COVID-19 Data Analysis with R - Worldwide

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

06 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] 259  78
```

Each dataset has 259 rows, corresponding to country/region/province/state. It has 78 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" "2020-04-04"
```

```
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 04 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 181 countries with confirmed COVID-19 cases, as of 04 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
```

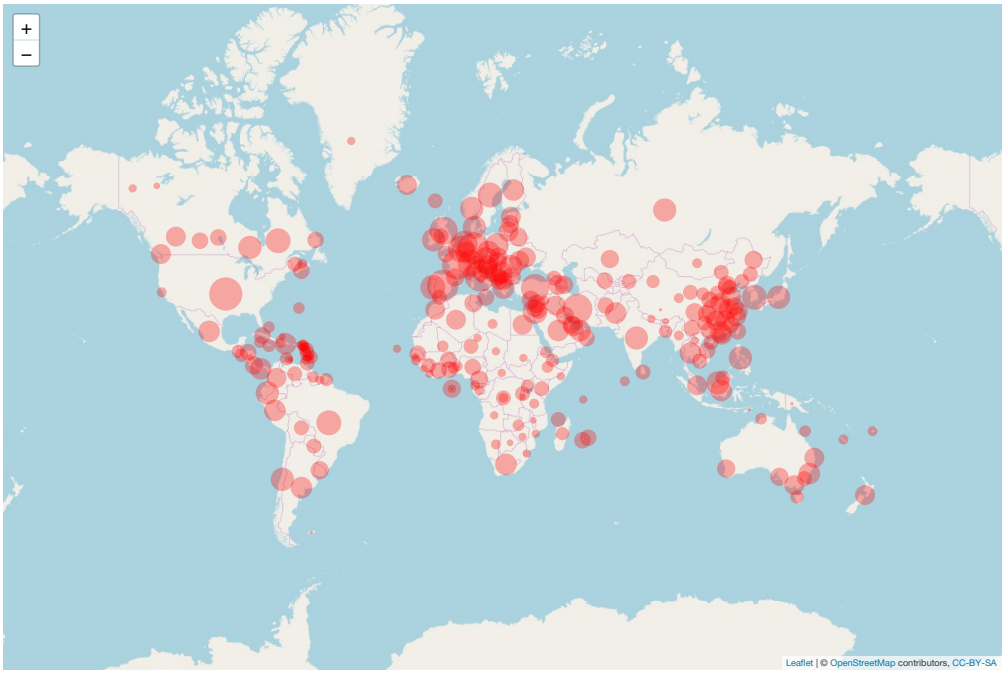


Figure 1: World Map

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)
```

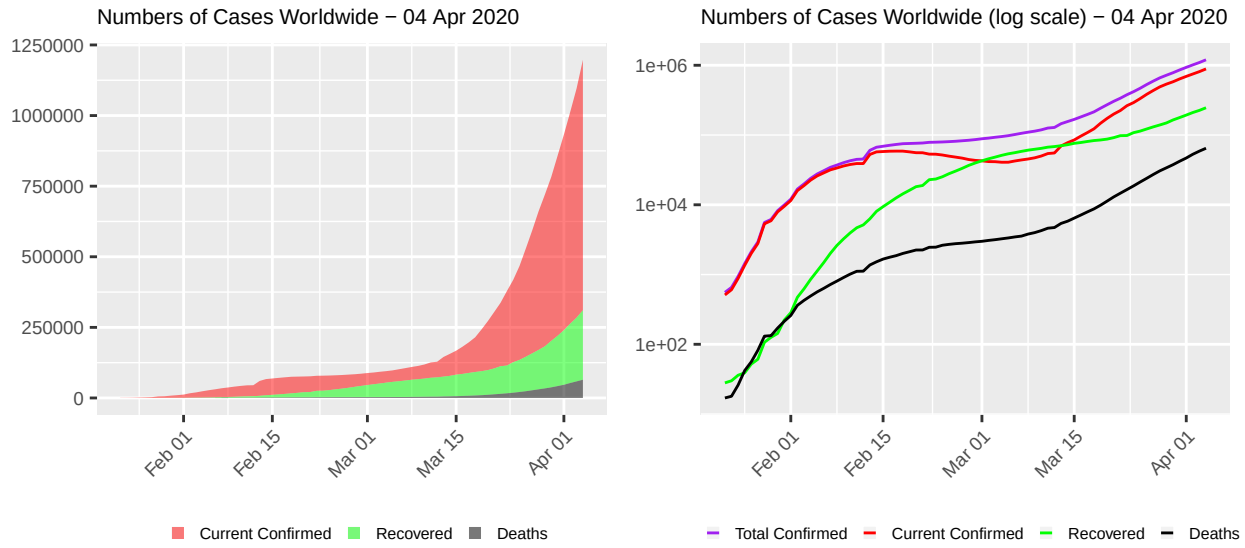


Figure 2: COVID-19 Cases Worldwide

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)
```

