COVID-19 HW

Rabya Ghafoor

4/9/2020

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------

## v tibble  2.1.3     v purrr   0.3.3
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

1.

```
dat <-
  read_csv("https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-counties.csv")
```

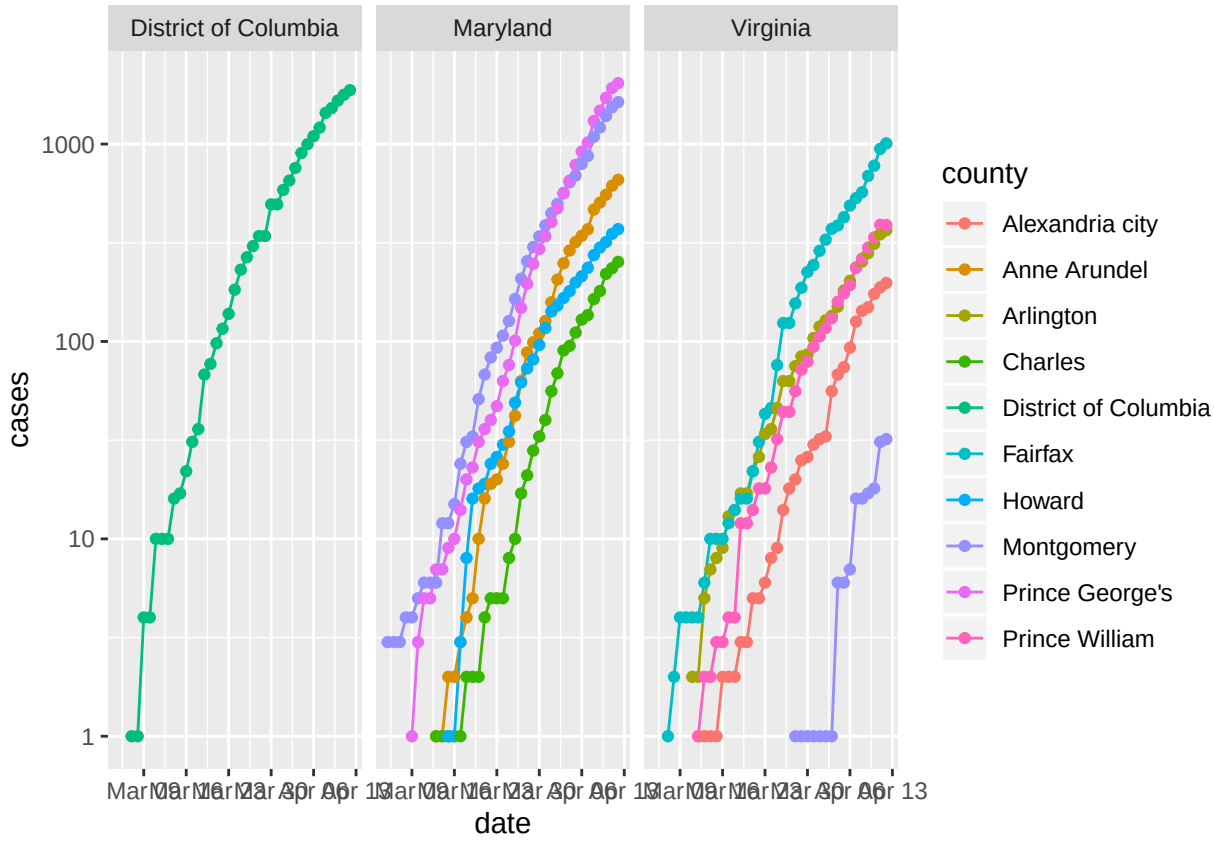
```
## Parsed with column specification:
## cols(
##   date = col_date(format = ""),
##   county = col_character(),
##   state = col_character(),
##   fips = col_character(),
##   cases = col_double(),
##   deaths = col_double()
## )
```

```
total_by_state <- dat %>%
  group_by(state, date) %>%
  summarize(total_deaths = sum(deaths), total_cases = sum(cases)) %>%
  arrange(desc(total_deaths))

total_by_state
```

```
## # A tibble: 2,269 x 4
## # Groups:   state [55]
##    state    date       total_deaths total_cases
##    <chr>    <date>            <dbl>       <dbl>
##  1 New York 2020-04-12         9385      188694
##  2 New York 2020-04-11         8627      180458
##  3 New York 2020-04-10         7844      170512
##  4 New York 2020-04-09         7067      159937
##  5 New York 2020-04-08         6268      149401
##  6 New York 2020-04-07         5563      140081
##  7 New York 2020-04-06         5505      130703
##  8 New York 2020-04-05         4161      122911
##  9 New York 2020-04-04         3568      114996
## 10 New York 2020-04-03         2935      102945
## # ... with 2,259 more rows
```

2.

```
dat_small <-
  dat %>%
 filter(state %in% c("District of Columbia", "Maryland", "Virginia"))  %>%
  mutate(county = factor(county))
dat_filter <-
  dat_small %>%
  filter(county %in% c("Alexandria city", "Anne Arundel", "Arlington", "Charles", "Arlington", "District

dat_filter %>%
  filter(state == "Virginia" | state == "Maryland" | state == "District of Columbia") %>%
  ggplot(aes(x = date, y = cases, group = county, col = county)) +
  geom_line() +
  geom_point() +
  facet_wrap(~ state) +
  scale_y_log10()
```



3.

```
dat <-
  read_csv("https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-counties.csv")
```

```
## Parsed with column specification:
## cols(
##   date = col_date(format = ""),
##   county = col_character(),
##   state = col_character(),
##   fips = col_character(),
##   cases = col_double(),
##   deaths = col_double()
## )
```

```
deaths_by_state <- dat %>%
  group_by(state,date) %>%
  summarize(total_deaths = sum(deaths)) %>%
  ungroup() %>%
  arrange(desc(total_deaths))
deaths_by_state %>%
  filter(date == max(date))
```

```
## # A tibble: 55 x 3
```

```
##    state         date       total_deaths
##    <chr>         <date>            <dbl>
##  1 New York      2020-04-12         9385
##  2 New Jersey    2020-04-12         2350
##  3 Michigan      2020-04-12         1488
##  4 Louisiana     2020-04-12          840
##  5 Massachusetts 2020-04-12          756
##  6 Illinois      2020-04-12          727
##  7 California    2020-04-12          676
##  8 Connecticut   2020-04-12          554
##  9 Pennsylvania  2020-04-12          525
## 10 Washington    2020-04-12          511
## # ... with 45 more rows
```

```
cases_by_state <- dat %>%
  group_by(state, date) %>%
  summarize(total_cases = sum(cases)) %>%
  ungroup() %>%
  arrange(desc(total_cases))
cases_by_state
```

```
## # A tibble: 2,269 x 3
##    state    date       total_cases
##    <chr>    <date>           <dbl>
##  1 New York 2020-04-12      188694
##  2 New York 2020-04-11      180458
##  3 New York 2020-04-10      170512
##  4 New York 2020-04-09      159937
##  5 New York 2020-04-08      149401
##  6 New York 2020-04-07      140081
##  7 New York 2020-04-06      130703
##  8 New York 2020-04-05      122911
##  9 New York 2020-04-04      114996
## 10 New York 2020-04-03      102945
## # ... with 2,259 more rows
```

```
DMV_case <- cases_by_state  %>%
  filter(state %in% c("District of Columbia", "Maryland", "Virginia")) %>%
  group_by(date) %>%
  summarize(total_cases_date = sum(total_cases)) %>%
  ungroup() %>%
  arrange(desc(total_cases_date))
DMV_case
```

```
## # A tibble: 39 x 2
##    date       total_cases_date
##    <date>                <dbl>
##  1 2020-04-12            15372
##  2 2020-04-11            14549
##  3 2020-04-10            13138
##  4 2020-04-09            11748
##  5 2020-04-08            10613
##  6 2020-04-07             8915
```

```
##  7 2020-04-06             8020
##  8 2020-04-05             7262
##  9 2020-04-04             6454
## 10 2020-04-03             5527
## # ... with 29 more rows
```

