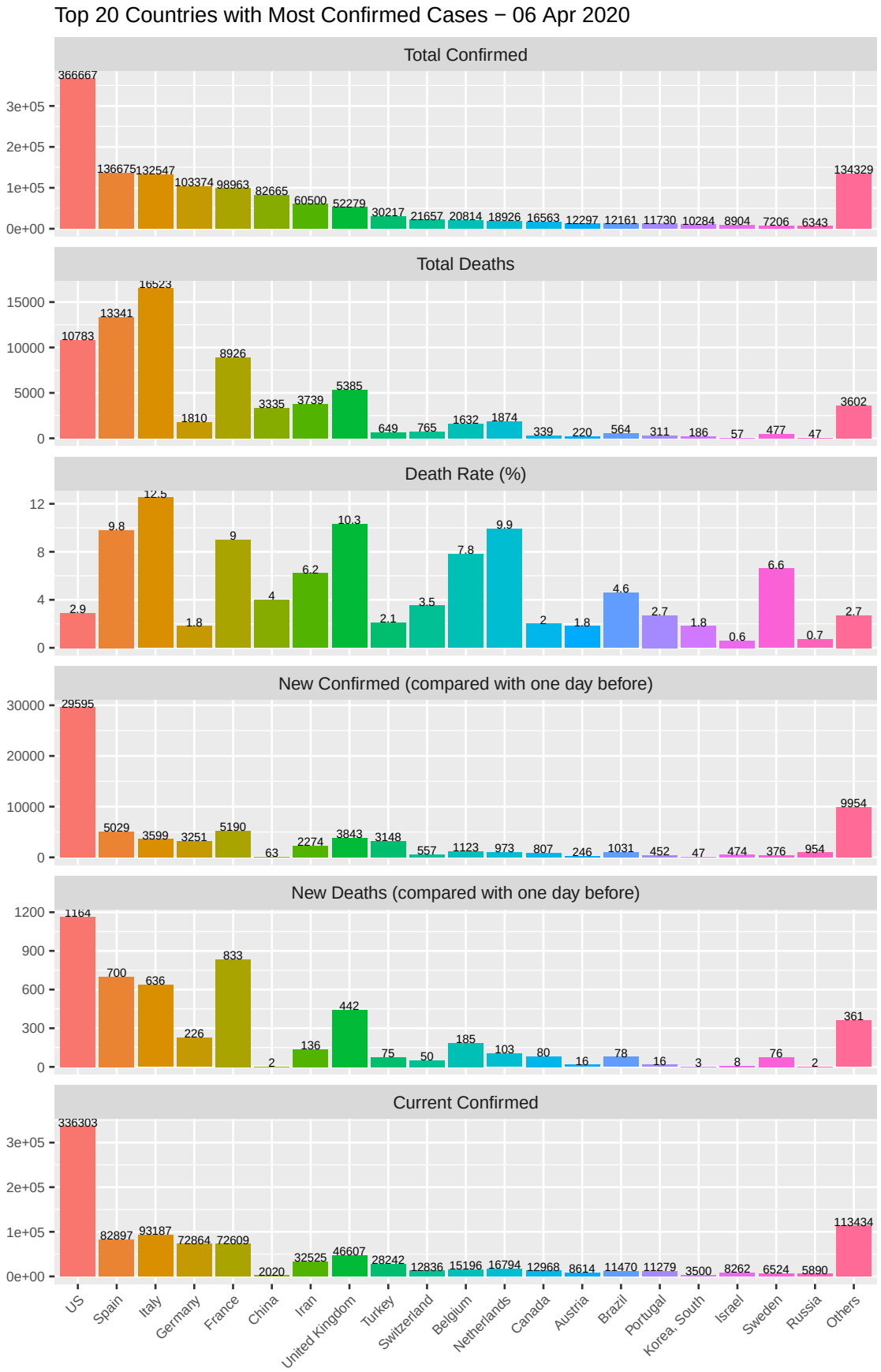
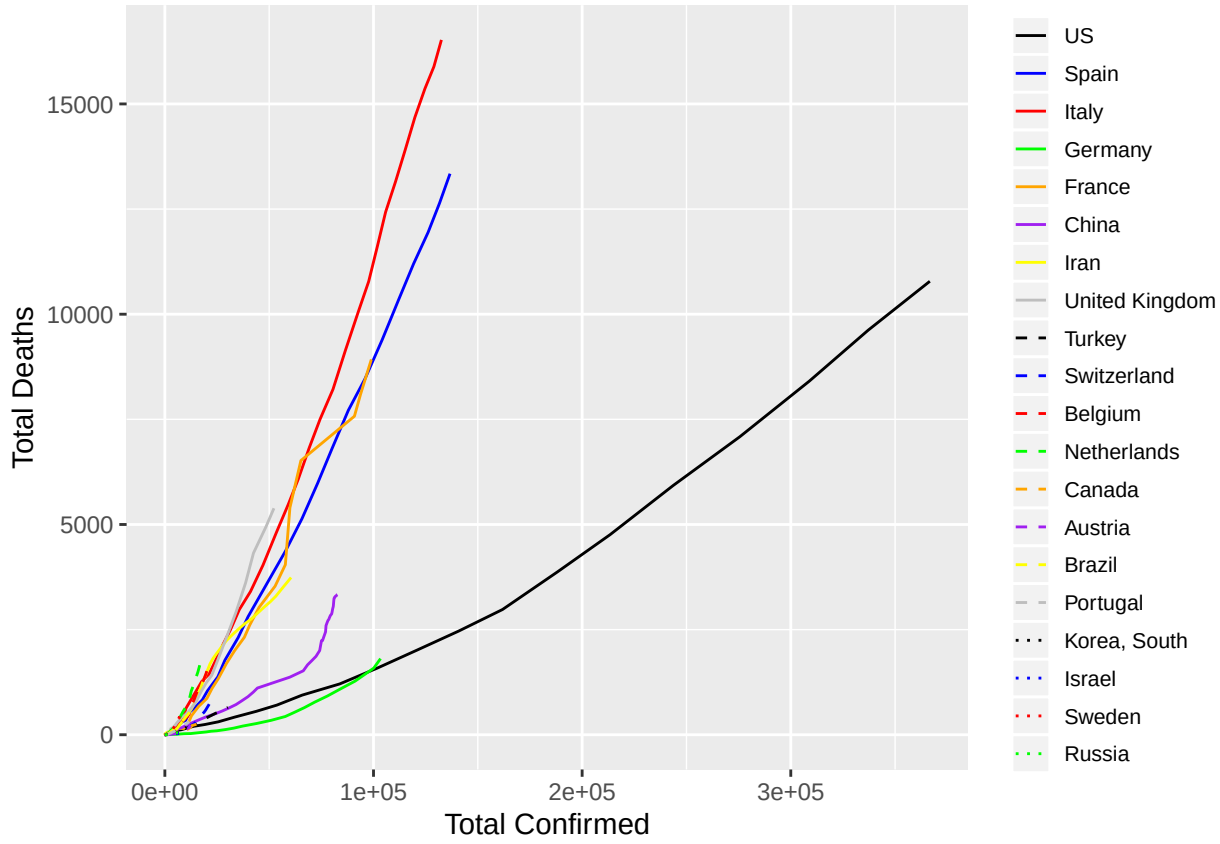