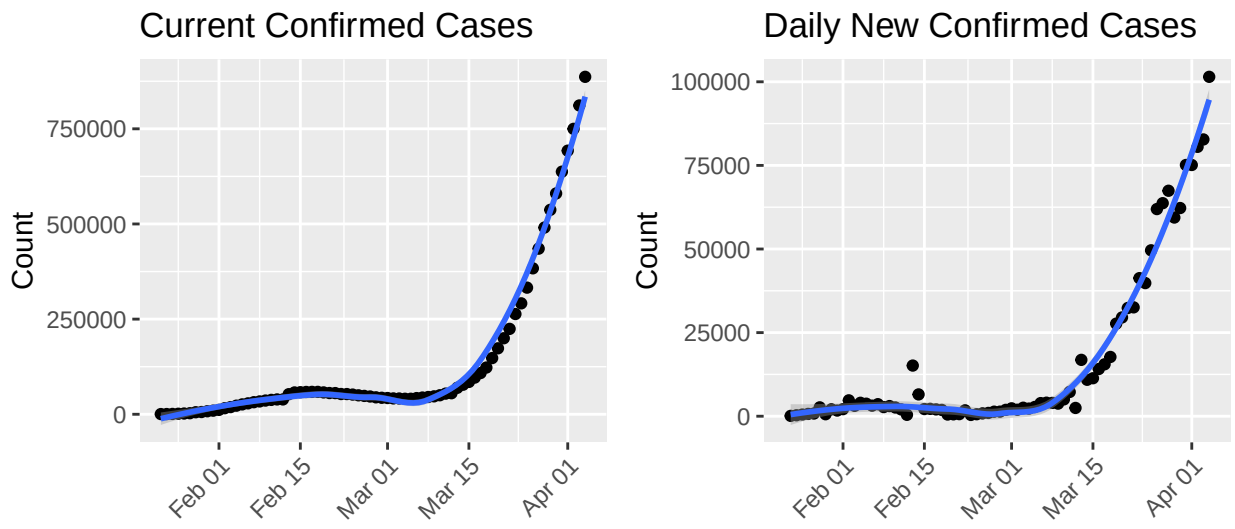


Figure 3: Current Confirmed Cases

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)
```

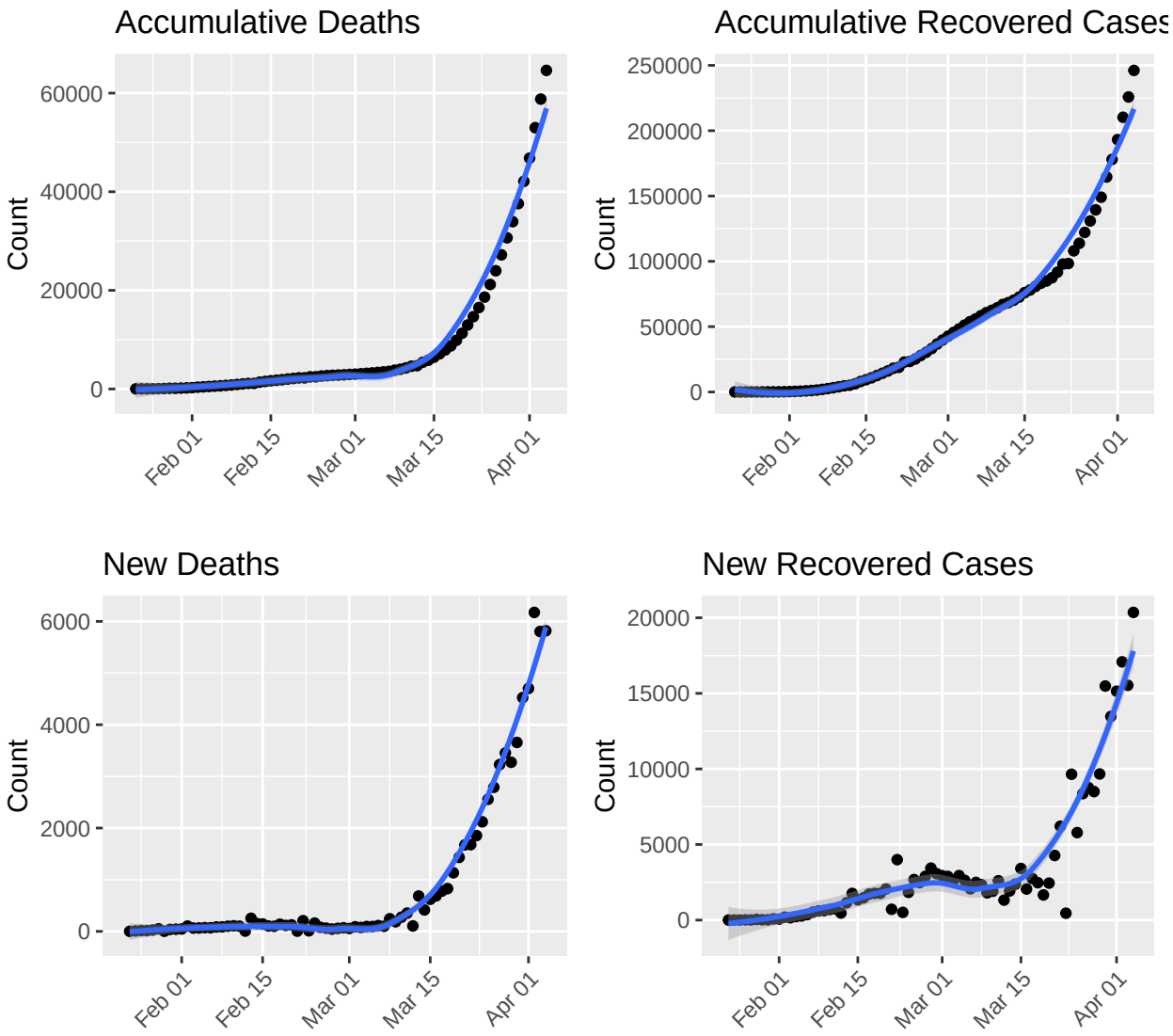


Figure 4: Deaths and Recovered Cases

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 04 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 04 Apr 2020 UTC, it will be between 5.4% and 20.8%.

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)
```

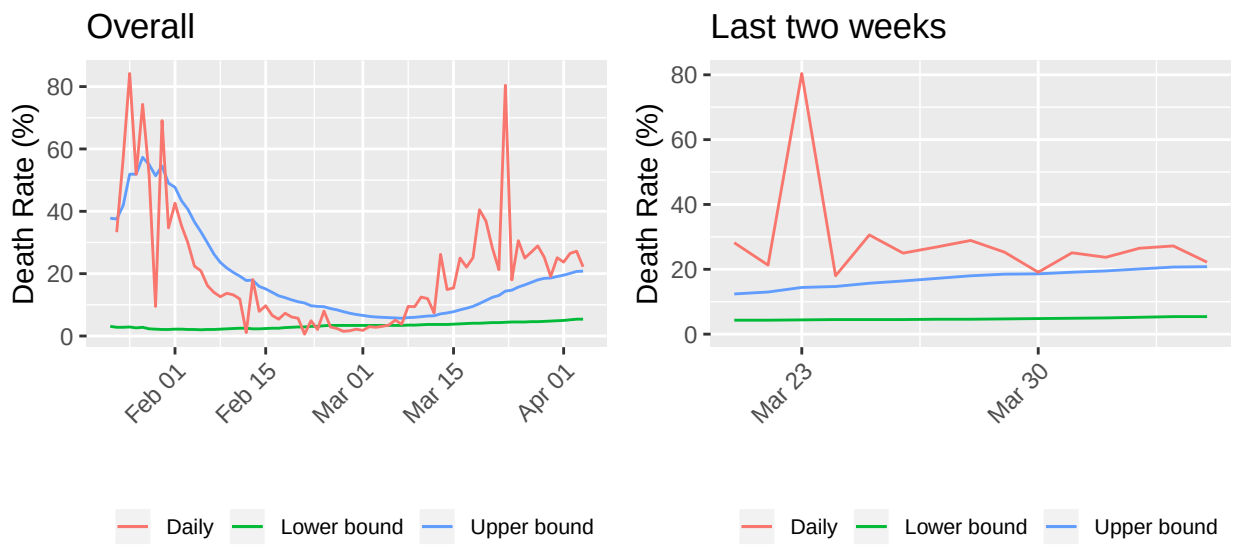


Figure 5: Death Rate

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

```
## 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',
```

Table 3: Cases in Top 20 Countries - 04 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	1,197,405	64,606	5.4%	101,488	5,819	886,647
2	US	308,850	8,407	2.7%	33,264	1,320	285,791
3	Spain	126,168	11,947	9.5%	6,969	749	80,002
4	Italy	124,632	15,362	12.3%	4,805	681	88,274
5	Germany	96,092	1,444	1.5%	4,933	169	68,248
6	France	90,848	7,574	8.3%	25,646	1,054	67,702
7	China	82,543	3,330	4.0%	32	4	2,267
8	Iran	55,743	3,452	6.2%	2,560	158	32,555
9	United Kingdom	42,477	4,320	10.2%	3,788	709	37,942
10	Turkey	23,934	501	2.1%	3,013	76	22,647
11	Switzerland	20,505	666	3.2%	899	75	13,424
12	Belgium	18,431	1,283	7.0%	1,661	140	13,901
13	Netherlands	16,727	1,656	9.9%	906	166	14,809
14	Canada	12,978	218	1.7%	541	39	10,183
15	Austria	11,781	186	1.6%	257	18	9,088
16	Portugal	10,524	266	2.5%	638	20	10,183
17	Brazil	10,360	445	4.3%	1,304	86	9,788
18	Korea, South	10,156	177	1.7%	94	3	3,654
19	Israel	7,851	44	0.6%	423	4	7,380
20	Sweden	6,443	373	5.8%	312	15	5,865
21	Australia	5,550	30	0.5%	220	2	4,819
22	Norway	5,550	62	1.1%	180	3	5,456
23	Others	109,262	2,863	2.6%	9,043	329	92,669

```
      plot.title=element_text(size=11),
      axis.text=element_text(size=7),
      axis.text.x=element_text(angle=45, hjust=1)) +
facet_wrap(~type, ncol=1, scales='free_y')
```