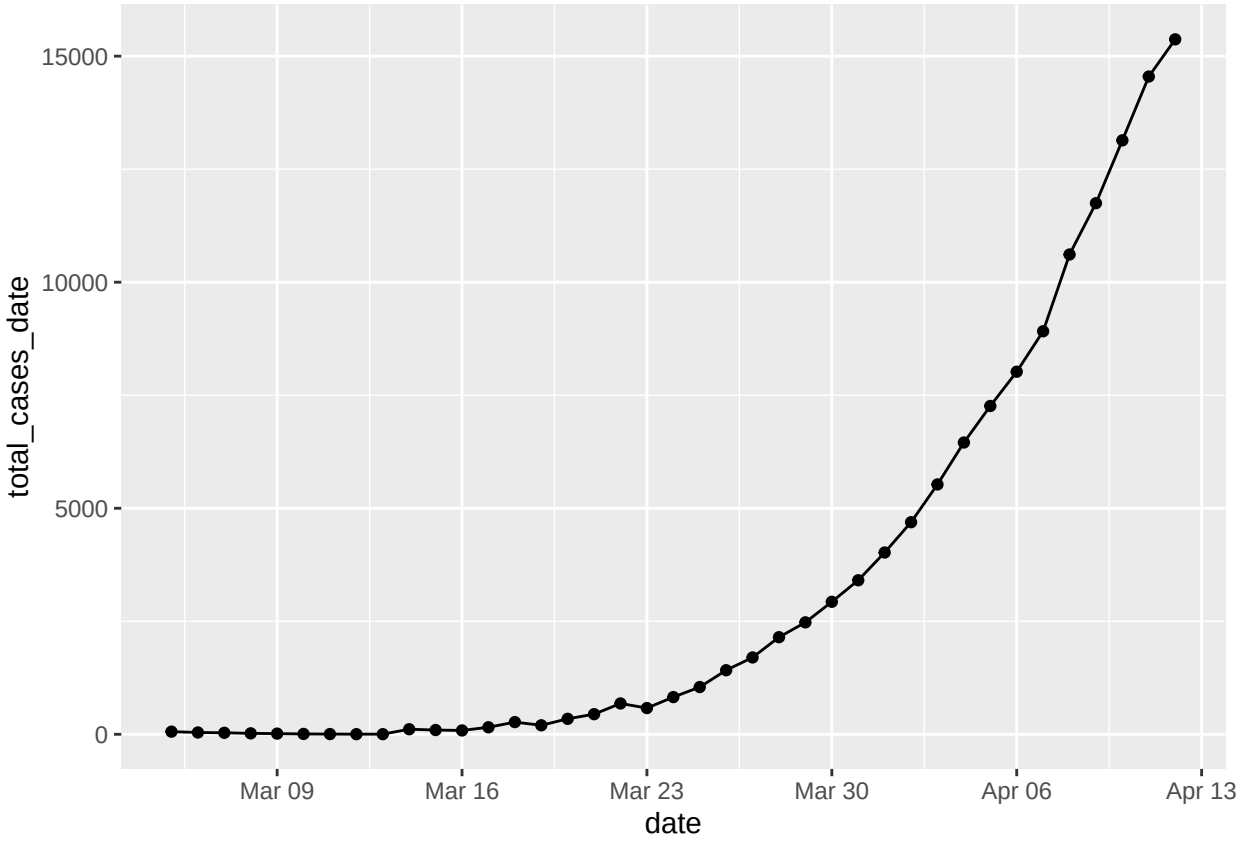
```
DMV_death <- deaths_by_state  %>%
  filter(state %in% c("District of Columbia", "Maryland", "Virginia")) %>%
  group_by(date) %>%
  summarize(total_deaths_date = sum(total_deaths)) %>%
  ungroup() %>%
  arrange(desc(total_deaths_date))
DMV_death
```

```
## # A tibble: 39 x 2
##    date       total_deaths_date
##    <date>                 <dbl>
##  1 2020-04-12               427
##  2 2020-04-11               384
##  3 2020-04-10               331
##  4 2020-04-09               280
##  5 2020-04-08               211
##  6 2020-04-07               195
##  7 2020-04-06               170
##  8 2020-04-05               142
##  9 2020-04-04               127
## 10 2020-04-03               103
## # ... with 29 more rows
```

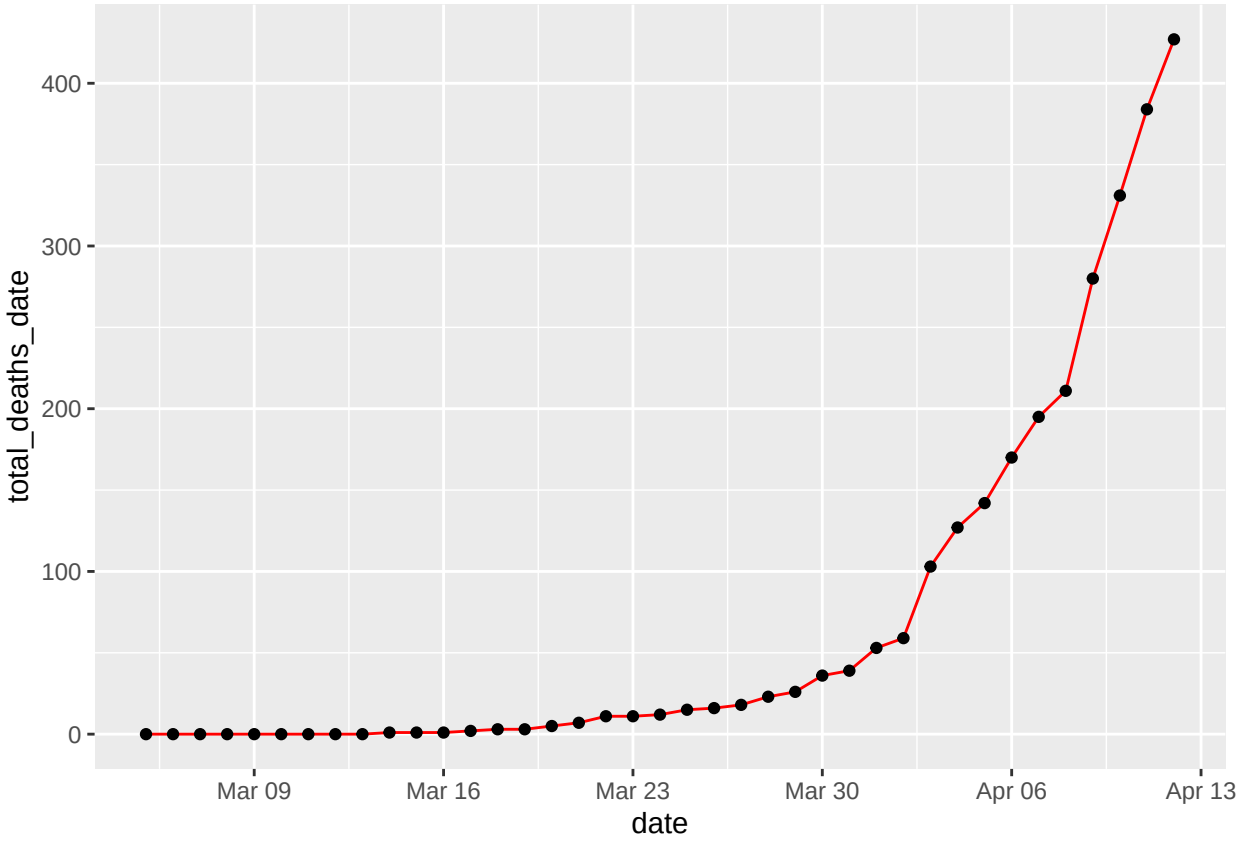
```
DMV <- DMV_death %>%
  mutate(total_cases_date = DMV_case$total_cases_date)
DMV
```

```
## # A tibble: 39 x 3
##    date       total_deaths_date total_cases_date
##    <date>                 <dbl>            <dbl>
##  1 2020-04-12               427            15372
##  2 2020-04-11               384            14549
##  3 2020-04-10               331            13138
##  4 2020-04-09               280            11748
##  5 2020-04-08               211            10613
##  6 2020-04-07               195             8915
##  7 2020-04-06               170             8020
##  8 2020-04-05               142             7262
##  9 2020-04-04               127             6454
## 10 2020-04-03               103             5527
## # ... with 29 more rows
```

```
DMV %>%
  ggplot(aes(x = date, y = total_cases_date)) +
  geom_line() +
  geom_point ()
```




```
DMV %>%
 ggplot(aes(x = date, y = total_deaths_date)) +
  geom_line(color = 'red') +
  geom_point ()
```



