Group 25 Case Study

Dilara Ademoğlu - 03722489
Aynur Süne - 03725788
Anna Illarionova - 03678137
Samira John - 03744714

```
Case <- fread('Case.csv')
PatientInfo <- fread('PatientInfo.csv')
Policy <- fread('Policy.csv')
Region <- fread('Region.csv')
SearchTrend <- fread('SearchTrend.csv')
SeoulFloating <- fread('SeoulFloating.csv')
Time <- fread('Time.csv')
TimeAge<- fread('TimeAge.csv')
TimeGender <- fread('TimeGender.csv')
TimeProvince <- fread('TimeProvince.csv')
Weather <- fread('Weather.csv')
```

Work on Case file

```
case_dt1 <- Case %>% select(province, confirmed) #change to df function
case_dt1 <- as.data.table(case_dt1) [, lapply(.SD, sum),
                                     by = .(province = tolower(province))]
case_dt1 <- case_dt1[order(case_dt1$confirmed), ]
case_dt1 <- case_dt1[, proportion := confirmed / sum(confirmed) * 100]
```

ANALYSIS ON PROVINCE

Work on TimeProvince file

```
timeprovince_dt <- TimeProvince %>% group_by(province)

outliers_dt <- TimeProvince[province == "Gyeongsangbuk-do" | province == "Daegu"]
outliers_dt <- outliers_dt %>% select(date, confirmed)
outliers_dt <- outliers_dt %>%
               group_by(date) %>%
               summarize(confirmed = sum(confirmed)) %>%
               ungroup
```