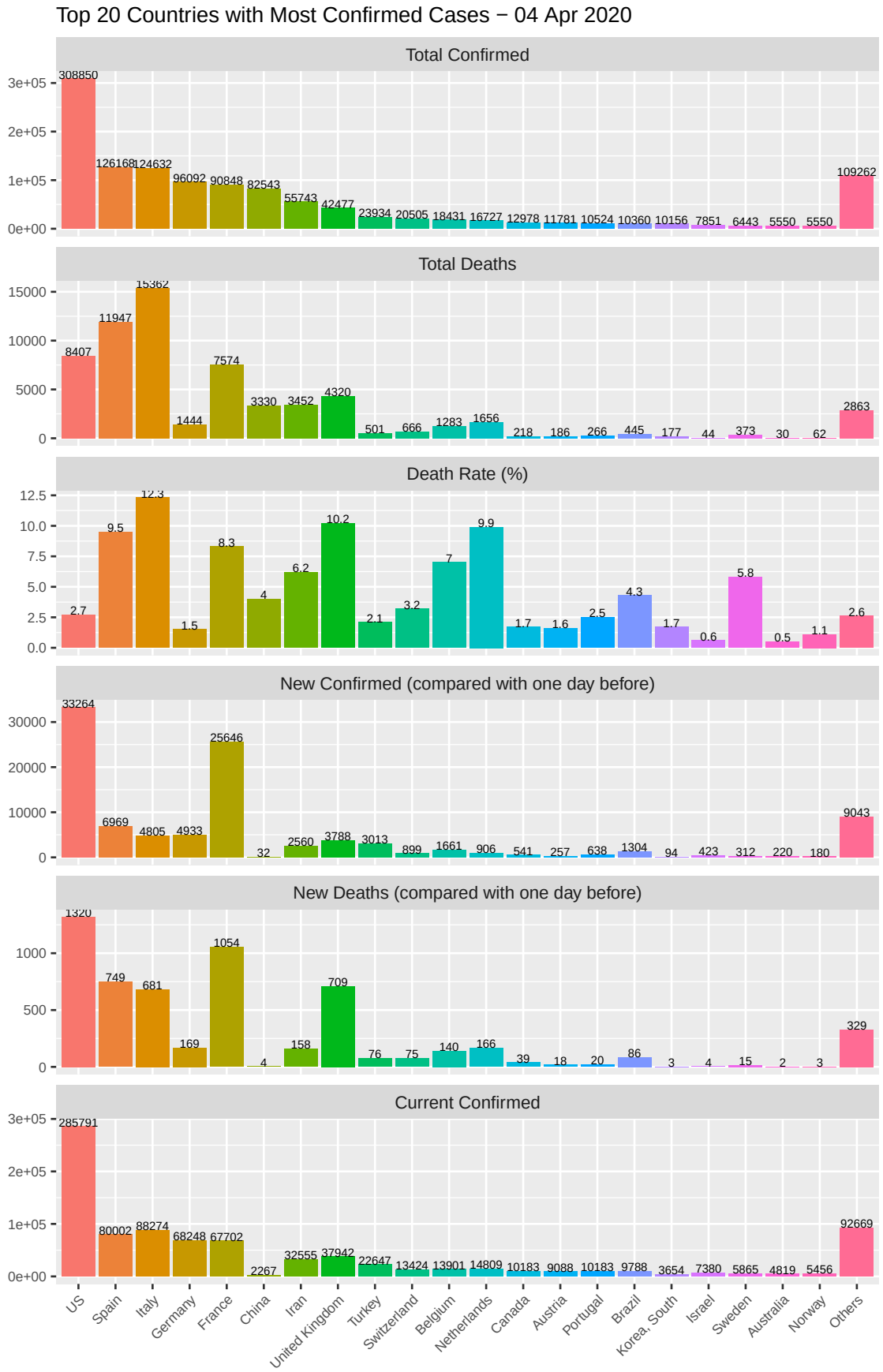
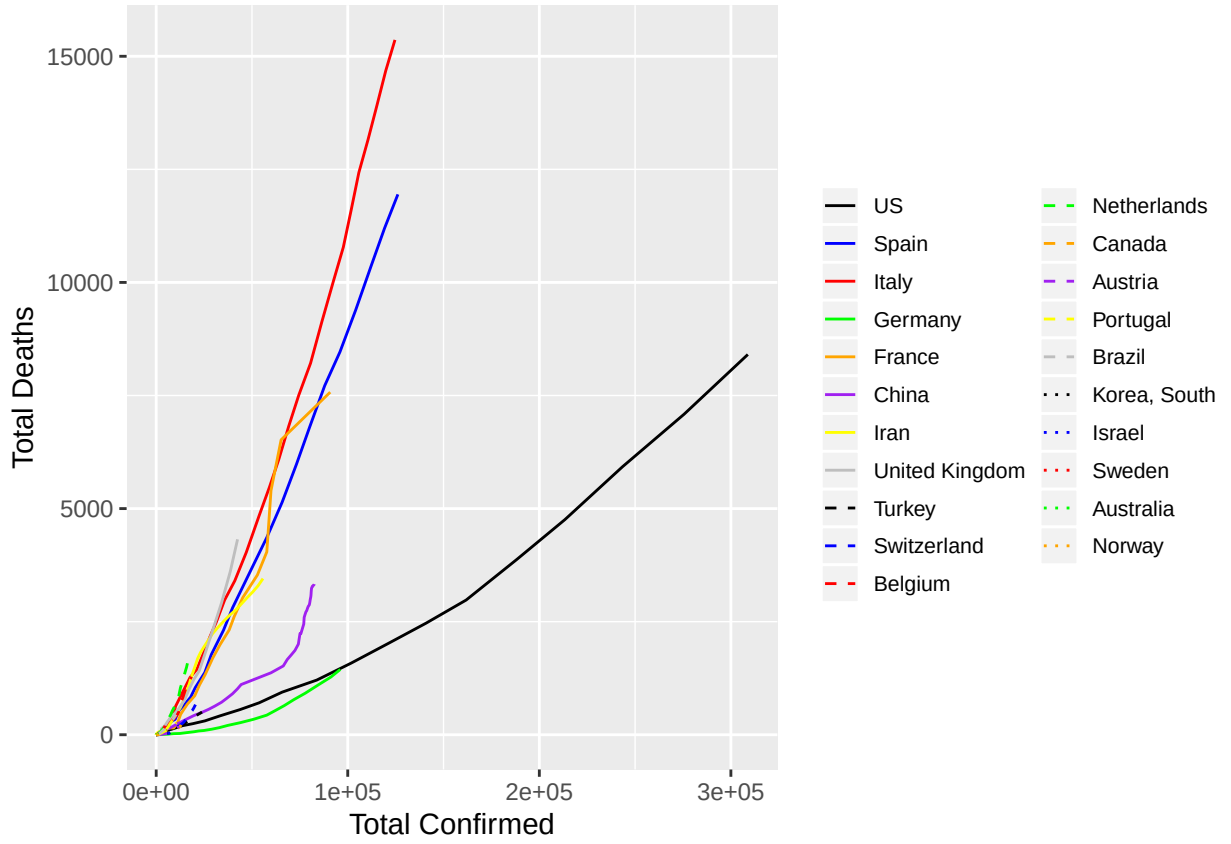