4.

```
#Recieved help from Erin
library(lubridate)
library(tidyr)
wdat <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/archived_data/archive

wdat$`Province/State`[is.na(wdat$`Province/State`)] <- " "
CS <- paste0(wdat$`Country/Region` , sep = " ", wdat$`Province/State`)
wdat1 <- wdat %>%
  mutate("Country/State" = CS)
wdat1$`Province/State`[wdat$`Province/State` == " "] <- "NA"
wdat_clean <- wdat1 %>%
  dplyr::select(`Country/State` , `Province/State` , `Country/Region`, everything()) %>%
  pivot_longer(cols = -c(`Country/State`, `Province/State` , `Country/Region` , Lat, Long), names_to =
  mutate(Lat = as.integer(Lat)) %>%
  mutate(Long = as.integer(Long)) %>%
  mutate(Date = as.character(Date)) %>%
  mutate(Date = as.Date(Date, "%m/%d/%y"))
wdat_clean
```

```
## # A tibble: 30,870 x 7
##    `Country/State` `Province/State` `Country/Region`   Lat  Long Date
##    <chr>           <chr>            <chr>            <int> <int> <date>
##  1 "Thailand  "    NA               Thailand            15   101 2020-01-22
```

```
##  2 "Thailand  "    NA               Thailand            15   101 2020-01-23
##  3 "Thailand  "    NA               Thailand            15   101 2020-01-24
##  4 "Thailand  "    NA               Thailand            15   101 2020-01-25
##  5 "Thailand  "    NA               Thailand            15   101 2020-01-26
##  6 "Thailand  "    NA               Thailand            15   101 2020-01-27
##  7 "Thailand  "    NA               Thailand            15   101 2020-01-28
##  8 "Thailand  "    NA               Thailand            15   101 2020-01-29
##  9 "Thailand  "    NA               Thailand            15   101 2020-01-30
## 10 "Thailand  "    NA               Thailand            15   101 2020-01-31
## # ... with 30,860 more rows, and 1 more variable: Confirmed_Cases <dbl>
```

5.

```
library(countrycode)
continent <- countrycode(sourcevar = wdat_clean[[3]], origin = "country.name", destination = "continent
```

```
## Warning in countrycode(sourcevar = wdat_clean[[3]], origin = "country.name", : Some values were not r
```

```
new_wdat <- wdat_clean %>%
  mutate(continet = continent) %>%
  mutate(continent= case_when(
    `Country/Region` == "Akrotiri and Dhekelia" ~ "South-Eastern Asia",
    `Country/Region` == "Central African Rep." ~ "Middle Africa",
    `Country/Region` == "Channel Islands" ~ "North America",
    `Country/Region`== "Christmas Island" ~ "Australia and New Zealand",
    `Country/Region` == "Cocos Island" ~ "Australia and New Zealand",
    `Country/Region` == "Czechoslovakia" ~ "Eastern Europe",
    `Country/Region` == "East Germany" ~ "Western Europe",
    `Country/Region` == "Eritrea and Ethiopia" ~ "Eastern Africa",
    `Country/Region` == "Kosovo" ~ "Eastern Europe",
    `Country/Region` == "North Yemen (former)" ~ "South-Eastern Asia",
    `Country/Region` == "Serbia amd Montenegro" ~ "Eastern Europe",
    `Country/Region` == "South Yemen (former)" ~ "South-Eastern Asia",
    `Country/Region` == "St. Martin" ~ "Caribbean",
    `Country/Region` == "Yuogoslavia" ~ "Eastern Europe",
    TRUE ~ continent)) %>%
  select(continent,  `Country/Region`, `Date`, everything()) %>%
  arrange(continent,  `Country/Region`, `Date`)

tbl_df(new_wdat)
```

```
## # A tibble: 30,870 x 9
##    continent `Country/Region` Date       `Country/State` `Province/State`   Lat
##    <chr>     <chr>            <date>     <chr>           <chr>            <int>
##  1 Africa    Algeria          2020-01-22 "Algeria  "     NA                  28
##  2 Africa    Algeria          2020-01-23 "Algeria  "     NA                  28
##  3 Africa    Algeria          2020-01-24 "Algeria  "     NA                  28
##  4 Africa    Algeria          2020-01-25 "Algeria  "     NA                  28
##  5 Africa    Algeria          2020-01-26 "Algeria  "     NA                  28
##  6 Africa    Algeria          2020-01-27 "Algeria  "     NA                  28
##  7 Africa    Algeria          2020-01-28 "Algeria  "     NA                  28
##  8 Africa    Algeria          2020-01-29 "Algeria  "     NA                  28
```

```
##  9 Africa    Algeria          2020-01-30 "Algeria  "     NA                  28
## 10 Africa    Algeria          2020-01-31 "Algeria  "     NA                  28
## # ... with 30,860 more rows, and 3 more variables: Long <int>,
## #   Confirmed_Cases <dbl>, continet <chr>
```

```
new_wdat$`continent`[is.na(new_wdat$`continent`)] <- " "
new_wdat <- subset(new_wdat, select = -c(continet) )
new_wdat
```

```
## # A tibble: 30,870 x 8
##    continent `Country/Region` Date       `Country/State` `Province/State`   Lat
##    <chr>     <chr>            <date>     <chr>           <chr>             <int>
##  1 Africa    Algeria          2020-01-22 "Algeria  "     NA                  28
##  2 Africa    Algeria          2020-01-23 "Algeria  "     NA                  28
##  3 Africa    Algeria          2020-01-24 "Algeria  "     NA                  28
##  4 Africa    Algeria          2020-01-25 "Algeria  "     NA                  28
##  5 Africa    Algeria          2020-01-26 "Algeria  "     NA                  28
##  6 Africa    Algeria          2020-01-27 "Algeria  "     NA                  28
##  7 Africa    Algeria          2020-01-28 "Algeria  "     NA                  28
##  8 Africa    Algeria          2020-01-29 "Algeria  "     NA                  28
##  9 Africa    Algeria          2020-01-30 "Algeria  "     NA                  28
## 10 Africa    Algeria          2020-01-31 "Algeria  "     NA                  28
## # ... with 30,860 more rows, and 2 more variables: Long <int>,
## #   Confirmed_Cases <dbl>
```

6.

```
new_max <- new_wdat %>%
  group_by(`Country/State`,  `Country/Region`, `continent`) %>%
  summarize(ttl = sum(Confirmed_Cases)) %>%
  ungroup() %>%
  arrange(desc(ttl)) %>%
  slice(1:25)
new_max
```

```
## # A tibble: 25 x 4
##    `Country/State`   `Country/Region` continent     ttl
##    <chr>             <chr>            <chr>       <dbl>
##  1 "China Hubei"     China            Asia      2894885
##  2 "Italy  "         Italy            Europe     497959
##  3 "Iran  "          Iran             Asia       252770
##  4 "Spain  "         Spain            Europe     186200
##  5 "Korea, South  "  Korea, South     Asia       181699
##  6 "Germany  "       Germany          Europe     160974
##  7 "France France"   France           Europe     117724
##  8 "China Guangdong" China            Asia        67015
##  9 "US New York"     US               Americas    64538
## 10 "China Henan"     China            Asia        61811
## # ... with 15 more rows
```