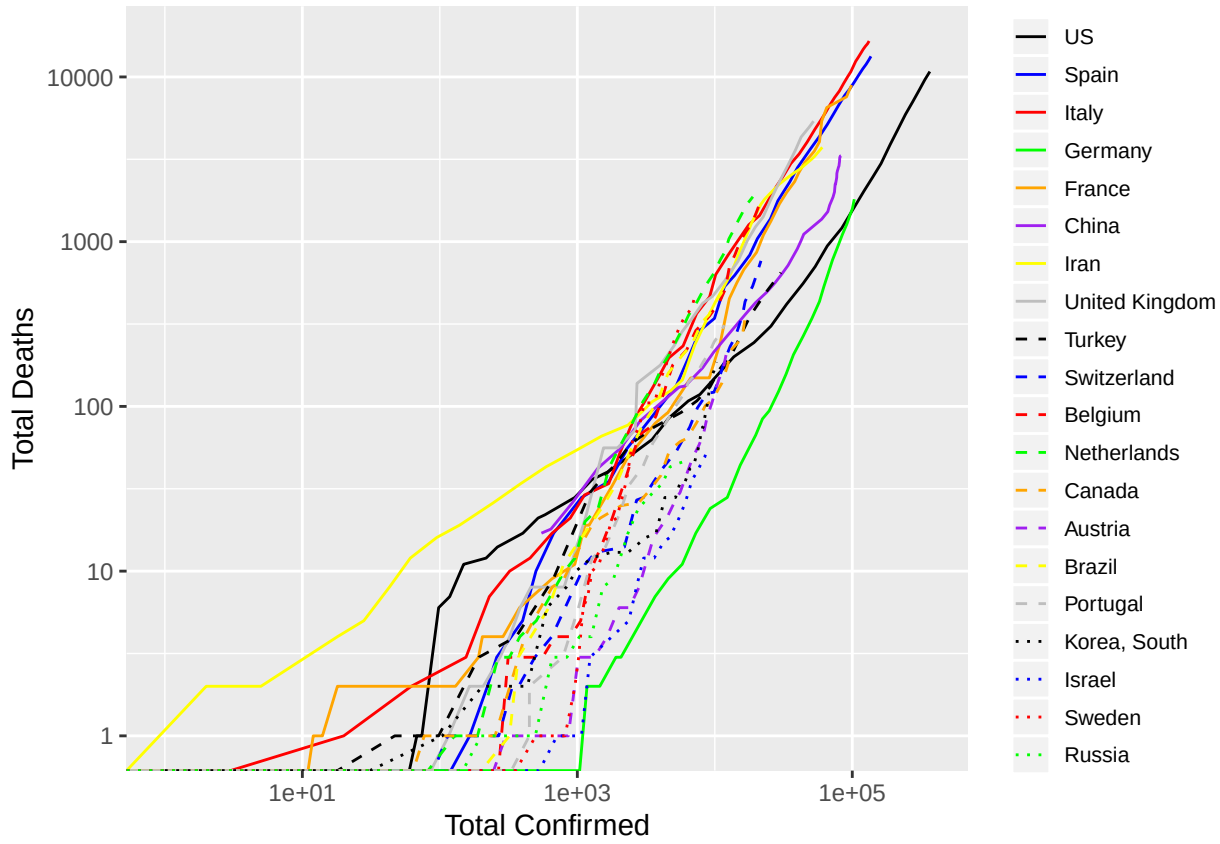
Figure 6: Top 20 Countries with Most Confirmed Cases