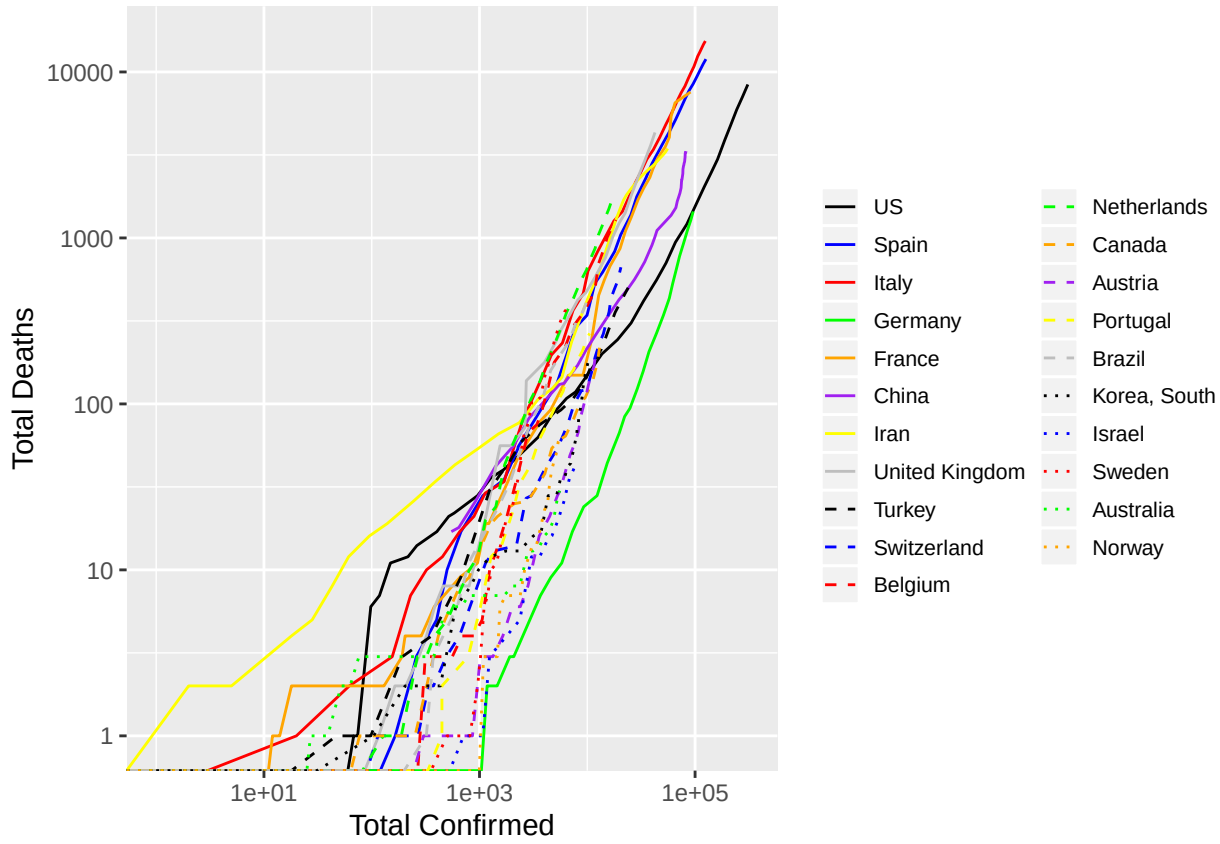
Figure 6: Top 20 Countries with Most Confirmed Cases

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 04 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)
```

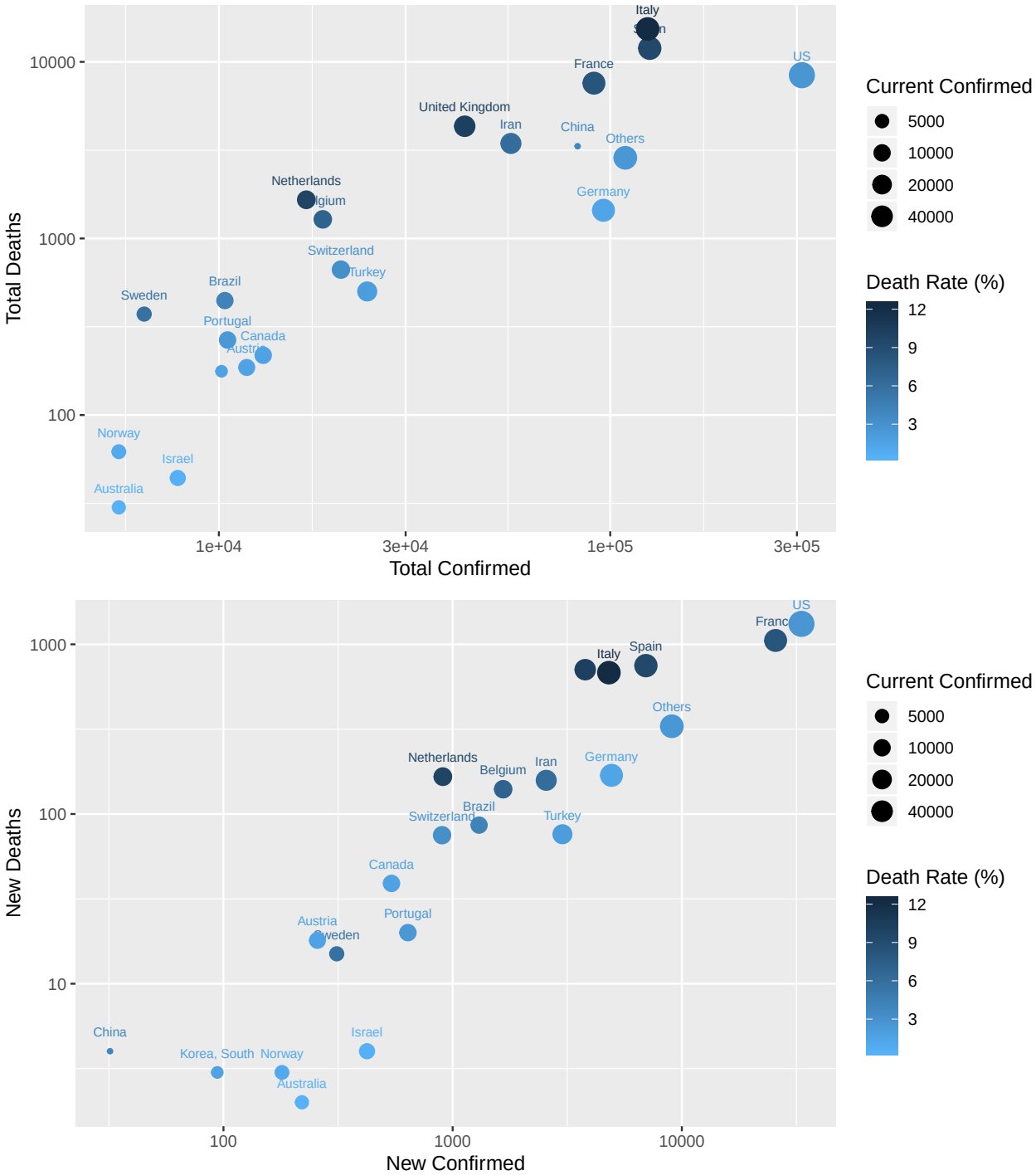


Figure 7: Top 20 Countries

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)
```