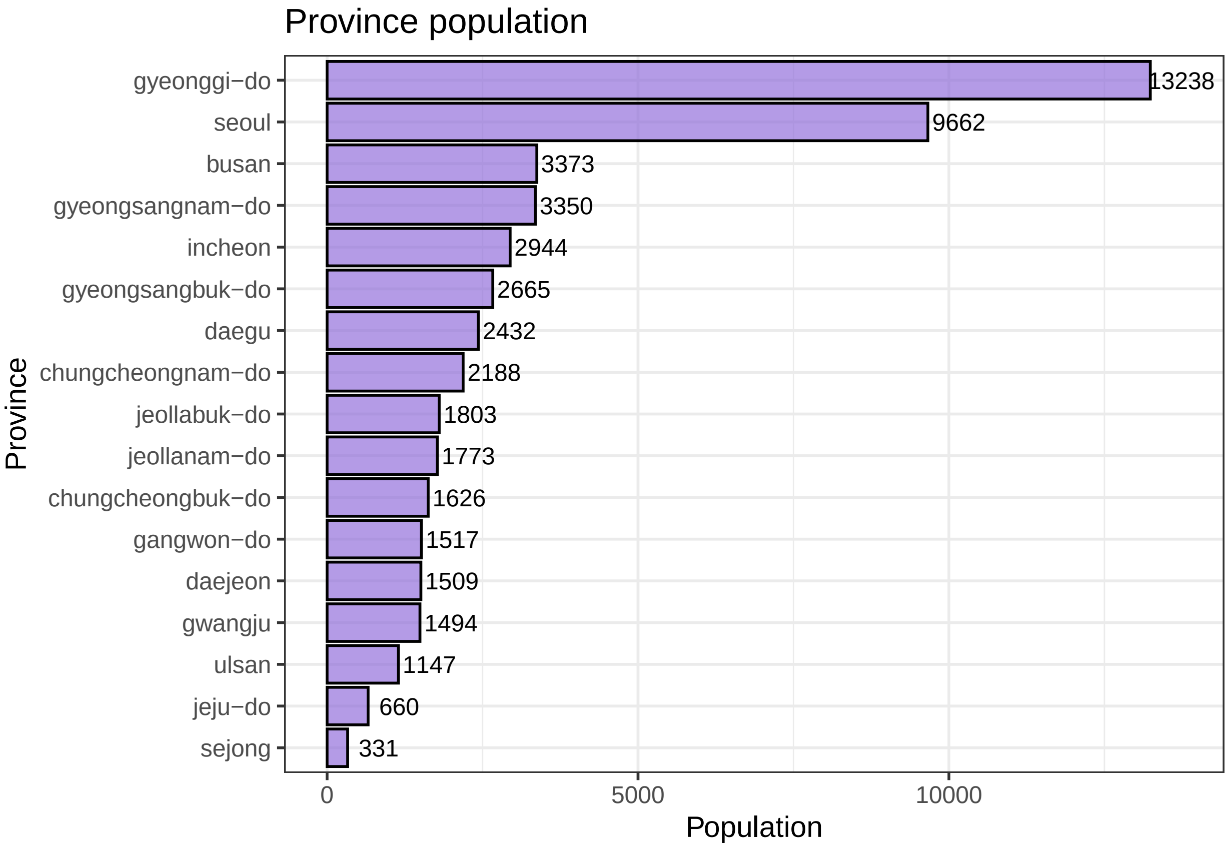
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
names(outliers_dt)[1:2] <- c("date", "confirmed_outliers")
time_dt <- merge(Time, outliers_dt, by = "date", all = FALSE)
time_dt <- time_dt[, no_outliers := confirmed - confirmed_outliers]
```

Plot Nr.1: Population* in provinces

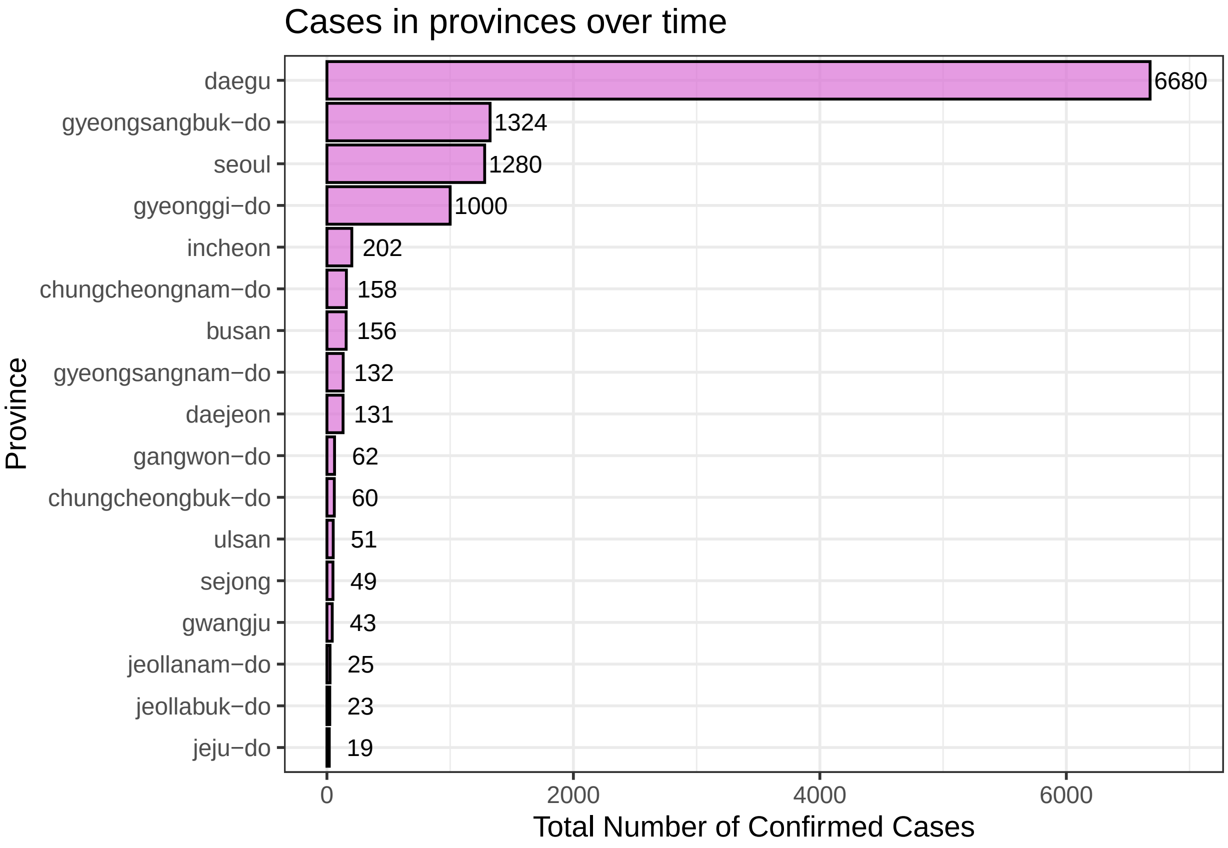
(*Data is obtained from **here**)