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 06 Apr 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)
```

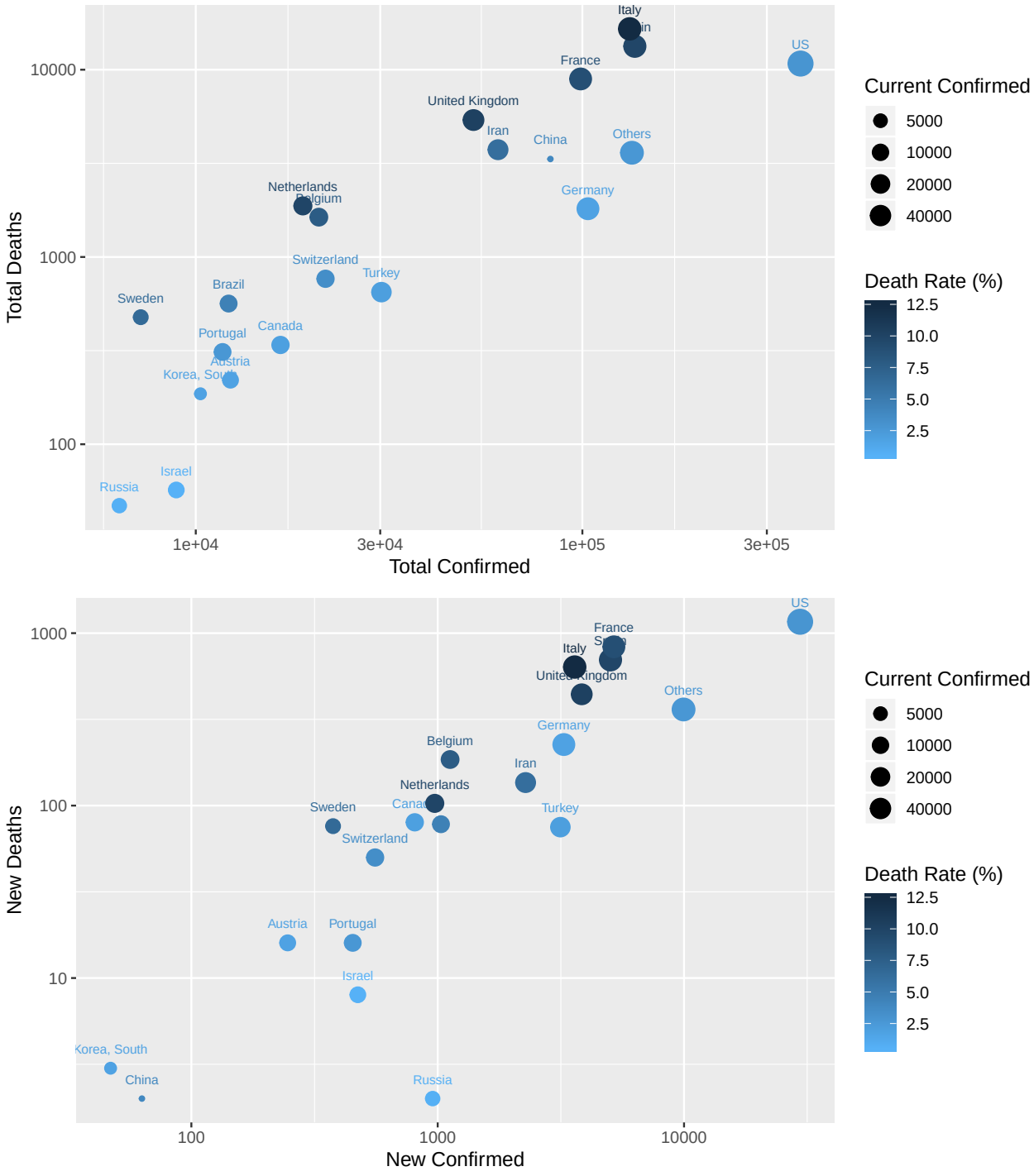


Figure 7: Top 20 Countries

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)
```

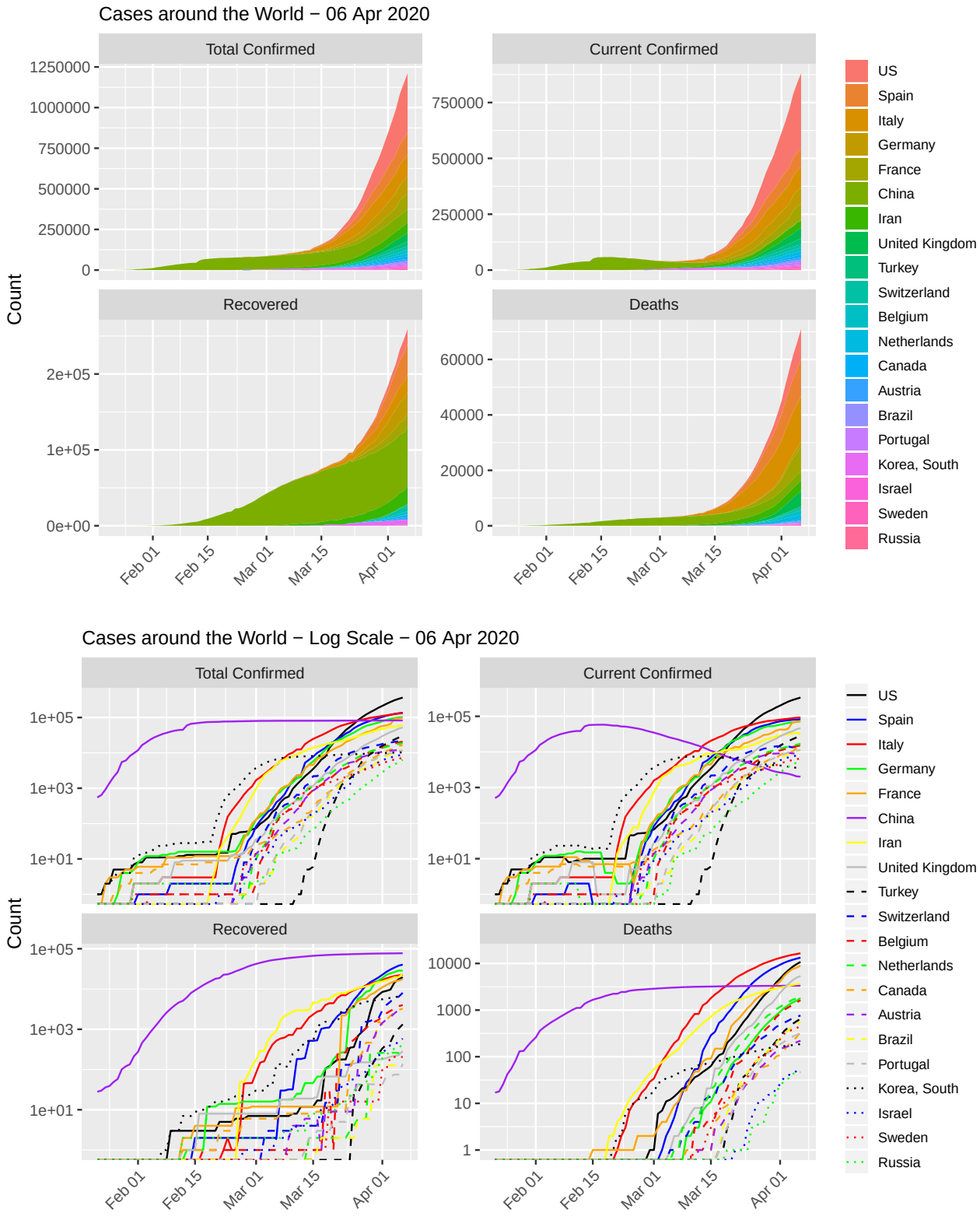


Figure 8: Cases around the World

```
## 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))
```