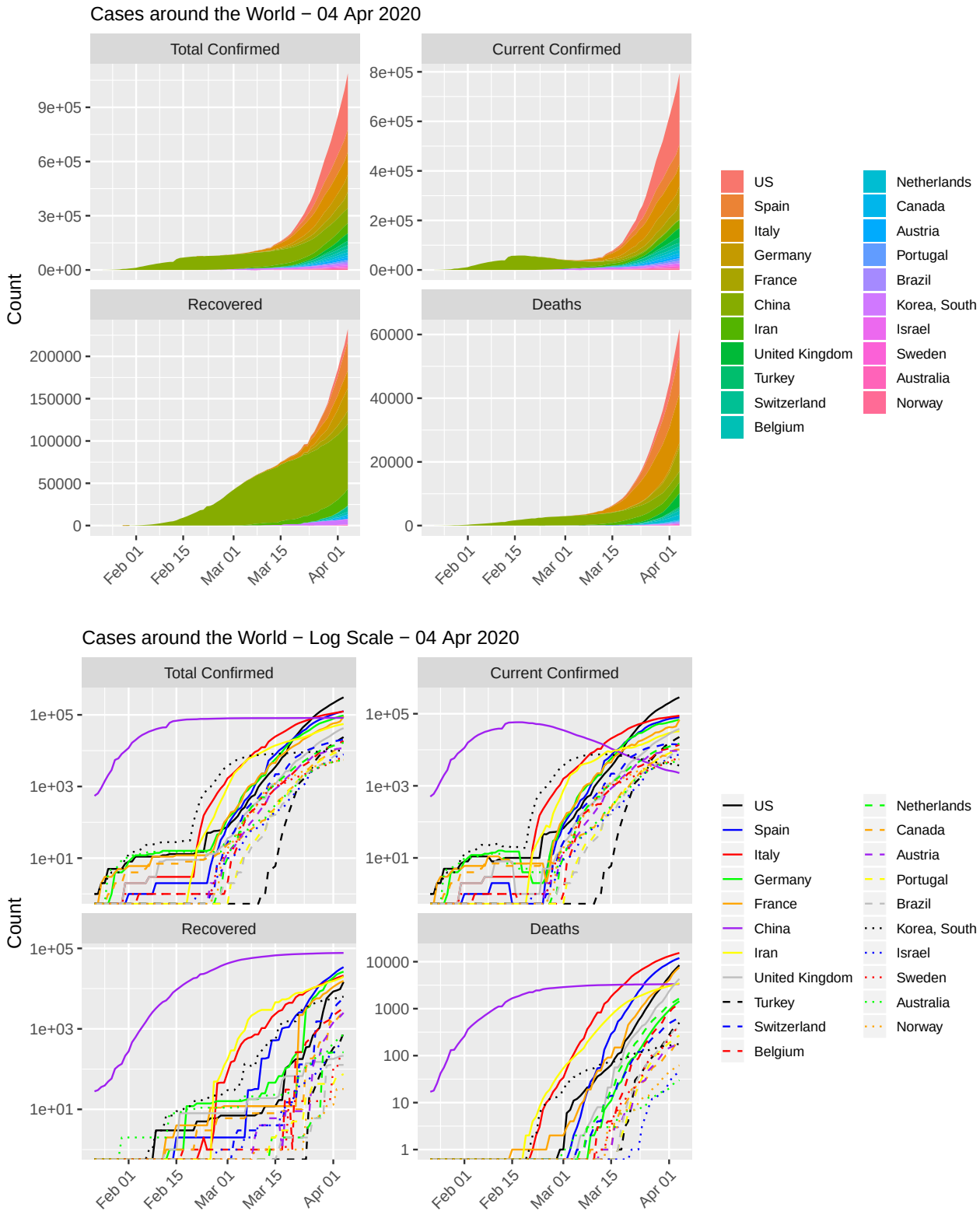


Figure 8: Cases around the World

```
## 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))
```

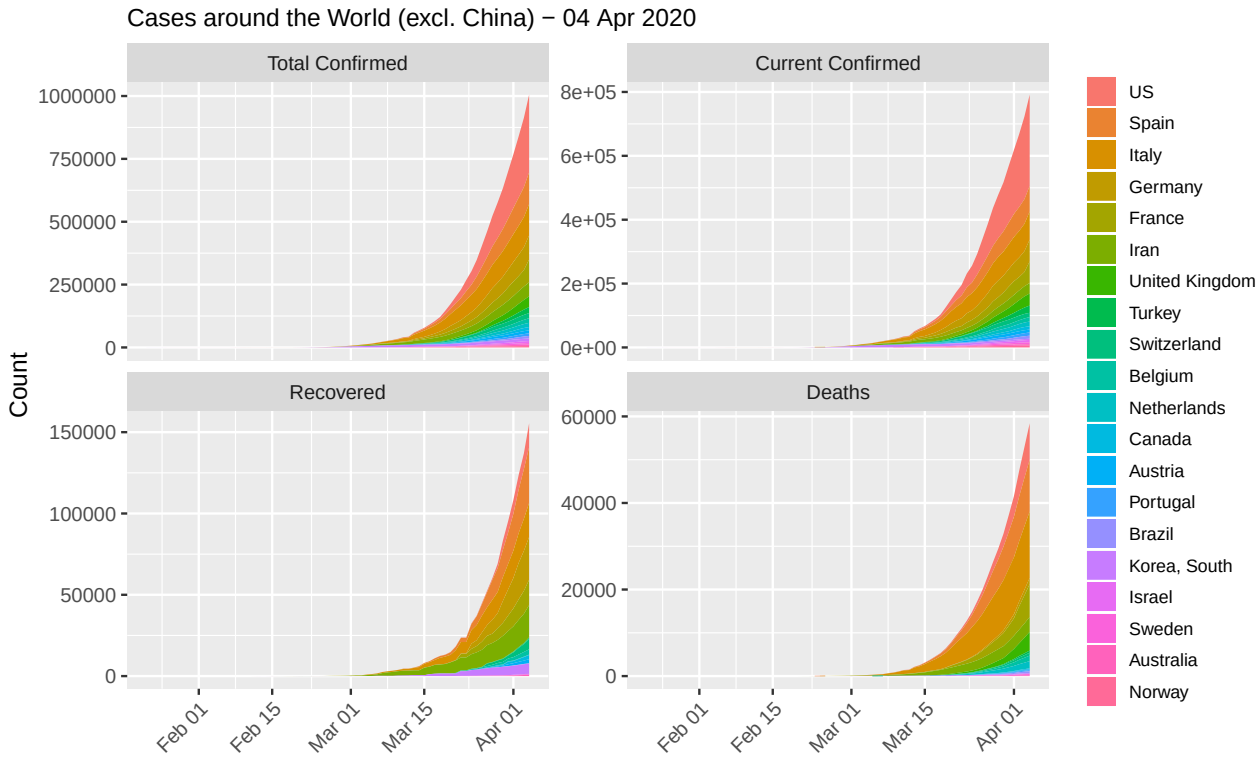


Figure 9: Cases around the World (excl. China)

```
## 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')
```