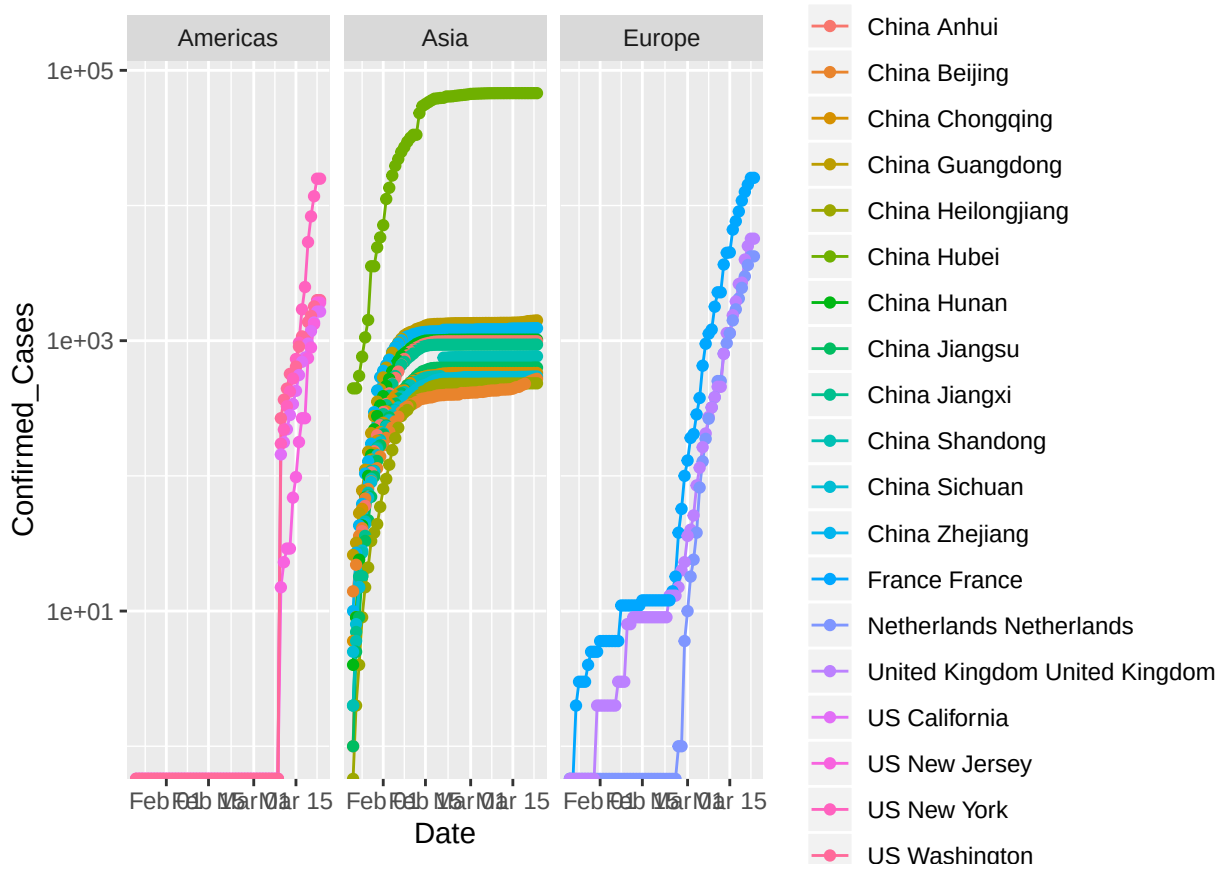
7.

```
new_graph <- new_wdat %>%
  group_by(`Country/State`,  `Country/Region`, `continent`, `Date`) %>%
  summarize(Confirmed_Cases = sum(Confirmed_Cases)) %>%
  ungroup() %>%
  arrange(desc(Confirmed_Cases))

new_graph
```

```
## # A tibble: 30,870 x 5
##    `Country/State` `Country/Region` continent Date       Confirmed_Cases
##    <chr>           <chr>            <chr>     <date>                <dbl>
##  1 China Hubei     China            Asia      2020-03-18            67800
##  2 China Hubei     China            Asia      2020-03-19            67800
##  3 China Hubei     China            Asia      2020-03-20            67800
##  4 China Hubei     China            Asia      2020-03-21            67800
##  5 China Hubei     China            Asia      2020-03-22            67800
##  6 China Hubei     China            Asia      2020-03-23            67800
##  7 China Hubei     China            Asia      2020-03-17            67799
##  8 China Hubei     China            Asia      2020-03-16            67798
##  9 China Hubei     China            Asia      2020-03-15            67794
## 10 China Hubei     China            Asia      2020-03-14            67790
## # ... with 30,860 more rows
```

```
wdat_small <-
  new_graph %>%
 filter(continent %in% c("Asia", "Americas", "Europe"))  %>%
  mutate(`Country/State` = factor(`Country/State`))
wdat_filter <-
  wdat_small %>%
  filter(`Country/State` %in% c("China Guangdong", "China Hunan" , "China Hubei" , "China Zhejiang", "Ch

wdat_filter %>%
  filter(continent == "Asia" | continent == "Americas" | continent == "Europe") %>%
  ggplot(aes(x = `Date`, y = `Confirmed_Cases`, group = `Country/State`, col = `Country/State`)) +
  geom_line() +
  geom_point() +
  facet_wrap(~ continent) +
  scale_y_log10()
```

```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis
```



8.

```
NY_csse <- new_graph %>%
 filter(`Country/State` %in% c("US New York")) %>%
  group_by(Date) %>%
  summarize(total_cases_date = sum(Confirmed_Cases)) %>%
  ungroup() %>%
  arrange(desc(total_cases_date))
NY_csse
```

```
## # A tibble: 62 x 2
##    Date       total_cases_date
##    <date>                <dbl>
##  1 2020-03-22            15793
##  2 2020-03-23            15793
##  3 2020-03-21            11710
##  4 2020-03-20             8310
##  5 2020-03-19             5365
##  6 2020-03-18             2495
##  7 2020-03-17             1706
##  8 2020-03-16              967
##  9 2020-03-15              732
## 10 2020-03-14              525
## # ... with 52 more rows
```

```r
NY_times <- cases_by_state  %>%
  filter(state %in% c("New York")) %>%
  group_by(date) %>%
  summarize(total_cases_date = sum(total_cases)) %>%
  ungroup() %>%
  arrange(desc(total_cases_date))
NY_times
```

```
## # A tibble: 43 x 2
##    date       total_cases_date
##    <date>                <dbl>
##  1 2020-04-12           188694
##  2 2020-04-11           180458
##  3 2020-04-10           170512
##  4 2020-04-09           159937
##  5 2020-04-08           149401
##  6 2020-04-07           140081
##  7 2020-04-06           130703
##  8 2020-04-05           122911
##  9 2020-04-04           114996
## 10 2020-04-03           102945
## # ... with 33 more rows
```

```r
merge(x = NY_csse, y = NY_times, by = "total_cases_date", all = TRUE)
```