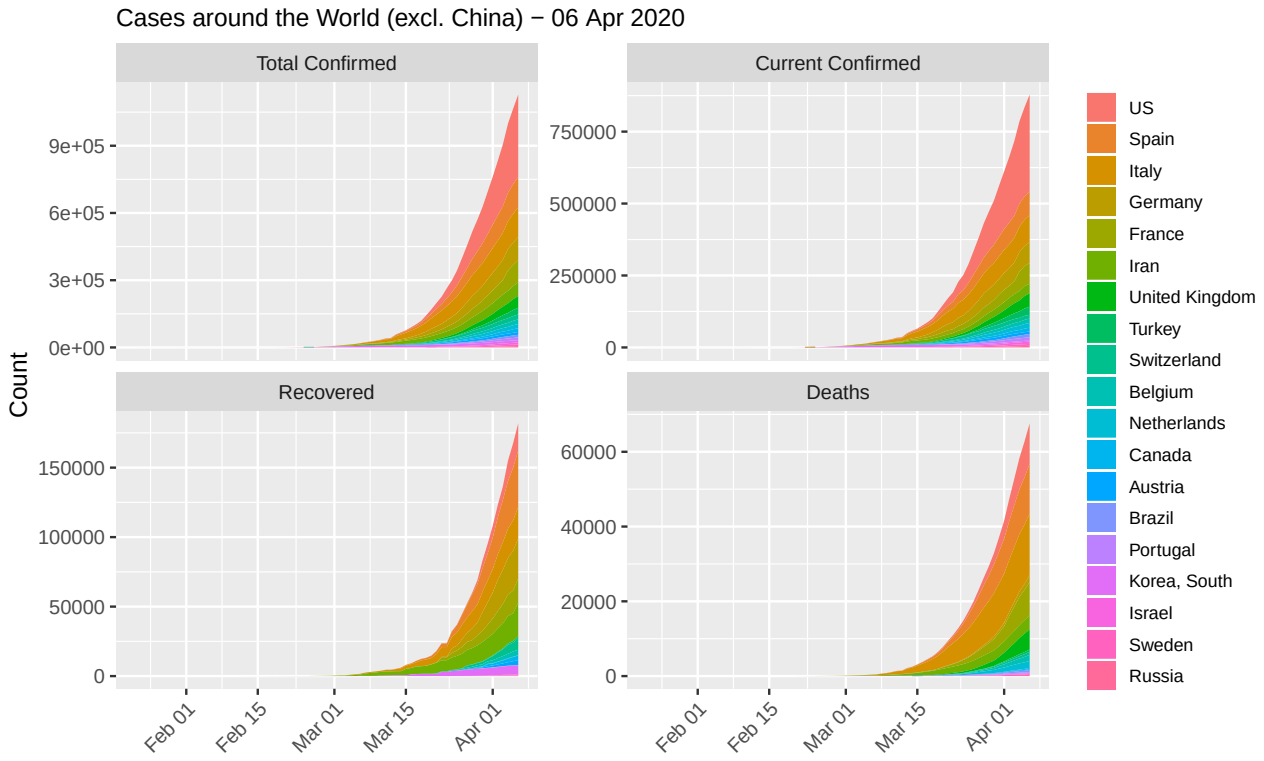


Figure 9: Cases around the World (excl. China)

```
## 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')
```