```r
case_dt1 <- case_dt1[order(case_dt1$province), ] #sort in alph. order
case_dt1 <- case_dt1[, population := as.integer(c("3373","1626","2188","2432","1509",
                                                  "1517","1494","13238","2665","3350",
                                                  "2944","660","1803","1773","331",
                                                  "9662","1147"))]

case_dt1 <- case_dt1[order(case_dt1$population), ]

plot1 <- ggplot(case_dt1, aes(x = reorder(province, population), y = population)) +
         geom_bar(stat = "identity",
                  color = "black",
                  fill = "mediumpurple",
                  alpha = 0.7) +
         coord_flip() +
         geom_text(aes(label = population, y = population + 500), size = 3) +
         labs(x = 'Province', y = 'Population') +
         ggtitle("Province population") +
         theme_bw()

plot1
```



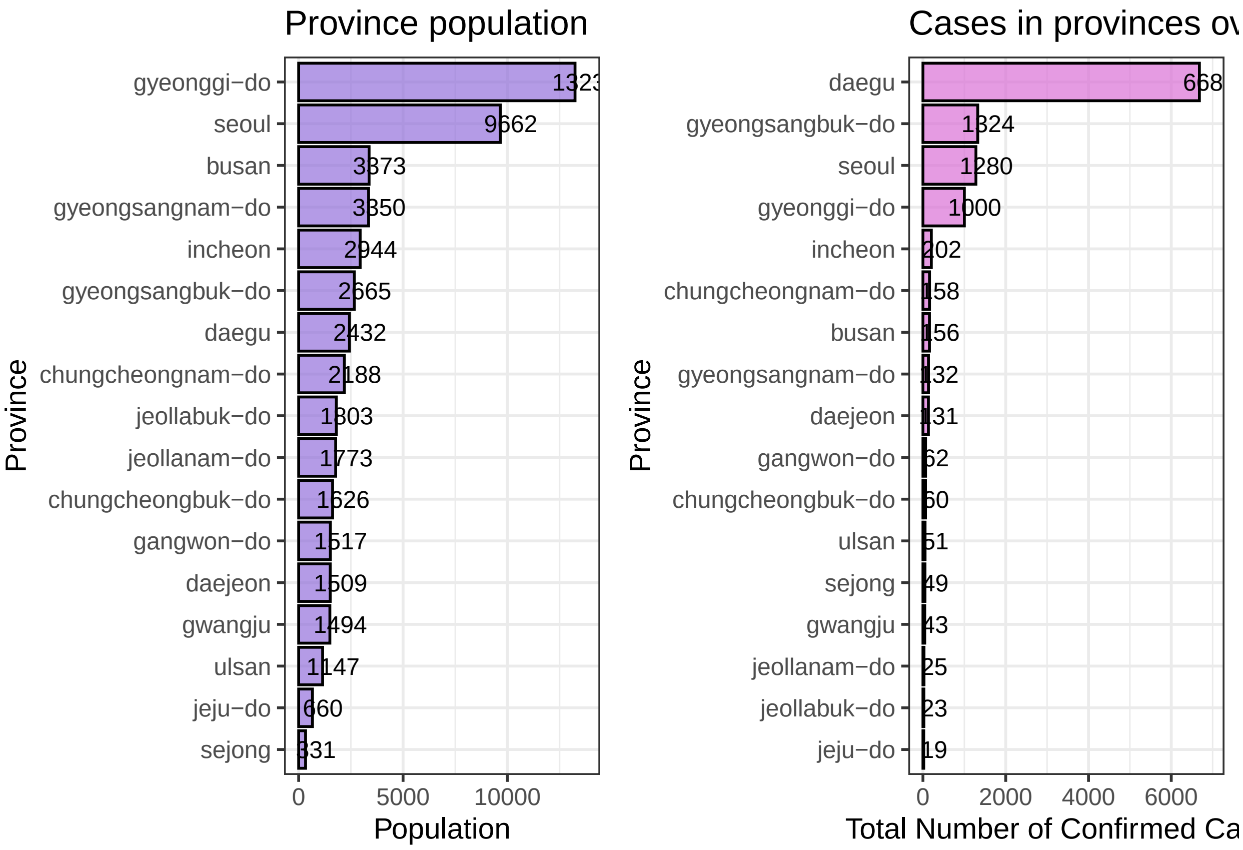
Plot Nr.2: Cases in provinces