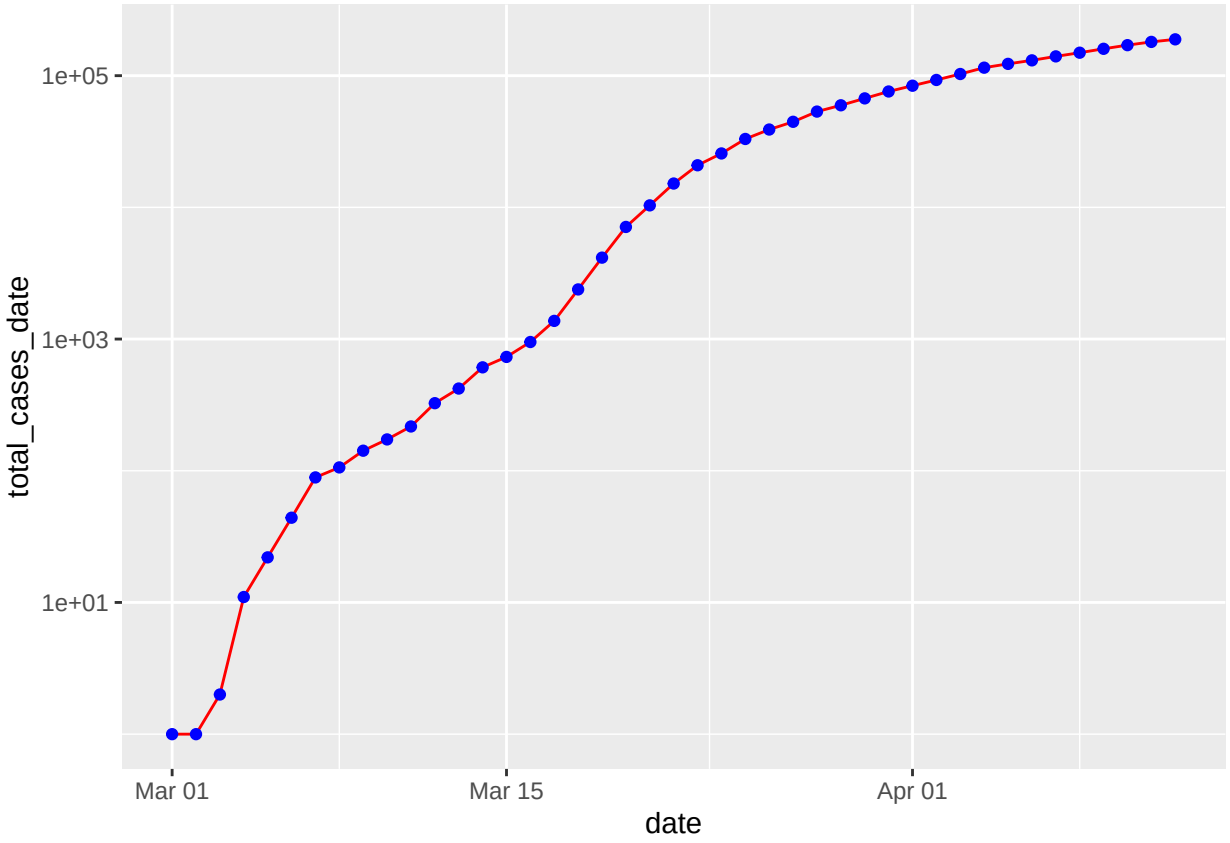
```
##     total_cases_date       Date       date
## 1                  0 2020-01-29       <NA>
## 2                  0 2020-01-31       <NA>
## 3                  0 2020-02-03       <NA>
## 4                  0 2020-01-22       <NA>
## 5                  0 2020-01-23       <NA>
## 6                  0 2020-01-24       <NA>
## 7                  0 2020-01-25       <NA>
## 8                  0 2020-01-26       <NA>
## 9                  0 2020-01-27       <NA>
## 10                 0 2020-01-28       <NA>
## 11                 0 2020-02-11       <NA>
## 12                 0 2020-01-30       <NA>
## 13                 0 2020-02-13       <NA>
## 14                 0 2020-02-01       <NA>
## 15                 0 2020-02-02       <NA>
## 16                 0 2020-02-16       <NA>
## 17                 0 2020-02-04       <NA>
## 18                 0 2020-02-05       <NA>
## 19                 0 2020-02-06       <NA>
## 20                 0 2020-02-07       <NA>
## 21                 0 2020-02-08       <NA>
## 22                 0 2020-02-09       <NA>
## 23                 0 2020-02-10       <NA>
## 24                 0 2020-02-24       <NA>
## 25                 0 2020-02-12       <NA>
## 26                 0 2020-02-26       <NA>
```

```
## 27                0 2020-02-14       <NA>
## 28                0 2020-02-15       <NA>
## 29                0 2020-02-29       <NA>
## 30                0 2020-02-17       <NA>
## 31                0 2020-02-18       <NA>
## 32                0 2020-02-19       <NA>
## 33                0 2020-02-20       <NA>
## 34                0 2020-02-21       <NA>
## 35                0 2020-02-22       <NA>
## 36                0 2020-02-23       <NA>
## 37                0 2020-03-08       <NA>
## 38                0 2020-02-25       <NA>
## 39                0 2020-03-01       <NA>
## 40                0 2020-02-27       <NA>
## 41                0 2020-02-28       <NA>
## 42                0 2020-03-04       <NA>
## 43                0 2020-03-05       <NA>
## 44                0 2020-03-02       <NA>
## 45                0 2020-03-03       <NA>
## 46                0 2020-03-09       <NA>
## 47                0 2020-03-06       <NA>
## 48                0 2020-03-07       <NA>
## 49                1       <NA> 2020-03-01
## 50                1       <NA> 2020-03-02
## 51                2       <NA> 2020-03-03
## 52               11       <NA> 2020-03-04
## 53               22       <NA> 2020-03-05
## 54               44       <NA> 2020-03-06
## 55               89       <NA> 2020-03-07
## 56              106       <NA> 2020-03-08
## 57              142       <NA> 2020-03-09
## 58              173 2020-03-10 2020-03-10
## 59              217       <NA> 2020-03-11
## 60              220 2020-03-11       <NA>
## 61              326       <NA> 2020-03-12
## 62              328 2020-03-12       <NA>
## 63              421 2020-03-13 2020-03-13
## 64              525 2020-03-14       <NA>
## 65              610       <NA> 2020-03-14
## 66              732 2020-03-15 2020-03-15
## 67              950       <NA> 2020-03-16
## 68              967 2020-03-16       <NA>
## 69             1374       <NA> 2020-03-17
## 70             1706 2020-03-17       <NA>
## 71             2382       <NA> 2020-03-18
## 72             2495 2020-03-18       <NA>
## 73             4152       <NA> 2020-03-19
## 74             5365 2020-03-19       <NA>
## 75             7102       <NA> 2020-03-20
## 76             8310 2020-03-20       <NA>
## 77            10356       <NA> 2020-03-21
## 78            11710 2020-03-21       <NA>
## 79            15168       <NA> 2020-03-22
## 80            15793 2020-03-22       <NA>
```

```
## 81             15793 2020-03-23       <NA>
## 82             20875       <NA> 2020-03-23
## 83             25666       <NA> 2020-03-24
## 84             33067       <NA> 2020-03-25
## 85             38988       <NA> 2020-03-26
## 86             44636       <NA> 2020-03-27
## 87             53364       <NA> 2020-03-28
## 88             59568       <NA> 2020-03-29
## 89             67216       <NA> 2020-03-30
## 90             75832       <NA> 2020-03-31
## 91             83890       <NA> 2020-04-01
## 92             92770       <NA> 2020-04-02
## 93            102945       <NA> 2020-04-03
## 94            114996       <NA> 2020-04-04
## 95            122911       <NA> 2020-04-05
## 96            130703       <NA> 2020-04-06
## 97            140081       <NA> 2020-04-07
## 98            149401       <NA> 2020-04-08
## 99            159937       <NA> 2020-04-09
## 100           170512       <NA> 2020-04-10
## 101           180458       <NA> 2020-04-11
## 102           188694       <NA> 2020-04-12
```