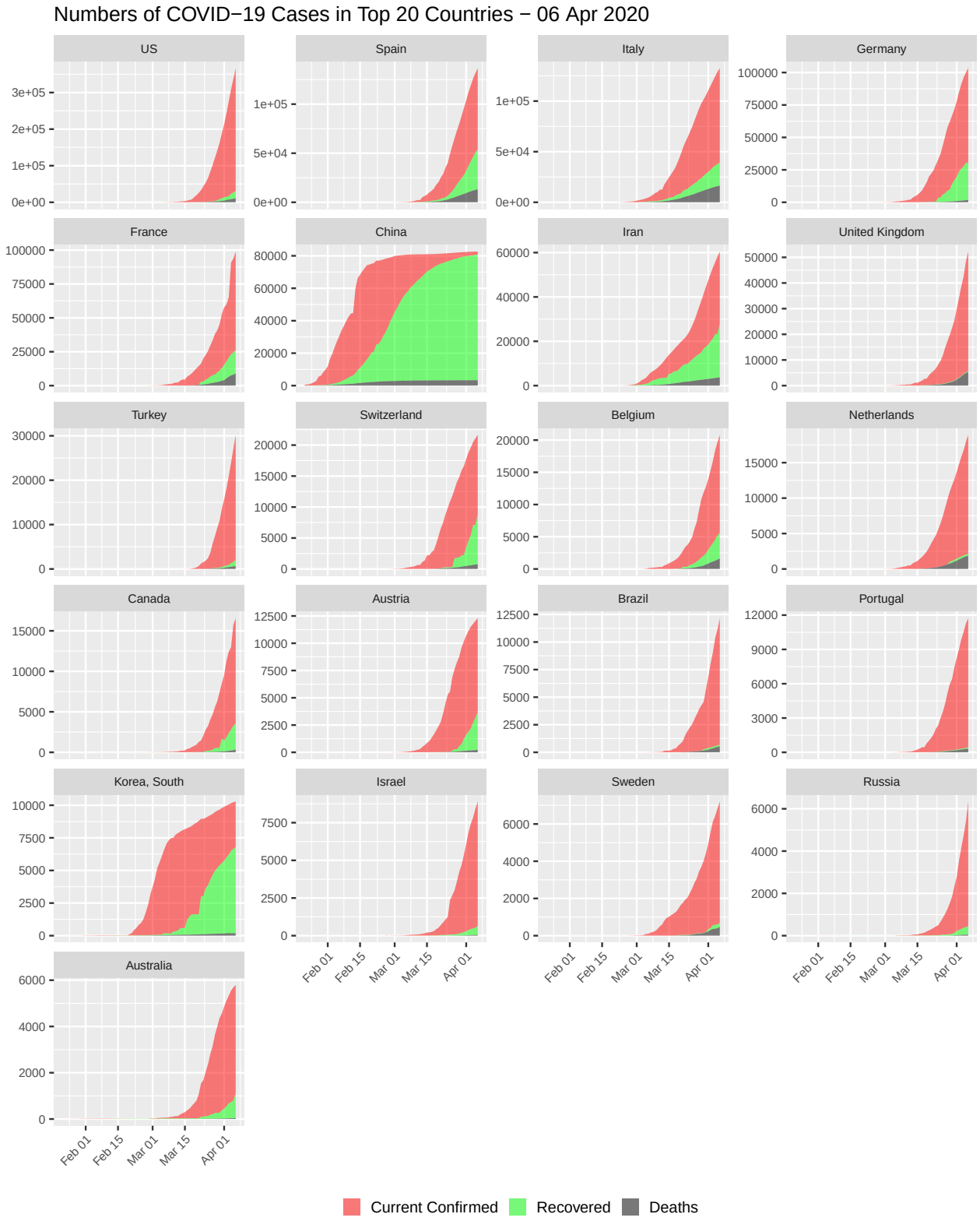


Figure 10: COVID-19 Cases in Top 20 Countries. Ordered descendingly by number of confirmed cases.

```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')
```

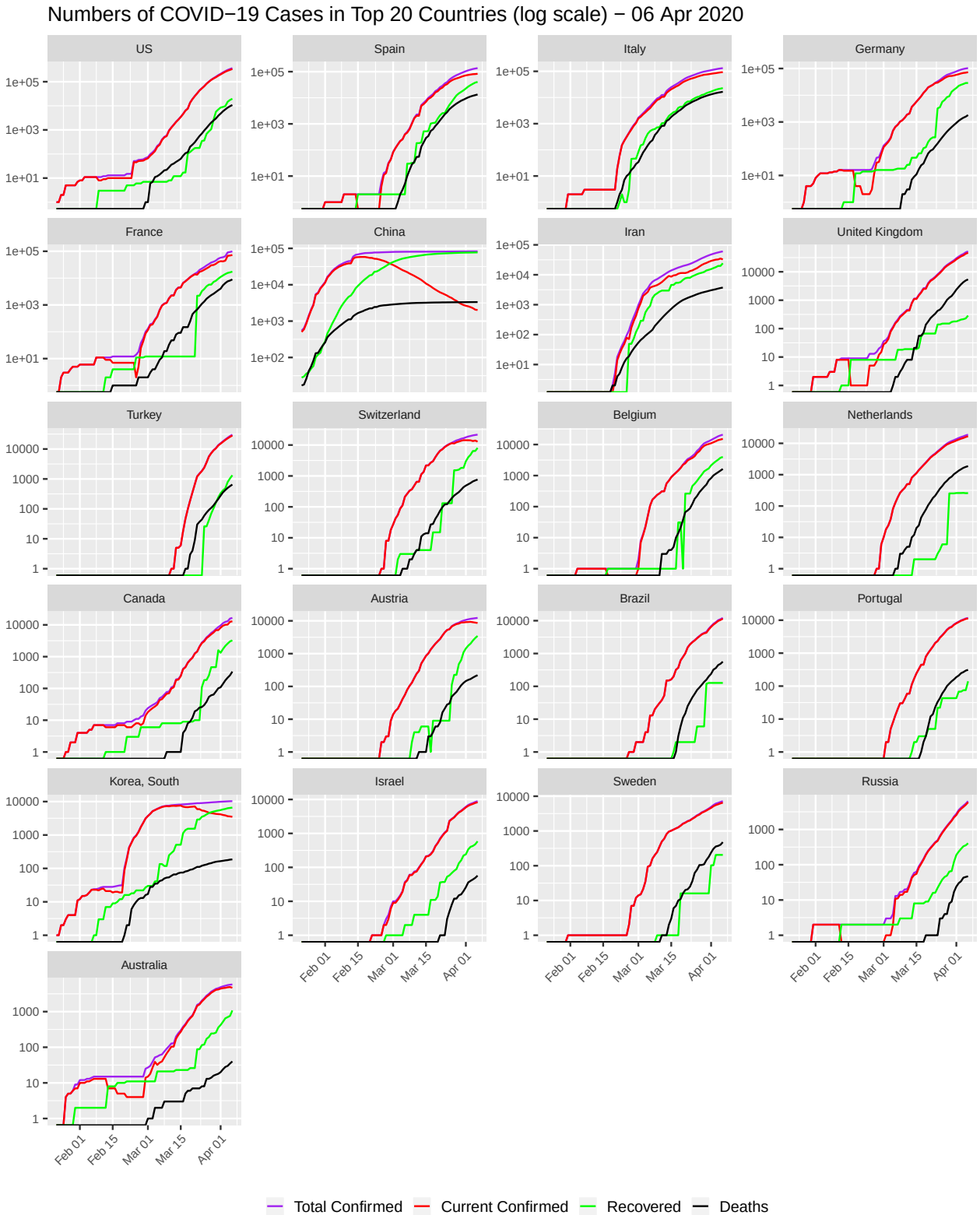


Figure 11: COVID-19 Cases Top 20 Countries (log scale). Ordered descendingly by number of confirmed cases.

```
## 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)
```