```r
plot2 <- ggplot(case_dt1, aes(x=reorder(province, confirmed),y=confirmed)) +
         geom_bar(stat="identity",color="black",fill= "orchid", alpha = 0.7) +
         coord_flip() +
         geom_text(aes(label=confirmed, y = confirmed+250), size=3) +
         labs(x='Province', y='Total Number of Confirmed Cases') +
         ggtitle("Cases in provinces over time") +
         theme_bw()

plot2
```



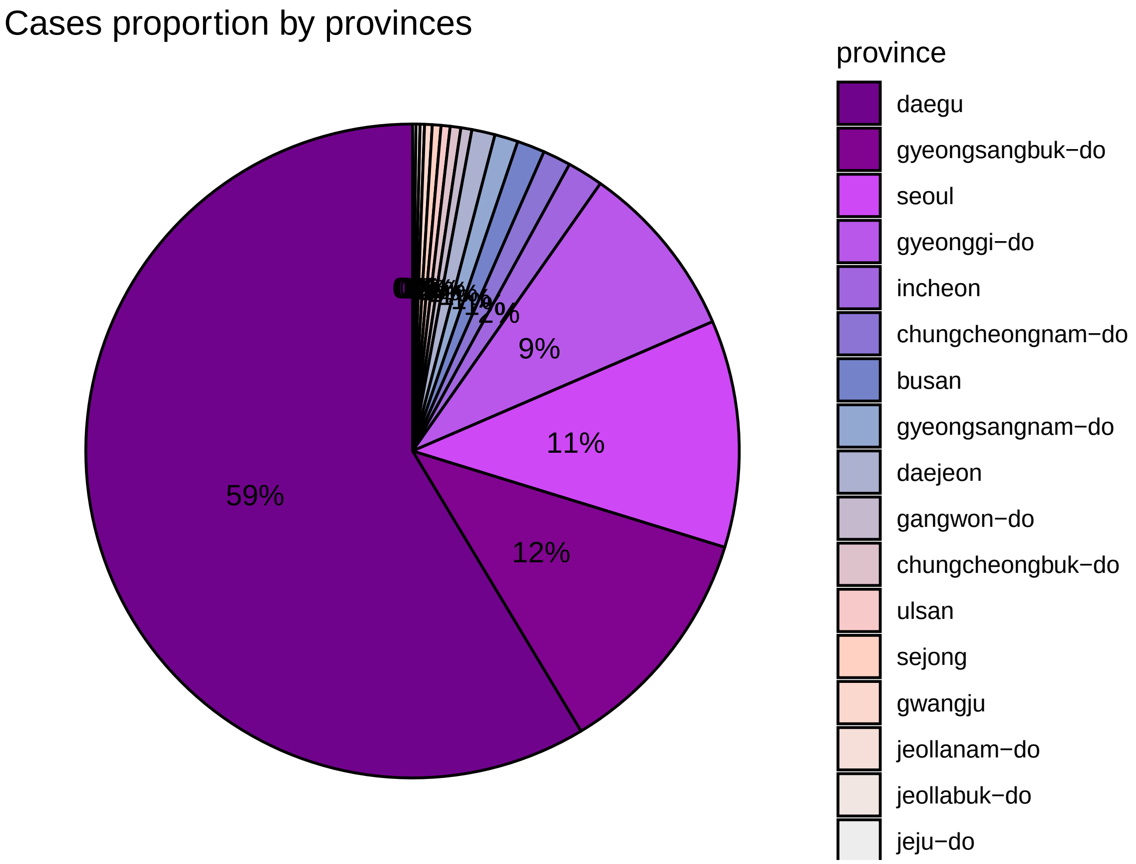
Plot Nr.3: Comparison of population and number of cases

