

# Covid-19

Bernt Arne Ødegaard

24 November 2021

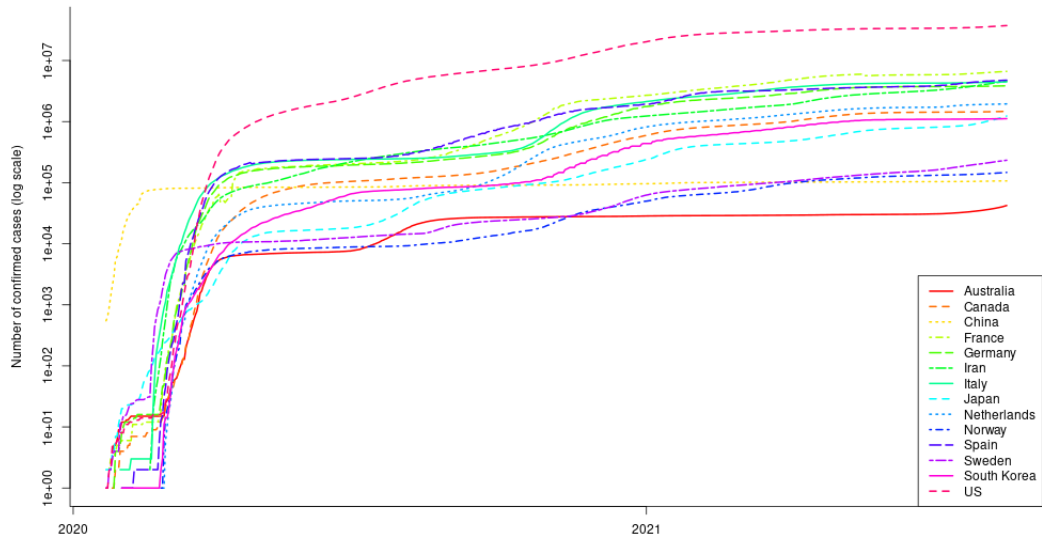
## 1 Events

Gradual spread of a “new” virus, after a while christened Covid-19.  
Figure 1 show the evolution of reported cases.

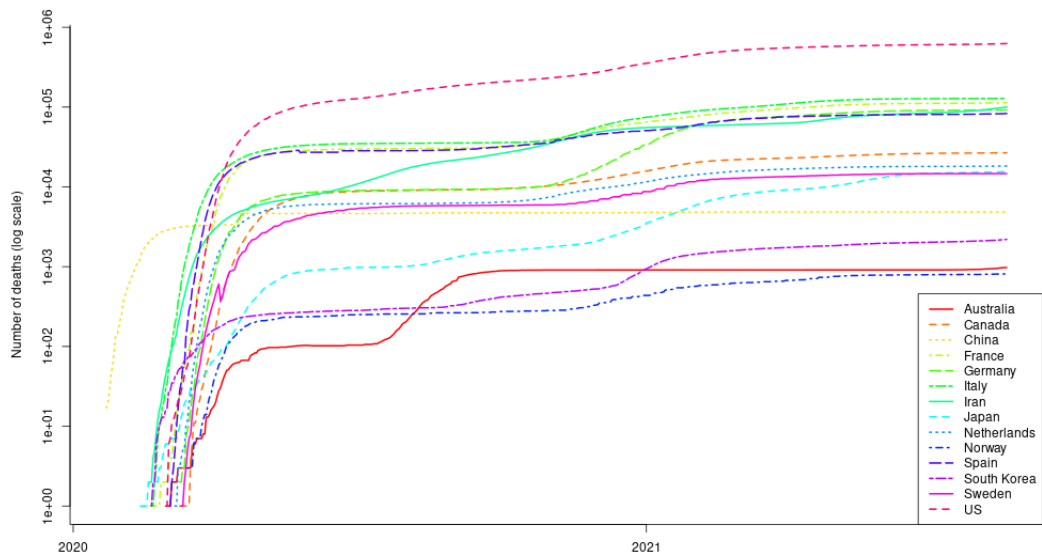
**Figure 1** The evolution of Covid-19 cases

The evolution of confirmed cases of Covid-19. The plot show the total number of confirmed cases (deaths) per country (log scale). Countries: Australia, Canada, China, France, Germany, Italy, Iran, Norway, South Korea, Spain, Sweden, USA. Data Source: John Hopkins University Covid-19 data.