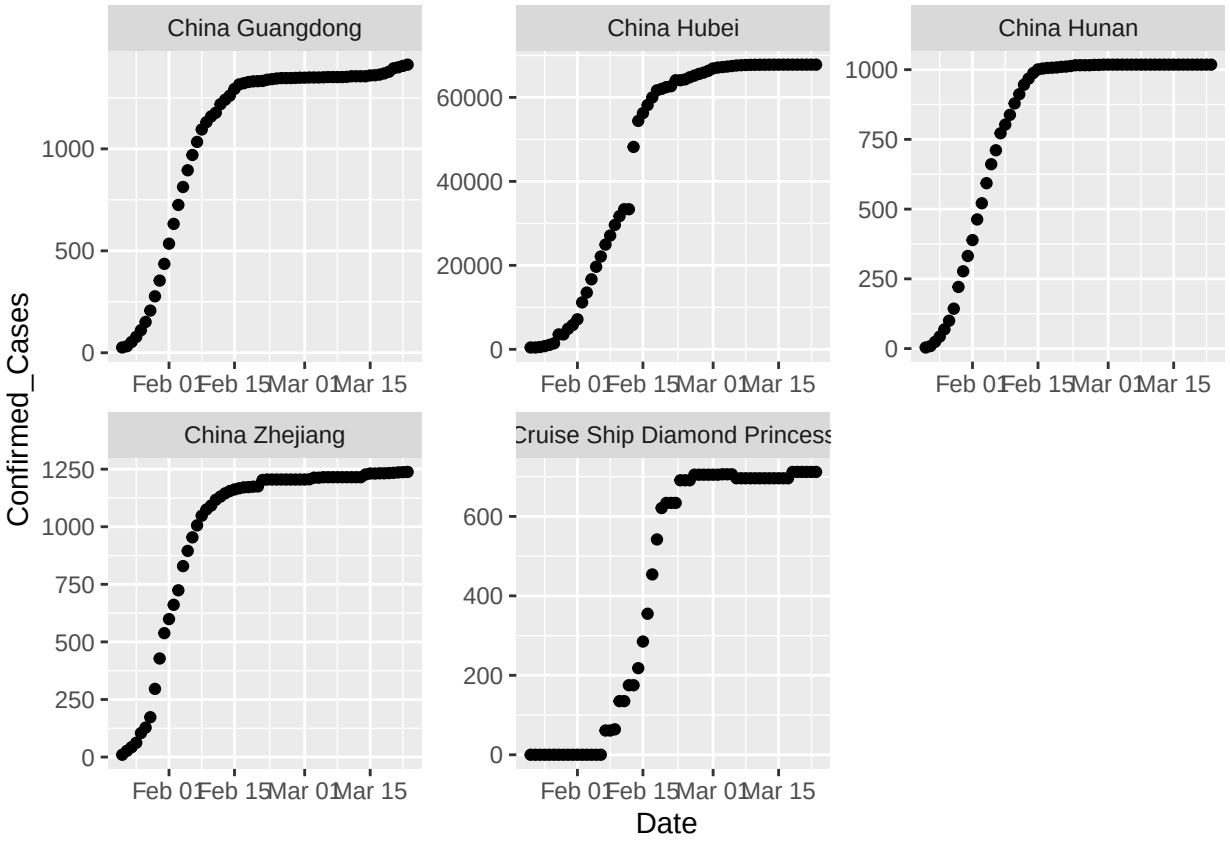
When comparing both data frames the date does not entirely match in terms of the number of cases reported each day. Although on some day both sources have reported the same number of cases, in general the NY times overestiamtes the number of confirmed cases by a couple of thousands.

```
NY_times %>%
  ggplot(aes(x = date, y = total_cases_date)) +
  geom_line(color = 'red') +
  geom_point (color = 'blue') +
  scale_y_log10()
```



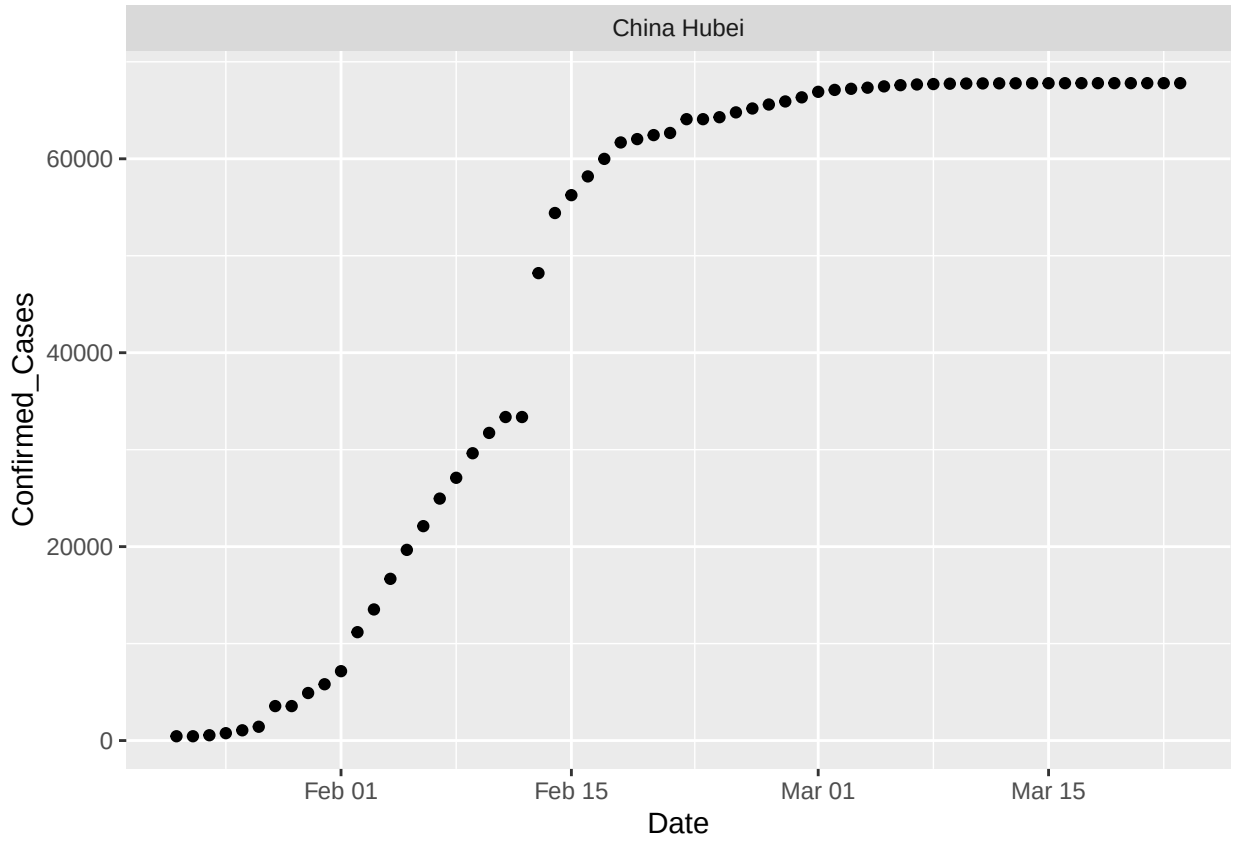
9.

```
Slowed_cases <- new_wdat %>%
  filter(`Country/State` %in% c("China Hubei", "Cruise Ship Diamond Princess", "China Hunan", "China Gu

Slowed_cases %>%
  ggplot(aes(x = Date, y = Confirmed_Cases)) +
  geom_point() +
  facet_wrap(~ `Country/State`, scales = "free")
```



10.

```
Hubei_cases <- new_wdat %>%
  filter(`Country/State` %in% c("China Hubei" ))

Hubei_cases %>%
  ggplot(aes(x = Date, y = Confirmed_Cases)) +
  geom_point() +
  facet_wrap(~ `Country/State`, scales = "free")
```




```
Hubei1_cases <- new_wdat %>%
  filter(`Country/State` %in% c("China Hubei" )) %>%
  mutate(Date = as.integer(Date))

x = Hubei1_cases$Date
y = Hubei1_cases$Confirmed_Cases
```

```
sigmoid = function(x, params) {
  params[1] / (1 + exp(-params[2] * (x - params[3])))
}


fitmodel <- nls(y ~ K /(1 + exp(-B * (x - t0))), data = Hubei1_cases,
                start = list(K = 60000, B = 0.5,  t0= 18300))
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
library(broom)
summary(fitmodel)
```