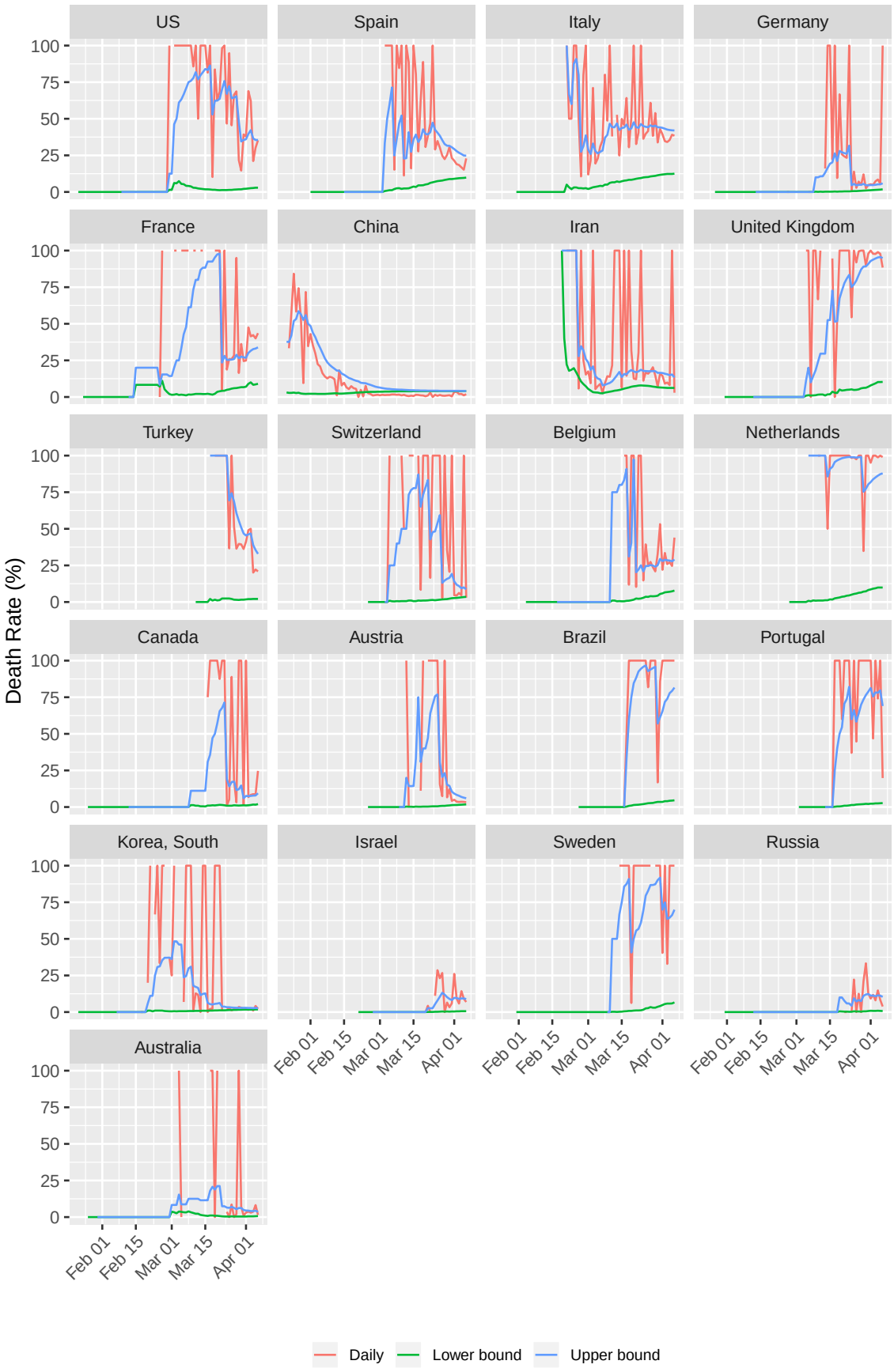


Figure 12: Death Rates

5.4 Countries with Highest Death Rates

Below are a list of top 20 countries 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 - 06 Apr 2020

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	Italy	132,547	3,599	93,187	22,837	16,523	636	12.5%
2	Algeria	1,423	103	1,160	90	173	21	12.2%
3	United Kingdom	52,279	3,843	46,607	287	5,385	442	10.3%
4	Netherlands	18,926	973	16,794	258	1,874	103	9.9%
5	Spain	136,675	5,029	82,897	40,437	13,341	700	9.8%
6	France	98,963	5,190	72,609	17,428	8,926	833	9.0%
7	Indonesia	2,491	218	2,090	192	209	11	8.4%
8	Belgium	20,814	1,123	15,196	3,986	1,632	185	7.8%
9	Morocco	1,120	99	959	81	80	10	7.1%
10	Sweden	7,206	376	6,524	205	477	76	6.6%
11	Egypt	1,322	149	978	259	85	7	6.4%
12	Iran	60,500	2,274	32,525	24,236	3,739	136	6.2%
13	Iraq	1,031	70	623	344	64	3	6.2%
14	Ecuador	3,747	101	3,456	100	191	11	5.1%
15	Dominican Republic	1,828	83	1,709	33	86	4	4.7%
16	Brazil	12,161	1,031	11,470	127	564	78	4.6%
17	Greece	1,755	20	1,407	269	79	6	4.5%
18	Philippines	3,660	414	3,424	73	163	11	4.5%
19	Mexico	2,143	253	1,416	633	94	15	4.4%
20	Romania	4,057	193	3,475	406	176	25	4.3%

6 Conclusions

As of 06 Apr 2020, there are 184 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 5.5% and 21.2%, 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
2020-04-06	1,345,101	74,565	276,515	994,021	72,986	5,191	16,503	5.5	21.2	23.9
2020-04-05	1,272,115	69,374	260,012	942,729	74,710	4,768	13,860	5.5	21.1	25.6
2020-04-04	1,197,405	64,606	246,152	886,647	101,488	5,819	20,356	5.4	20.8	22.2
2020-04-03	1,095,917	58,787	225,796	811,334	82,597	5,804	15,533	5.4	20.7	27.2
2020-04-02	1,013,320	52,983	210,263	750,074	80,715	6,174	17,086	5.2	20.1	26.5
2020-04-01	932,605	46,809	193,177	692,619	75,118	4,702	15,143	5.0	19.5	23.7
2020-03-31	857,487	42,107	178,034	637,346	75,092	4,525	13,468	4.9	19.1	25.1
2020-03-30	782,395	37,582	164,566	580,247	62,255	3,657	15,484	4.8	18.6	19.1
2020-03-29	720,140	33,925	149,082	537,133	59,434	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	32,446	1,678	6,207	4.3	13.0	21.3
2020-03-21	304,507	12,973	91,692	199,842	32,299	1,674	4,272	4.3	12.4	28.2
2020-03-20	272,208	11,299	87,420	173,489	29,638	1,432	2,445	4.2	11.4	36.9
2020-03-19	242,570	9,867	84,975	147,728	27,749	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=','), align=c('l', rep('r', 7))) %>%
  kable_styling(font_size=6, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