Panel A: Confirmed Cases



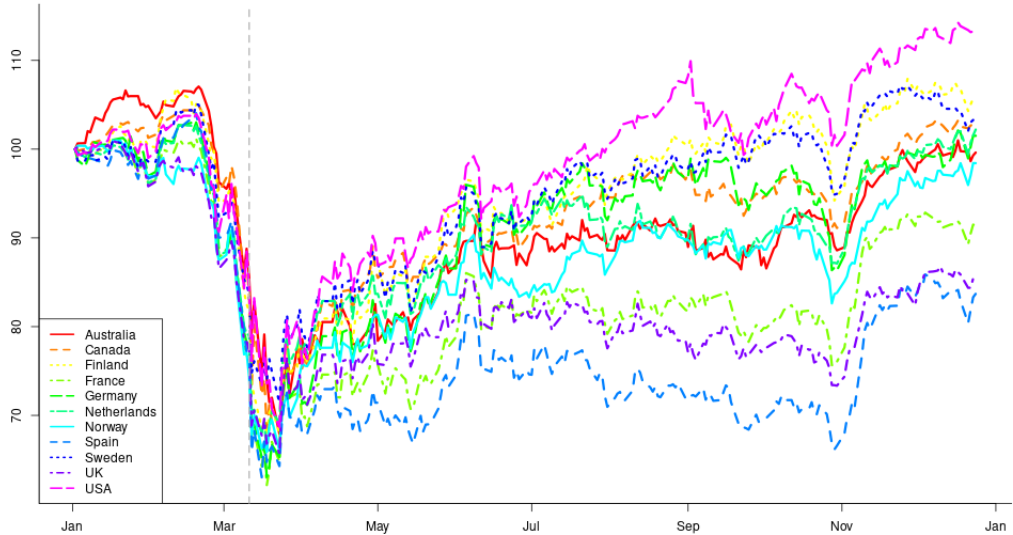
Panel B: Deaths related to Covid-19



# 2 Stock Market Reactions

**Figure 2** Comparison of stock market levels

Cross-country comparison of stock market evolution. Indices set to 100 at beginning of 2020.



### 3 R usage, read data

```
library(xts)
```

```
read_all_jhu_confirmed <- function(){  
  indir <- "/home/bernt/data/2021/jhu_covid_data/"  
  filename <- paste0(indir, "time_series_covid19_confirmed_global_narrow.csv")  
  data <- read.csv(filename, header=TRUE, sep=",", comment.char="#")  
  names(data) <- c("Province", "Country", "Latitude", "Longitude",  
                  "Date", "Confirmed_Cases", "Country_Code",  
                  "Region_Code", "Sub_Region_Code", "Intr_Region_Code")  
  return (data)  
}
```

```
sum_cut_na <- function(x) { sum(na.omit(x))}
```

```
time_series_confirmed_cases_for_country <- function(country){  
  data <- read_all_jhu_confirmed()  
  data <- data[(data$Country==country),]  
  data$Date <- as.Date(data$Date, format="%Y-%m-%d")  
  # now need to sum across provinces, e.g. for austr  
  confirmed_cases <- data.frame(data$Date,  
                                data$Province,  
                                data$Confirmed_Cases)  
  names(confirmed_cases) <- c("Date", "Province", "Confirmed_Cases")  
  ts <- reshape(confirmed_cases, idvar="Date", timevar="Province", direction="wide")  
  ts <- xts(ts[,2:ncol(ts)], order.by=ts$Date)  
  country_confirmed_cases <- xts(apply(ts, MARGIN=1, sum_cut_na), order.by=index(ts))  
  return (country_confirmed_cases)  
};
```

```
time_series_log_confirmed_cases_for_country <- function(country){  
  ts_confirmed_cases <- time_series_confirmed_cases_for_country(country);  
  log_ts_confirmed_cases <- log(ts_confirmed_cases[ts_confirmed_cases>0])  
  return (log_ts_confirmed_cases)  
};
```