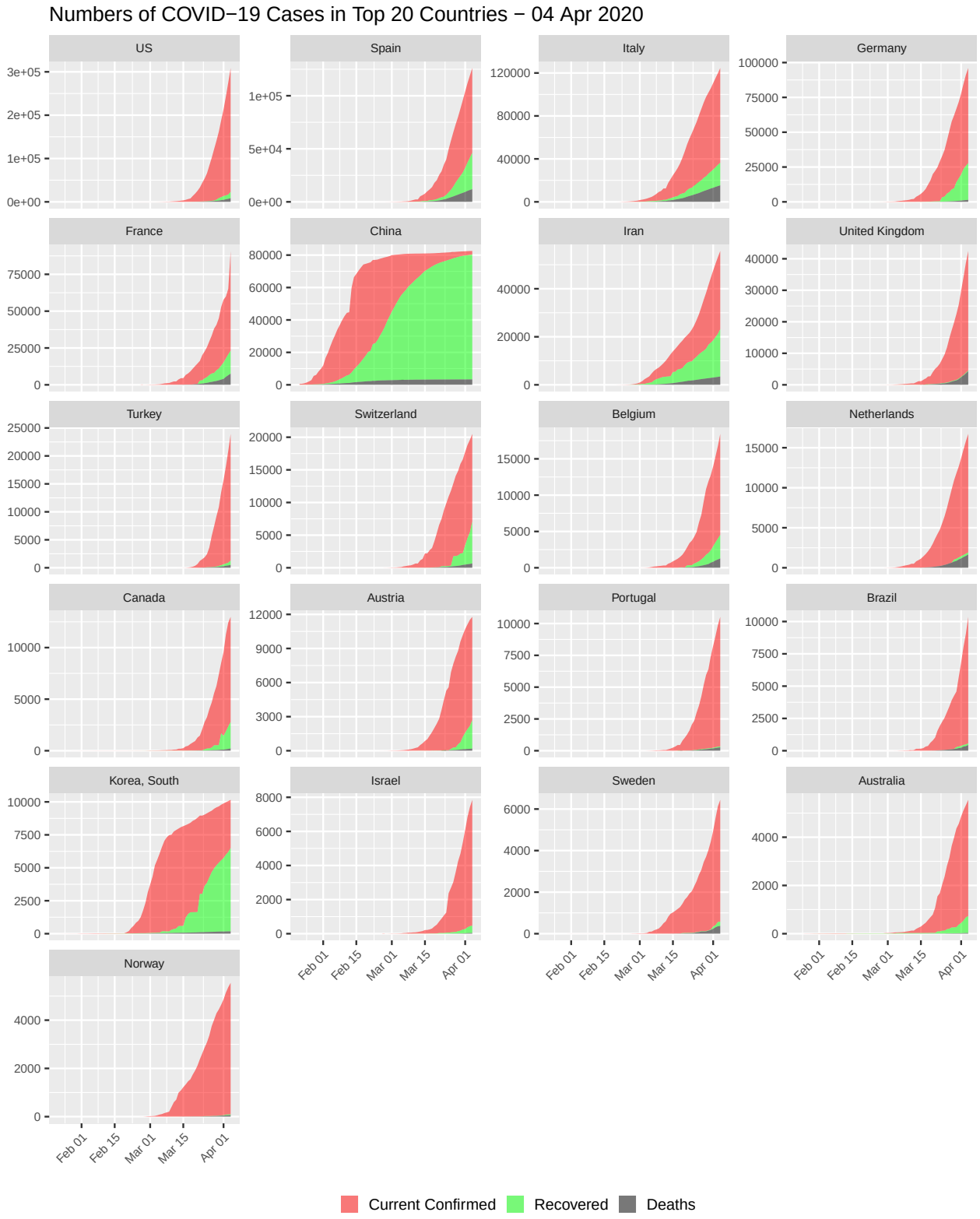


Figure 10: COVID-19 Cases in Top 20 Countries. Ordered descendingly by number of confirmed cases.

```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')
```

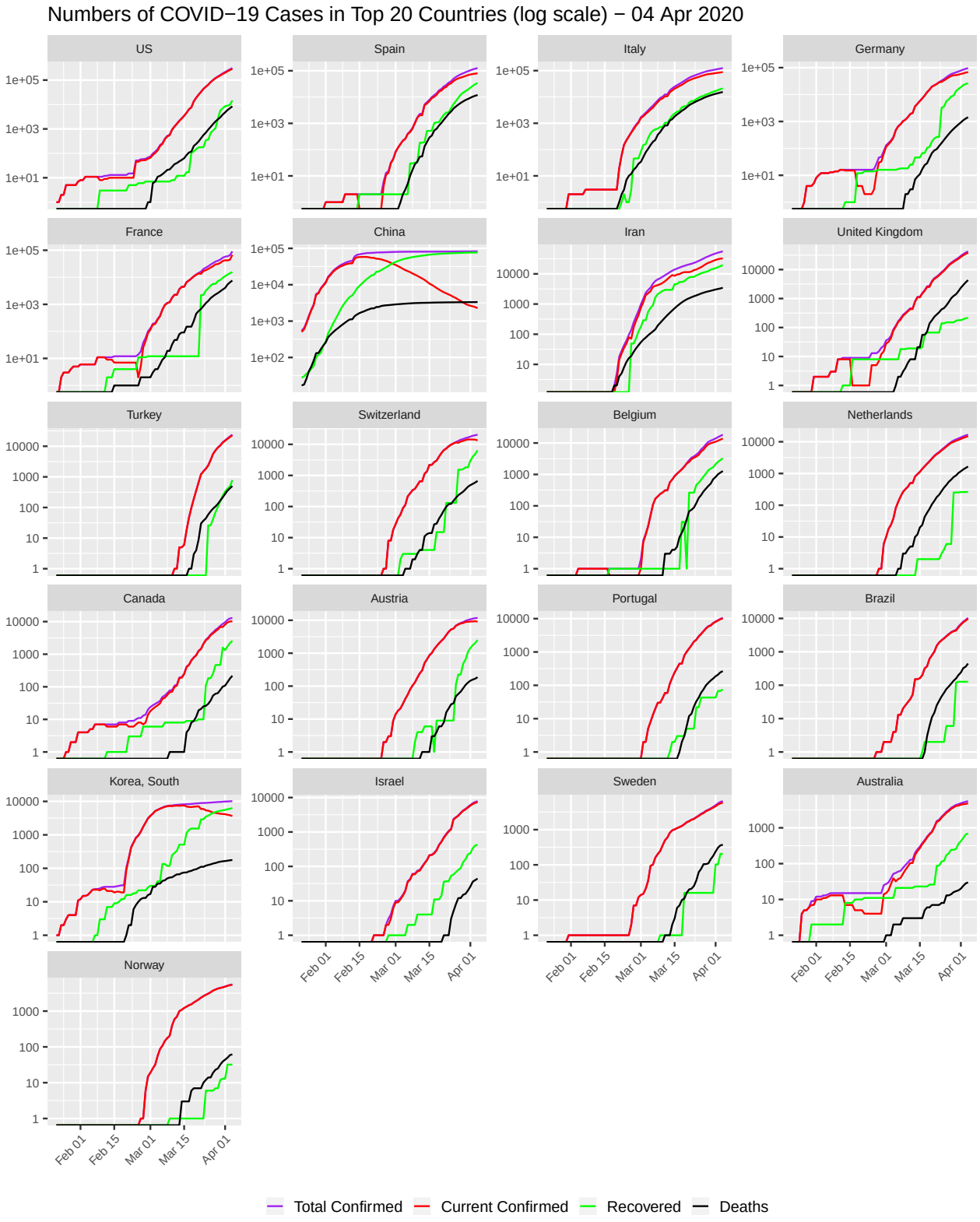


Figure 11: COVID-19 Cases Top 20 Countries (log scale). Ordered descendingly by number of confirmed cases.

```
## 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)
```