Table 6: Cases by Country (06 Apr 2020)

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	World	1,345,101	72,986	994,021	276,515	74,565	5,191	**5.5**
2	US	366,667	29,595	336,303	19,581	10,783	1,164	2.9
3	Spain	136,675	5,029	82,897	40,437	13,341	700	**9.8**
4	Italy	132,547	3,599	93,187	22,837	16,523	636	**12.5**
5	Germany	103,374	3,251	72,864	28,700	1,810	226	1.8
6	France	98,963	5,190	72,609	17,428	8,926	833	**9**
7	China	82,665	63	2,020	77,310	3,335	2	4
8	Iran	60,500	2,274	32,525	24,236	3,739	136	**6.2**
9	United Kingdom	52,279	3,843	46,607	287	5,385	442	**10.3**
10	Turkey	30,217	3,148	28,242	1,326	649	75	2.1
11	Switzerland	21,657	557	12,836	8,056	765	50	3.5
12	Belgium	20,814	1,123	15,196	3,986	1,632	185	**7.8**
13	Netherlands	18,926	973	16,794	258	1,874	103	**9.9**
14	Canada	16,563	807	12,968	3,256	339	80	2
15	Austria	12,297	246	8,614	3,463	220	16	1.8
16	Brazil	12,161	1,031	11,470	127	564	78	4.6
17	Portugal	11,730	452	11,279	140	311	16	2.7
18	Korea, South	10,284	47	3,500	6,598	186	3	1.8
19	Israel	8,904	474	8,262	585	57	8	0.6
20	Sweden	7,206	376	6,524	205	477	76	**6.6**
21	Russia	6,343	954	5,890	406	47	2	0.7
22	Norway	5,865	178	5,757	32	76	5	1.3
23	Australia	5,797	110	4,677	1,080	40	5	0.7
24	Ireland	5,364	370	5,165	25	174	16	3.2
25	Denmark	4,875	314	3,199	1,489	187	8	3.8
26	Czechia	4,822	235	4,623	121	78	11	1.6
27	Chile	4,815	344	4,050	728	37	3	0.8
28	India	4,778	1,190	4,267	375	136	37	2.8
29	Poland	4,413	311	4,144	162	107	13	2.4
30	Romania	4,057	193	3,475	406	176	25	4.3
31	Malaysia	3,793	131	2,490	1,241	62	1	1.6
32	Pakistan	3,766	609	3,454	259	53	6	1.4
33	Ecuador	3,747	101	3,456	100	191	11	**5.1**
34	Philippines	3,660	414	3,424	73	163	11	4.5
35	Japan	3,654	515	2,994	575	85	8	2.3
36	Luxembourg	2,843	39	2,302	500	41	5	1.4
37	Saudi Arabia	2,605	203	2,016	551	38	4	1.5
38	Peru	2,561	280	1,472	997	92	9	3.6
39	Indonesia	2,491	218	2,090	192	209	11	**8.4**
40	Thailand	2,220	51	1,401	793	26	3	1.2
41	Serbia	2,200	292	2,142	0	58	7	2.6
42	Finland	2,176	249	1,849	300	27	0	1.2
43	Mexico	2,143	253	1,416	633	94	15	4.4
44	United Arab Emirates	2,076	277	1,898	167	11	1	0.5
45	Panama	1,988	187	1,921	13	54	8	2.7
46	Qatar	1,832	228	1,697	131	4	0	0.2
47	Dominican Republic	1,828	83	1,709	33	86	4	4.7
48	Greece	1,755	20	1,407	269	79	6	4.5
49	South Africa	1,686	31	1,579	95	12	1	0.7
50	Colombia	1,579	94	1,445	88	46	11	2.9
51	Iceland	1,562	76	1,096	460	6	2	0.4
52	Argentina	1,554	103	1,181	325	48	4	3.1