## 4 R usage, plot covid data

```
Sys.setlocale(category = "LC_ALL", locale = "C")

outdir <- "../.. / results/2021_08_plot_jhu_covid_data/"
progdir <- "../2021_08_read_jhu_covid_data/"
filename <- paste0(progdir, "read_jhu_data_confirmed.R")
source (filename)

last_date <- as.Date("2021-08-20")

ts_confirmed_australia <- time_series_confirmed_cases_for_country ("Australia")
ts_confirmed_canada <- time_series_confirmed_cases_for_country ("Canada")
ts_confirmed_denmark <- time_series_confirmed_cases_for_country ("Denmark")
ts_confirmed_finland <- time_series_confirmed_cases_for_country ("Finland")
ts_confirmed_france <- time_series_confirmed_cases_for_country ("France")
ts_confirmed_germany <- time_series_confirmed_cases_for_country ("Germany")
ts_confirmed_iceland <- time_series_confirmed_cases_for_country ("Iceland")
ts_confirmed_italy <- time_series_confirmed_cases_for_country ("Italy")
ts_confirmed_netherlands <- time_series_confirmed_cases_for_country ("Netherlands")
ts_confirmed_norway <- time_series_confirmed_cases_for_country ("Norway")
ts_confirmed_spain <- time_series_confirmed_cases_for_country ("Spain")
ts_confirmed_sweden <- time_series_confirmed_cases_for_country ("Sweden")
ts_confirmed_us <- time_series_confirmed_cases_for_country ("US")

ts_confirmed_china <- time_series_confirmed_cases_for_country ("China")
ts_confirmed_japan <- time_series_confirmed_cases_for_country ("Japan")
ts_confirmed_iran <- time_series_confirmed_cases_for_country ("Iran")
ts_confirmed_south_korea <- time_series_confirmed_cases_for_country ("Korea, South")

to_plot <- merge(ts_confirmed_australia ,
                 ts_confirmed_canada ,
                 ts_confirmed_china ,
                 ts_confirmed_france ,
                 ts_confirmed_germany ,
                 ts_confirmed_iran ,
                 ts_confirmed_italy ,
                 ts_confirmed_japan ,
                 ts_confirmed_netherlands ,
                 ts_confirmed_norway ,
                 ts_confirmed_spain ,
                 ts_confirmed_south_korea ,
                 ts_confirmed_sweden ,
                 ts_confirmed_us ,
                 all=TRUE)
to_plot [(to_plot <1)] <- NA

countries=c("Australia" ,
            "Canada" ,
            "China" ,
            "France" ,
```

```

    "Germany",
    "Iran",
    "Italy",
    "Japan",
    "Netherlands",
    "Norway",
    "Spain",
    "Sweden",
    "South Korea",
    "US")
to_plot <- window(to_plot, end=last_date)
print(to_plot)
colors <- rainbow(14)
ofilename <- paste0(outdir, "evolution_confirmed_cases.png")
png(ofilename, width=1000, height=600)
plot.zoo(as.zoo(to_plot),
         log="y",
         screens=c(1),
         lwd=c(2),
         lty=1:20,
         main="",
         ylab="Number of confirmed cases (log scale)",
         xlab="",
         bty="n",
         col=colors)
legend("bottomright",
      legend=countries,
      col=colors,
      lty=1:20,
      lwd=c(2),
      )
box(bty="L")
dev.off()

```

## 5 R usage, read data

```
Sys.setlocale(category = "LC_ALL", locale = "C")
# note that there are time zone issues, canada/aus
# force everything to be a straight date,

library(quantmod)
outdir <- "../.. / results / 2020_07_stock_market_evolution /"

getSymbols("^OSEAX", from="2020-01-01", source="yahoo")
oseax_daily_prices = na.omit(OSEAX$OSEAX.Adjusted)
index(oseax_daily_prices) <- as.Date(index(oseax_daily_prices))
names(oseax_daily_prices) <- "Norway_OSEAX"

getSymbols("^OMX", from="2020-01-01", source="yahoo")
summary(OMX)
omx_daily_prices <- na.omit(OMX$OMX.Adjusted)
index(omx_daily_prices) <- as.Date(index(omx_daily_prices))
names(omx_daily_prices) <- "Sweden_OMX"

getSymbols("^OMXH25", from = "2020-01-01", source = "yahoo")
finland_daily_prices <- na.omit(OMXH25$OMXH25.Adjusted)
index(finland_daily_prices) <- as.Date(index(finland_daily_prices))
names(finland_daily_prices) <- "Finland_OMXH25"

getSymbols("^GDAXI", from = "2020-01-01", source = "yahoo")
germany_daily_prices <- na.omit(GDAXI$GDAXI.Adjusted)
index(germany_daily_prices) <- as.Date(index(germany_daily_prices))
names(germany_daily_prices) <- "Germany_DAX"

getSymbols("^FCHI", from = "2020-01-01", source = "yahoo")
france_daily_prices <- na.omit(FCHI$FCHI.Adjusted)
names(france_daily_prices) <- "France_CAC40"
index(france_daily_prices) <- as.Date(index(france_daily_prices))
summary(france_daily_prices)

getSymbols("^AXJO", from = "2020-01-01", source = "yahoo")
australia_daily_prices <- na.omit(AXJO$AXJO.Adjusted) # issues with time zones here
australia_daily_prices <- xts(coredata(australia_daily_prices),
order.by=as.Date(index(australia_daily_prices)))
names(australia_daily_prices) <- "Australia_ASX200"
index(australia_daily_prices) <- as.Date(index(australia_daily_prices))
summary(australia_daily_prices)

getSymbols("^GSPTSE", from = "2020-01-01", source = "yahoo")
canada_daily_prices <- na.omit(GSPTSE$GSPTSE.Adjusted)
names(canada_daily_prices) <- "Canada_SPTSE"
index(canada_daily_prices) <- as.Date(index(canada_daily_prices))
summary(canada_daily_prices)

getSymbols("^AEX", from = "2020-01-01", source = "yahoo")
netherlands_daily_prices <- na.omit(AEX$AEX.Adjusted)
```

```

names(netherlands_daily_prices) <- "Netherlands_AEX"
index(netherlands_daily_prices) <- as.Date(index(netherlands_daily_prices))
summary(netherlands_daily_prices)