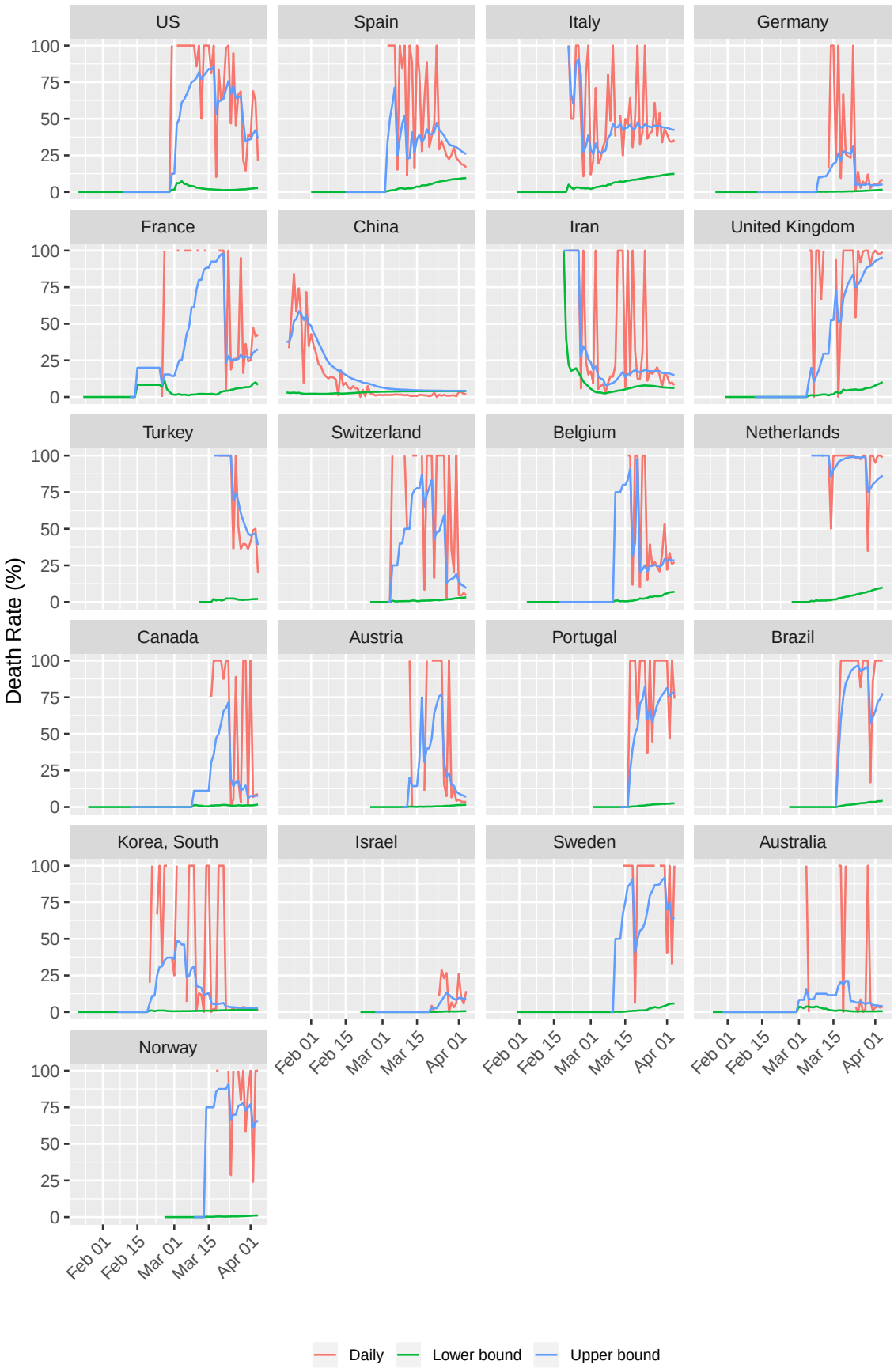


Figure 12: Death Rates

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 - 04 Apr 2020

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	Italy	124,632	4,805	88,274	20,996	15,362	681	12.3%
2	Algeria	1,251	80	1,031	90	130	25	10.4%
3	United Kingdom	42,477	3,788	37,942	215	4,320	709	10.2%
4	Netherlands	16,727	906	14,809	262	1,656	166	9.9%
5	Spain	126,168	6,969	80,002	34,219	11,947	749	9.5%
6	Indonesia	2,092	106	1,751	150	191	10	9.1%
7	France	90,848	25,646	67,702	15,572	7,574	1,054	8.3%
8	Belgium	18,431	1,661	13,901	3,247	1,283	140	7.0%
9	Egypt	1,070	85	758	241	71	5	6.6%
10	Iran	55,743	2,560	32,555	19,736	3,452	158	6.2%
11	Sweden	6,443	312	5,865	205	373	15	5.8%
12	Ecuador	3,465	97	3,193	100	172	27	5.0%
13	Philippines	3,094	76	2,893	57	144	8	4.7%
14	Dominican Republic	1,488	0	1,404	16	68	0	4.6%
15	Brazil	10,360	1,304	9,788	127	445	86	4.3%
16	Peru	1,746	151	759	914	73	12	4.2%
17	Greece	1,673	60	1,527	78	68	5	4.1%
18	China	82,543	32	2,267	76,946	3,330	4	4.0%
19	Romania	3,613	430	3,138	329	146	13	4.0%
20	Denmark	4,269	323	2,729	1,379	161	22	3.8%