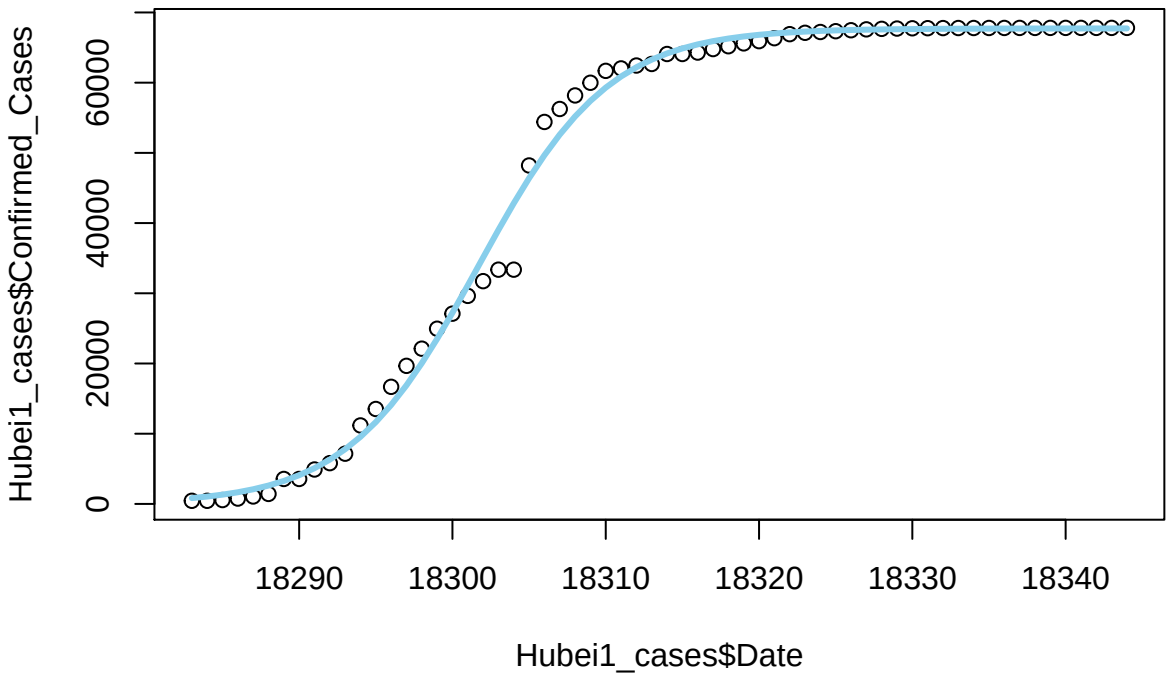
```
##
## Formula: y ~ K/(1 + exp(-B * (x - t0)))
```

```
##
## Parameters:
##     Estimate Std. Error   t value Pr(>|t|)
## K  6.773e+04  3.839e+02    176.44   <2e-16 ***
## B  2.348e-01  7.916e-03     29.66   <2e-16 ***
## t0 1.830e+04  1.660e-01 110241.51   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2002 on 59 degrees of freedom
##
## Number of iterations to convergence: 6
## Achieved convergence tolerance: 2.132e-06
```

```
broom::glance(fitmodel)
```

```
## # A tibble: 1 x 8
##   sigma isConv     finTol logLik   AIC   BIC   deviance df.residual
##   <dbl> <lgl>       <dbl>  <dbl> <dbl> <dbl>      <dbl>       <int>
## 1 2002. TRUE   0.00000213  -558. 1123. 1132. 236424289.          59
```

```
plot(Hubei1_cases$Date, Hubei1_cases$Confirmed_Cases)
lines(Hubei1_cases$Date, predict(fitmodel, list(x = Hubei1_cases$Date)), col = 'skyblue', lwd = 3)
```



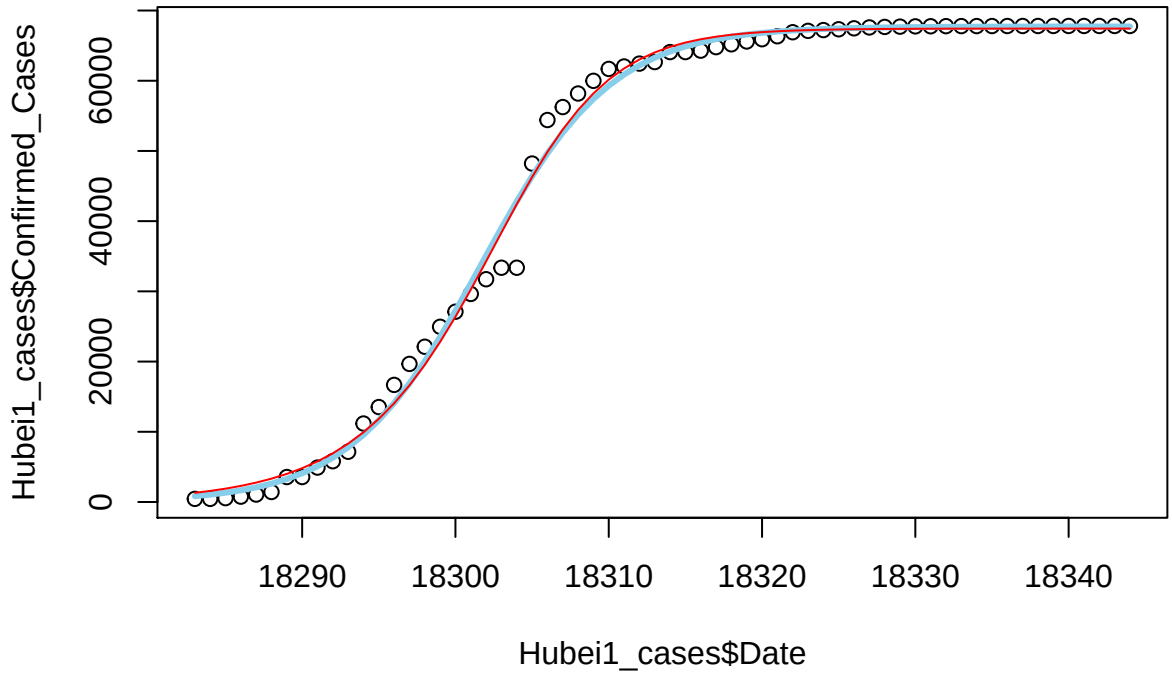
11.

```
fitmodel2 <- nls(y ~ K /(1 + exp(-B * (x - t0)))^(1/v), data = Hubei1_cases,
                 start = list(K = 60000, B = 0.5,  t0= 18300, v = 1.490 ))
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
plot(Hubei1_cases$Date, Hubei1_cases$Confirmed_Cases)
lines(Hubei1_cases$Date, predict(fitmodel, list(x = Hubei1_cases$Date)), col = 'skyblue', lwd = 3)
lines(Hubei1_cases$Date, predict(fitmodel2, list(x = Hubei1_cases$Date)), col = 'red', lwd = 1)
```




```
summary(fitmodel2)
```

```
##
## Formula: y ~ K/(1 + exp(-B * (x - t0)))^(1/v)
##
## Parameters:
##     Estimate Std. Error   t value Pr(>|t|)
## K  6.743e+04  3.909e+02   172.491  < 2e-16 ***
## B  2.800e-01  2.746e-02    10.196 1.50e-14 ***
## t0 1.830e+04  1.018e+00 17984.491  < 2e-16 ***
## v  1.490e+00  2.885e-01     5.164 3.09e-06 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1976 on 58 degrees of freedom
##
## Number of iterations to convergence: 11
## Achieved convergence tolerance: 9.012e-06
```

```
broom::glance(fitmodel2)
```

```
## # A tibble: 1 x 8
##   sigma isConv     finTol logLik   AIC   BIC   deviance df.residual
##   <dbl> <lgl>       <dbl>  <dbl> <dbl> <dbl>      <dbl>       <int>
## 1 1976. TRUE   0.00000901  -556. 1123. 1133. 226420388.          58
```