```

```

getSymbols("^IBEX", from = "2020-01-01", source = "yahoo")
spain_daily_prices <- na.omit(IBEX$IBEX.Adjusted)
names(spain_daily_prices) <- "Spain_IBEX35"
index(spain_daily_prices) <- as.Date(index(spain_daily_prices))
summary(spain_daily_prices)

```

```

getSymbols("^GSPC", from=as.Date("2020-01-01"))
sp500_daily_prices <- na.omit(GSPC$GSPC.Adjusted)
index(sp500_daily_prices) <- as.Date(index(sp500_daily_prices))
names(sp500_daily_prices) <- "USA_SP500"

```

```

getSymbols("^FTSE", from=as.Date("2020-01-01"))
ftse100_daily_prices <- FTSE$FTSE.Adjusted
index(ftse100_daily_prices) <- as.Date(index(ftse100_daily_prices))
names(ftse100_daily_prices) <- "UK_FTSE100"

```

```

event_date_2020 <- as.POSIXct("2020-03-11")
event_date_2020 <- as.Date("2020-03-11")
library("zoo")
library("xts")

```

```
PERCENT <- 100
```

```

aus <- australia_daily_prices
aus_first_price <- as.double(aus[1])
rel_aus <- aus * (1/aus_first_price)

```

```

can <- canada_daily_prices
names(can) <- "can"
can_first_price <- as.double(can[1])
rel_can <- can * (1/can_first_price)

```

```

fin <- finland_daily_prices
names(fin) <- "fin"
fin_first_price <- as.double(fin[1])
rel_fin <- fin * (1/fin_first_price)
print(rel_fin)

```

```

fra <- france_daily_prices
names(fra) <- "fra"
fra_first_price <- as.double(fra[1])
rel_fra <- fra * (1/fra_first_price)
print(rel_fra)

```

```

ger <- germany_daily_prices
names(ger) <- "ger"

```



```

ger_first_price <- as.double(ger[1])
rel_ger <- ger * (1/ger_first_price)
print(rel_ger)

ned <- netherlands_daily_prices
names(ned) <- "ned"
ned_first_price <- as.double(ned[1])
rel_ned <- ned * (1/ned_first_price)
print(rel_ned)

nor <- oseax_daily_prices
names(nor) <- "nor"
nor_first_price <- as.double(nor[1])
rel_nor <- nor * (1/nor_first_price)

spa <- spain_daily_prices
names(spa) <- "spa"
spa_first_price <- as.double(spa[1])
rel_spa <- spa * (1/spa_first_price)

swe <- omx_daily_prices
names(swe) <- "swe"
swe_first_price <- as.double(swe[1])
rel_swe <- swe * (1/swe_first_price)

usa <- sp500_daily_prices
names(usa) <- "usa"
usa_first_price <- as.double(usa[1])
rel_usa <- usa * (1/usa_first_price)

uk <- ftse100_daily_prices
names(uk) <- "uk"
uk_first_price <- as.double(uk[1])
rel_uk <- uk * (1/uk_first_price)

to_plot <- merge(rel_aus ,
                 rel_can ,
                 rel_fin ,
                 rel_fra ,
                 rel_ger ,
                 rel_ned ,
                 rel_nor ,
                 rel_spa ,
                 rel_swe ,
                 rel_uk ,
                 rel_usa) * PERCENT
to_plot <- na.locf(to_plot)
countries <- c("Australia",
               "Canada",
               "Finland",
               "France",

```

```

        "Germany",
        "Netherlands",
        "Norway",
        "Spain",
        "Sweden",
        "UK",
        "USA")

print(to_plot)
ofilename <- paste0(outdir, "relative_stock_index_levels.png")
png(ofilename, width=1000, height=600)
colors <- rainbow(12)
plot.zoo(to_plot,
        screens=c(1),
        type=c("l", "l"),
        lwd=c(3),
        lty=1:20,
        xlab="",
        ylab="",
        main="",
        col=colors,
        bty="n"
)
legend("bottomleft",
        legend=countries,
        col=colors,
        lty=1:20,
        lwd=c(2,2)
)
box(bty="L")
abline(v=event_date_2020, lty=2, lwd=2, col="grey")
dev.off()

```