6 Conclusions

As of 04 Apr 2020, there are 181 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 5.4% and 20.8%, 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
2020-04-04	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	82,760	5,804	15,533	5.4	20.7	27.2
2020-04-02	1,013,157	52,983	210,263	749,911	80,552	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	75,122	4,525	13,468	4.9	19.1	25.1
2020-03-30	782,365	37,582	164,566	580,217	62,248	3,657	15,484	4.8	18.6	19.1
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=','), align=c('l', rep('r', 7))) %>%
  kable_styling(font_size=6, latex_options=c('striped', 'hold_position', 'repeat_header'))
```

Table 6: Cases by Country (04 Apr 2020)

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
1	World	1,197,405	101,488	886,647	246,152	64,606	5,819	**5.4**
2	US	308,850	33,264	285,791	14,652	8,407	1,320	2.7
3	Spain	126,168	6,969	80,002	34,219	11,947	749	**9.5**
4	Italy	124,632	4,805	88,274	20,996	15,362	681	**12.3**
5	Germany	96,092	4,933	68,248	26,400	1,444	169	1.5
6	France	90,848	25,646	67,702	15,572	7,574	1,054	**8.3**
7	China	82,543	32	2,267	76,946	3,330	4	4
8	Iran	55,743	2,560	32,555	19,736	3,452	158	**6.2**
9	United Kingdom	42,477	3,788	37,942	215	4,320	709	**10.2**
10	Turkey	23,934	3,013	22,647	786	501	76	2.1
11	Switzerland	20,505	899	13,424	6,415	666	75	3.2
12	Belgium	18,431	1,661	13,901	3,247	1,283	140	**7**
13	Netherlands	16,727	906	14,809	262	1,656	166	**9.9**
14	Canada	12,978	541	10,183	2,577	218	39	1.7
15	Austria	11,781	257	9,088	2,507	186	18	1.6
16	Portugal	10,524	638	10,183	75	266	20	2.5
17	Brazil	10,360	1,304	9,788	127	445	86	4.3
18	Korea, South	10,156	94	3,654	6,325	177	3	1.7
19	Israel	7,851	423	7,380	427	44	4	0.6
20	Sweden	6,443	312	5,865	205	373	15	**5.8**
21	Australia	5,550	220	4,819	701	30	2	0.5
22	Norway	5,550	180	5,456	32	62	3	1.1
23	Russia	4,731	582	4,355	333	43	9	0.9
24	Ireland	4,604	331	4,442	25	137	17	3
25	Czechia	4,472	381	4,335	78	59	6	1.3
26	Denmark	4,269	323	2,729	1,379	161	22	3.8
27	Chile	4,161	424	3,606	528	27	5	0.6
28	Poland	3,627	244	3,432	116	79	8	2.2
29	Romania	3,613	430	3,138	329	146	13	4
30	Malaysia	3,483	150	2,511	915	57	4	1.6
31	Ecuador	3,465	97	3,193	100	172	27	**5**
32	Japan	3,139	522	2,548	514	77	14	2.5
33	Philippines	3,094	76	2,893	57	144	8	4.7
34	India	3,082	515	2,767	229	86	14	2.8
35	Pakistan	2,818	132	2,646	131	41	1	1.5
36	Luxembourg	2,729	117	2,198	500	31	0	1.1
37	Saudi Arabia	2,179	140	1,730	420	29	4	1.3
38	Indonesia	2,092	106	1,751	150	191	10	**9.1**
39	Thailand	2,067	89	1,373	674	20	1	1
40	Finland	1,882	267	1,557	300	25	5	1.3
41	Peru	1,746	151	759	914	73	12	4.2
42	Mexico	1,688	178	995	633	60	10	3.6
43	Greece	1,673	60	1,527	78	68	5	4.1
44	Panama	1,673	198	1,619	13	41	4	2.5
45	Serbia	1,624	148	1,580	0	44	5	2.7
46	South Africa	1,585	80	1,481	95	9	0	0.6
47	United Arab Emirates	1,505	241	1,370	125	10	1	0.7
48	Dominican Republic	1,488	0	1,404	16	68	0	4.6
49	Argentina	1,451	186	1,129	279	43	4	3
50	Iceland	1,417	53	1,017	396	4	0	0.3
51	Colombia	1,406	139	1,289	85	32	7	2.3
52	Qatar	1,325	250	1,213	109	3	0	0.2

Table 6: Cases by Country (04 Apr 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
53	Algeria	1,251	80	1,031	90	130	25	**10.4**
54	Ukraine	1,225	153	1,168	25	32	5	2.6
55	Singapore	1,189	75	886	297	6	1	0.5
56	Croatia	1,126	47	995	119	12	4	1.1
57	Egypt	1,070	85	758	241	71	5	**6.6**
58	Estonia	1,039	78	967	59	13	1	1.3
59	Slovenia	977	43	876	79	22	2	2.3
60	New Zealand	950	82	822	127	1	0	0.1
61	Morocco	919	128	794	66	59	11	6.4
62	Iraq	878	58	563	259	56	2	6.4
63	Lithuania	771	75	753	7	11	2	1.4
64	Armenia	770	34	720	43	7	0	0.9
65	Moldova	752	161	711	29	12	4	1.6
66	Diamond Princess	712	0	82	619	11	0	1.5
67	Bahrain	688	16	261	423	4	0	0.6
68	Hungary	678	55	588	58	32	6	4.7
69	Bosnia and Herzegovina	624	45	573	30	21	4	3.4
70	Cameroon	555	46	529	17	9	1	1.6
71	Tunisia	553	58	530	5	18	0	3.3
72	Kazakhstan	531	67	490	36	5	0	0.9
73	Azerbaijan	521	78	484	32	5	0	1
74	Lebanon	520	12	449	54	17	0	3.3
75	Latvia	509	16	507	1	1	0	0.2
76	Bulgaria	503	18	452	34	17	3	3.4
77	North Macedonia	483	53	446	20	17	5	3.5
78	Kuwait	479	62	385	93	1	1	0.2
79	Slovakia	471	21	460	10	1	0	0.2
80	Andorra	466	27	428	21	17	1	3.6
81	Belarus	440	89	382	53	5	1	1.1
82	Costa Rica	435	19	420	13	2	0	0.5
83	Cyprus	426	30	382	33	11	0	2.6
84	Uruguay	400	31	302	93	5	1	1.2
85	Taiwan*	355	7	300	50	5	0	1.4
86	Albania	333	29	214	99	20	3	6
87	Jordan	323	13	244	74	5	0	1.5
88	Burkina Faso	318	16	236	66	16	0	5
89	Afghanistan	299	18	282	10	7	1	2.3
90	Cuba	288	19	267	15	6	0	2.1
91	Oman	277	25	214	61	2	1	0.7
92	Uzbekistan	266	39	239	25	2	0	0.8
93	Honduras	264	42	246	3	15	0	5.7
94	San Marino	259	14	200	27	32	2	12.4
95	Cote d'Ivoire	245	27	219	25	1	0	0.4
96	Vietnam	240	3	150	90	0	0	0
97	Senegal	219	12	145	72	2	1	0.9
98	West Bank and Gaza	217	23	195	21	1	0	0.5
99	Nigeria	214	4	185	25	4	0	1.9
100	Malta	213	11	211	2	0	0	0
101	Ghana	205	0	169	31	5	0	2.4
102	Montenegro	201	27	198	1	2	0	1
103	Mauritius	196	10	182	7	7	0	3.6
104	Sri Lanka	166	7	134	27	5	1	3
105	Georgia	162	7	125	36	1	1	0.6
106	Venezuela	155	2	96	52	7	0	4.5
107	Congo (Kinshasa)	154	20	133	3	18	5	11.7
108	Kyrgyzstan	144	14	134	9	1	0	0.7
109	Niger	144	24	136	0	8	3	5.6
110	Bolivia	139	7	128	1	10	1	7.2
111	Brunei	135	1	68	66	1	0	0.7
112	Kosovo	135	9	118	16	1	0	0.7
113	Kenya	126	4	118	4	4	0	3.2
114	Cambodia	114	0	64	50	0	0	0
115	Guinea	111	38	106	5	0	0	0
116	Trinidad and Tobago	103	5	96	1	6	0	5.8
117	Rwanda	102	13	102	0	0	0	0
118	Paraguay	96	4	81	12	3	0	3.1
119	Liechtenstein	77	2	76	0	1	1	1.3
120	Bangladesh	70	9	32	30	8	2	11.4

Table 6: Cases by Country (04 Apr 2020) *(continued)*

	country	confirmed	new.confirmed	current.confirmed	recovered	deaths	new.deaths	death.rate
121	Madagascar	70	0	70	0	0	0	0
122	Monaco	66	2	62	3	1	0	1.5
123	Guatemala	61	11	44	15	2	1	3.3
124	El Salvador	56	10	51	2	3	1	5.4
125	Jamaica	53	6	43	7	3	0	5.7
126	Barbados	52	1	52	0	0	0	0
127	Djibouti	50	1	42	8	0	0	0
128	Uganda	48	0	48	0	0	0	0
129	Mali	41	2	37	1	3	0	7.3
130	Togo	41	1	21	17	3	0	7.3
131	Zambia	39	0	36	2	1	0	2.6
132	Ethiopia	38	3	34	4	0	0	0
133	Eritrea	29	7	29	0	0	0	0
134	Bahamas	28	4	24	0	4	3	14.3
135	Guyana	23	0	19	0	4	0	17.4
136	Congo (Brazzaville)	22	0	18	2	2	0	9.1
137	Burma	21	1	20	0	1	0	4.8
138	Gabon	21	0	19	1	1	0	4.8
139	Haiti	20	2	19	1	0	0	0
140	Tanzania	20	0	16	3	1	0	5
141	Maldives	19	0	6	13	0	0	0
142	Guinea-Bissau	18	3	18	0	0	0	0
143	Libya	18	7	17	0	1	0	5.6
144	Benin	16	0	14	2	0	0	0
145	Equatorial Guinea	16	0	15	1	0	0	0
146	Syria	16	0	12	2	2	0	12.5
147	Antigua and Barbuda	15	0	15	0	0	0	0
148	Dominica	14	2	14	0	0	0	0
149	Mongolia	14	0	12	2	0	0	0
150	Namibia	14	0	11	3	0	0	0
151	Saint Lucia	14	1	13	1	0	0	0
152	Fiji	12	5	12	0	0	0	0
153	Grenada	12	0	12	0	0	0	0
154	Angola	10	2	6	2	2	0	20
155	Laos	10	0	10	0	0	0	0
156	Liberia	10	3	6	3	1	1	10
157	Mozambique	10	0	9	1	0	0	0
158	Seychelles	10	0	10	0	0	0	0
159	Sudan	10	0	6	2	2	0	20
160	Suriname	10	0	9	0	1	0	10
161	Chad	9	1	9	0	0	0	0
162	Eswatini	9	0	9	0	0	0	0
163	MS Zaandam	9	0	7	0	2	0	22.2
164	Nepal	9	3	8	1	0	0	0
165	Saint Kitts and Nevis	9	0	9	0	0	0	0
166	Zimbabwe	9	0	8	0	1	0	11.1
167	Central African Republic	8	0	8	0	0	0	0
168	Cabo Verde	7	1	6	0	1	0	14.3
169	Holy See	7	0	7	0	0	0	0
170	Saint Vincent and the Grenadines	7	4	6	1	0	0	0
171	Somalia	7	0	6	1	0	0	0
172	Mauritania	6	0	3	2	1	0	16.7
173	Bhutan	5	0	3	2	0	0	0
174	Nicaragua	5	0	4	0	1	0	20
175	Belize	4	0	4	0	0	0	0
176	Botswana	4	0	3	0	1	0	25
177	Gambia	4	0	1	2	1	0	25
178	Malawi	4	1	4	0	0	0	0
179	Sierra Leone	4	2	4	0	0	0	0
180	Burundi	3	0	3	0	0	0	0
181	Papua New Guinea	1	0	1	0	0	0	0
182	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!