12.

```
library(purrr)
library(modelr)
```

```
##
## Attaching package: 'modelr'
```

```
## The following object is masked from 'package:broom':
##
##     bootstrap
```

```
by_country <- Slowed_cases %>%
    group_by(`Country/State`, continent) %>%
    nest()
by_country %>% head
```

```
## # A tibble: 5 x 3
## # Groups:   continent, Country/State [5]
##   continent `Country/State`              data
##   <chr>     <chr>                        <list>
## 1 "Asia"    China Hubei                  <tibble [62 x 6]>
## 2 "Asia"    China Guangdong              <tibble [62 x 6]>
## 3 "Asia"    China Zhejiang               <tibble [62 x 6]>
## 4 "Asia"    China Hunan                  <tibble [62 x 6]>
## 5 " "       Cruise Ship Diamond Princess <tibble [62 x 6]>
```

```
country_model <- function(df) {
    nls(y ~ K /(1 + exp(-B * (x - t0)))^(1/v), data = df,
                start = list(K = 60000, B = 0.5,  t0= 18300, v = 1.490 ))
}
```

```
by_country <- by_country %>%
    mutate(model = map(.x = data, .f = country_model))
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
by_country %>% head()
```

```
## # A tibble: 5 x 4
## # Groups:   continent, Country/State [5]
##   continent `Country/State`              data              model
##   <chr>     <chr>                        <list>            <list>
## 1 "Asia"    China Hubei                  <tibble [62 x 6]> <nls>
## 2 "Asia"    China Guangdong              <tibble [62 x 6]> <nls>
## 3 "Asia"    China Zhejiang               <tibble [62 x 6]> <nls>
## 4 "Asia"    China Hunan                  <tibble [62 x 6]> <nls>
## 5 " "       Cruise Ship Diamond Princess <tibble [62 x 6]> <nls>
```

```
by_country <- by_country %>%
    mutate(
        preds  = map2(.x = data, .y = model, .f = add_predictions),
        resids = map2(.x = data, .y = model, .f = add_residuals)
    )
by_country %>% head()
```

```
## # A tibble: 5 x 6
## # Groups:   continent, Country/State [5]
##   continent `Country/State`         data         model  preds       resids
##   <chr>     <chr>                   <list>       <list> <list>      <list>
## 1 "Asia"    China Hubei             <tibble [62~ <nls>  <tibble [6~ <tibble [62~
## 2 "Asia"    China Guangdong         <tibble [62~ <nls>  <tibble [6~ <tibble [62~
## 3 "Asia"    China Zhejiang          <tibble [62~ <nls>  <tibble [6~ <tibble [62~
## 4 "Asia"    China Hunan             <tibble [62~ <nls>  <tibble [6~ <tibble [62~
## 5 " "       Cruise Ship Diamond Pr~ <tibble [62~ <nls>  <tibble [6~ <tibble [62~
```

#Unnesting

```
preds <- unnest(data = by_country, preds)
preds %>% head()
```

```
## # A tibble: 6 x 12
## # Groups:   continent, Country/State [1]
##   continent `Country/State` data  model `Country/Region` Date
##   <chr>     <chr>           <lis> <lis> <chr>            <date>
## 1 Asia      China Hubei     <tib~ <nls> China            2020-01-22
## 2 Asia      China Hubei     <tib~ <nls> China            2020-01-23
## 3 Asia      China Hubei     <tib~ <nls> China            2020-01-24
## 4 Asia      China Hubei     <tib~ <nls> China            2020-01-25
## 5 Asia      China Hubei     <tib~ <nls> China            2020-01-26
## 6 Asia      China Hubei     <tib~ <nls> China            2020-01-27
## # ... with 6 more variables: `Province/State` <chr>, Lat <int>, Long <int>,
## #   Confirmed_Cases <dbl>, pred <dbl>, resids <list>
```

```
resids <- unnest(data = by_country, resids)
resids %>% head()
```

```
## # A tibble: 6 x 12
## # Groups:   continent, Country/State [1]
##   continent `Country/State` data  model preds `Country/Region` Date
##   <chr>     <chr>           <lis> <lis> <lis> <chr>            <date>
## 1 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-22
## 2 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-23
## 3 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-24
## 4 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-25
## 5 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-26
## 6 Asia      China Hubei     <tib~ <nls> <tib~ China            2020-01-27
## # ... with 5 more variables: `Province/State` <chr>, Lat <int>, Long <int>,
## #   Confirmed_Cases <dbl>, resid <dbl>
```

```
tidy(country_model(by_country))
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```

```
## # A tibble: 4 x 5
##   term   estimate std.error statistic   p.value
##   <chr>     <dbl>     <dbl>     <dbl>     <dbl>
## 1 K     67430.     391.        172.   2.52e- 80
## 2 B         0.280    0.0275     10.2  1.50e- 14
## 3 t0    18304.       1.02    17984.   2.36e-197
## 4 v         1.49     0.289       5.16 3.09e-  6
```

```
tidy <- by_country %>%
    mutate(tidy = map(model, broom::tidy)) %>%
    unnest(tidy, .drop = TRUE)
```

```
## Warning: The `.drop` argument of `unnest()` is deprecated as of tidyr 1.0.0.
## All list-columns are now preserved.
## This warning is displayed once per session.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
tidy
```

```
## # A tibble: 20 x 11
## # Groups:   continent, Country/State [5]
##    continent `Country/State` data  model preds resids term  estimate std.error
##    <chr>     <chr>           <lis> <lis> <lis> <list> <chr>    <dbl>     <dbl>
##  1 "Asia"    China Hubei     <tib~ <nls> <tib~ <tibb~ K      6.74e+4  391.
##  2 "Asia"    China Hubei     <tib~ <nls> <tib~ <tibb~ B      2.80e-1    0.0275
##  3 "Asia"    China Hubei     <tib~ <nls> <tib~ <tibb~ t0     1.83e+4    1.02
##  4 "Asia"    China Hubei     <tib~ <nls> <tib~ <tibb~ v      1.49e+0    0.289
##  5 "Asia"    China Guangdong <tib~ <nls> <tib~ <tibb~ K      6.74e+4  391.
##  6 "Asia"    China Guangdong <tib~ <nls> <tib~ <tibb~ B      2.80e-1    0.0275
##  7 "Asia"    China Guangdong <tib~ <nls> <tib~ <tibb~ t0     1.83e+4    1.02
##  8 "Asia"    China Guangdong <tib~ <nls> <tib~ <tibb~ v      1.49e+0    0.289
##  9 "Asia"    China Zhejiang  <tib~ <nls> <tib~ <tibb~ K      6.74e+4  391.
## 10 "Asia"    China Zhejiang  <tib~ <nls> <tib~ <tibb~ B      2.80e-1    0.0275
## 11 "Asia"    China Zhejiang  <tib~ <nls> <tib~ <tibb~ t0     1.83e+4    1.02
## 12 "Asia"    China Zhejiang  <tib~ <nls> <tib~ <tibb~ v      1.49e+0    0.289
## 13 "Asia"    China Hunan     <tib~ <nls> <tib~ <tibb~ K      6.74e+4  391.
## 14 "Asia"    China Hunan     <tib~ <nls> <tib~ <tibb~ B      2.80e-1    0.0275
## 15 "Asia"    China Hunan     <tib~ <nls> <tib~ <tibb~ t0     1.83e+4    1.02
## 16 "Asia"    China Hunan     <tib~ <nls> <tib~ <tibb~ v      1.49e+0    0.289
## 17 " "       Cruise Ship Di~ <tib~ <nls> <tib~ <tibb~ K      6.74e+4  391.
## 18 " "       Cruise Ship Di~ <tib~ <nls> <tib~ <tibb~ B      2.80e-1    0.0275
## 19 " "       Cruise Ship Di~ <tib~ <nls> <tib~ <tibb~ t0     1.83e+4    1.02
## 20 " "       Cruise Ship Di~ <tib~ <nls> <tib~ <tibb~ v      1.49e+0    0.289
## # ... with 2 more variables: statistic <dbl>, p.value <dbl>
```