Table 6: Cases by Country (06 Apr 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
53	Algeria	1,423	103	1,160	90	173	21	**12.2**
54	Singapore	1,375	66	1,025	344	6	0	0.4
55	Egypt	1,322	149	978	259	85	7	**6.4**
56	Ukraine	1,319	11	1,253	28	38	1	2.9
57	Croatia	1,222	40	1,076	130	16	1	1.3
58	Morocco	1,120	99	959	81	80	10	**7.1**
59	Estonia	1,108	11	1,027	62	19	4	1.7
60	New Zealand	1,106	67	929	176	1	0	0.1
61	Iraq	1,031	70	623	344	64	3	**6.2**
62	Slovenia	1,021	24	889	102	30	2	2.9
63	Moldova	965	101	909	37	19	4	2
64	Lithuania	843	32	820	8	15	2	1.8
65	Armenia	833	11	763	62	8	1	1
66	Bahrain	756	56	294	458	4	0	0.5
67	Hungary	744	11	639	67	38	4	5.1
68	Diamond Princess	712	0	82	619	11	0	1.5
69	Belarus	700	138	634	53	13	5	1.9
70	Bosnia and Herzegovina	674	20	598	47	29	6	4.3
71	Kuwait	665	109	561	103	1	0	0.2
72	Kazakhstan	662	78	610	46	6	0	0.9
73	Cameroon	658	8	632	17	9	0	1.4
74	Azerbaijan	641	57	590	44	7	0	1.1
75	Tunisia	596	22	569	5	22	0	3.7
76	North Macedonia	570	15	517	30	23	5	4
77	Bulgaria	549	18	488	39	22	2	4
78	Latvia	542	9	525	16	1	0	0.2
79	Lebanon	541	14	462	60	19	1	3.5
80	Slovakia	534	49	524	8	2	1	0.4
81	Andorra	525	24	473	31	21	3	4
82	Costa Rica	467	13	447	18	2	0	0.4
83	Cyprus	465	19	411	45	9	0	1.9
84	Uzbekistan	457	115	425	30	2	0	0.4
85	Uruguay	406	6	296	104	6	1	1.5
86	Albania	377	16	240	116	21	1	5.6
87	Taiwan*	373	10	311	57	5	0	1.3
88	Afghanistan	367	18	338	18	11	4	3
89	Burkina Faso	364	19	238	108	18	1	4.9
90	Cuba	350	30	323	18	9	1	2.6
91	Jordan	349	4	217	126	6	1	1.7
92	Oman	331	33	268	61	2	0	0.6
93	Cote d'Ivoire	323	62	279	41	3	0	0.9
94	Honduras	298	30	270	6	22	0	7.4
95	San Marino	266	0	199	35	32	0	12
96	West Bank and Gaza	254	17	229	24	1	0	0.4
97	Niger	253	69	217	26	10	0	4
98	Vietnam	245	4	150	95	0	0	0
99	Mauritius	244	17	230	7	7	0	2.9
100	Malta	241	14	236	5	0	0	0
101	Nigeria	238	6	198	35	5	0	2.1
102	Montenegro	233	19	230	1	2	0	0.9
103	Senegal	226	4	132	92	2	0	0.9
104	Kyrgyzstan	216	69	179	33	4	3	1.9
105	Ghana	214	0	178	31	5	0	2.3
106	Georgia	188	14	147	39	2	0	1.1
107	Bolivia	183	26	170	2	11	1	6
108	Sri Lanka	178	2	135	38	5	0	2.8
109	Venezuela	165	6	93	65	7	0	4.2
110	Congo (Kinshasa)	161	7	138	5	18	0	11.2
111	Kenya	158	16	148	4	6	2	3.8
112	Kosovo	145	0	121	23	1	0	0.7
113	Brunei	135	0	52	82	1	0	0.7
114	Guinea	128	7	123	5	0	0	0
115	Bangladesh	123	35	78	33	12	3	9.8
116	Cambodia	114	0	61	53	0	0	0
117	Paraguay	113	9	96	12	5	2	4.4
118	Rwanda	105	1	101	4	0	0	0
119	Trinidad and Tobago	105	1	96	1	8	1	7.6
120	Djibouti	90	31	81	9	0	0	0

Table 6: Cases by Country (06 Apr 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
121	Madagascar	82	10	80	2	0	0	0
122	Liechtenstein	77	0	21	55	1	0	1.3
123	Monaco	77	4	72	4	1	0	1.3
124	Guatemala	70	9	52	15	3	1	4.3
125	El Salvador	69	7	60	5	4	1	5.8
126	Barbados	60	4	52	6	2	1	3.3
127	Jamaica	58	0	47	8	3	0	5.2
128	Togo	58	14	32	23	3	0	5.2
129	Uganda	52	0	52	0	0	0	0
130	Mali	47	2	33	9	5	0	10.6
131	Congo (Brazzaville)	45	0	38	2	5	0	11.1
132	Ethiopia	44	1	38	4	2	0	4.5
133	Zambia	39	0	33	5	1	0	2.6
134	Eritrea	31	2	31	0	0	0	0
135	Guyana	31	7	19	8	4	0	12.9
136	Bahamas	29	1	20	4	5	1	17.2
137	Benin	26	4	20	5	1	1	3.8
138	Gabon	24	3	22	1	1	0	4.2
139	Haiti	24	3	23	0	1	0	4.2
140	Tanzania	24	2	20	3	1	0	4.2
141	Burma	22	1	21	0	1	0	4.5
142	Libya	19	1	17	1	1	0	5.3
143	Maldives	19	0	6	13	0	0	0
144	Syria	19	0	15	2	2	0	10.5
145	Guinea-Bissau	18	0	18	0	0	0	0
146	Angola	16	2	12	2	2	0	12.5
147	Equatorial Guinea	16	0	13	3	0	0	0
148	Namibia	16	0	13	3	0	0	0
149	Antigua and Barbuda	15	0	15	0	0	0	0
150	Dominica	15	1	14	1	0	0	0
151	Mongolia	15	1	13	2	0	0	0
152	Fiji	14	2	14	0	0	0	0
153	Liberia	14	1	8	3	3	0	21.4
154	Saint Lucia	14	0	13	1	0	0	0
155	Grenada	12	0	12	0	0	0	0
156	Laos	12	1	12	0	0	0	0
157	Sudan	12	0	8	2	2	0	16.7
158	Seychelles	11	1	11	0	0	0	0
159	Eswatini	10	1	6	4	0	0	0
160	Mozambique	10	0	9	1	0	0	0
161	Saint Kitts and Nevis	10	0	10	0	0	0	0
162	Suriname	10	0	9	0	1	0	10
163	Zimbabwe	10	1	9	0	1	0	10
164	Chad	9	0	9	0	0	0	0
165	MS Zaandam	9	0	7	0	2	0	22.2
166	Nepal	9	0	8	1	0	0	0
167	Central African Republic	8	0	8	0	0	0	0
168	Belize	7	2	6	0	1	1	14.3
169	Cabo Verde	7	0	5	1	1	0	14.3
170	Holy See	7	0	7	0	0	0	0
171	Saint Vincent and the Grenadines	7	0	6	1	0	0	0
172	Somalia	7	0	6	1	0	0	0
173	Botswana	6	0	5	0	1	0	16.7
174	Mauritania	6	0	3	2	1	0	16.7
175	Nicaragua	6	0	5	0	1	0	16.7
176	Sierra Leone	6	0	6	0	0	0	0
177	Bhutan	5	0	3	2	0	0	0
178	Malawi	5	1	5	0	0	0	0
179	Gambia	4	0	1	2	1	0	25
180	Sao Tome and Principe	4	4	4	0	0	0	0
181	Western Sahara	4	0	4	0	0	0	0
182	Burundi	3	0	3	0	0	0	0
183	Papua New Guinea	2	1	2	0	0	0	0
184	South Sudan	1	0	1	0	0	0	0
185	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!