```
require(gridExtra)
grid.arrange(plot1, plot2, ncol=2)
```



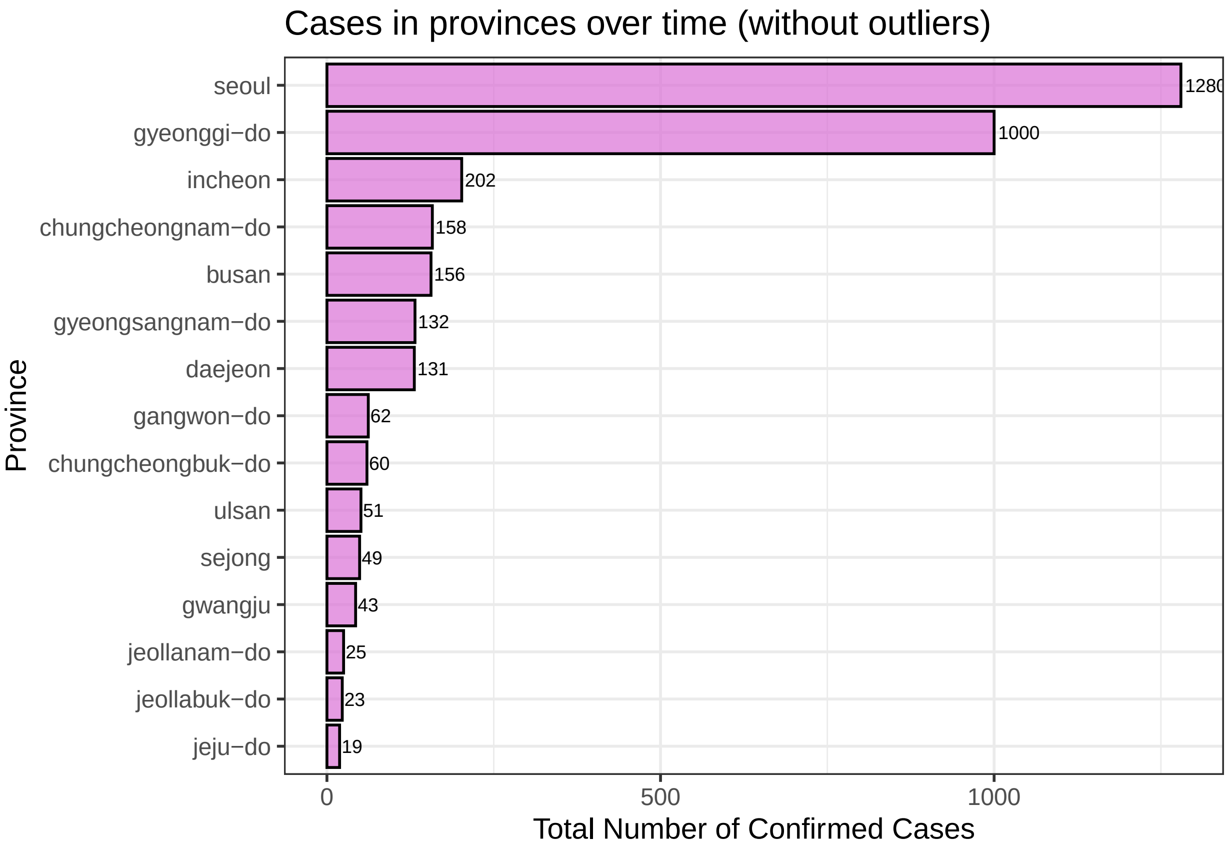
Plot Nr.4: Proportion of total cases by province

```
case_dt1 <- case_dt1[order(case_dt1$proportion),]
case_dt1$province <- factor(case_dt1$province,
                            levels = rev(as.character(case_dt1$province)))
case_dt1_x <- case_dt1 %>%
              arrange(desc(province)) %>%
              mutate(y = proportion / sum(proportion), y.stack = cumsum(y)) %>%
              ungroup()

ggplot(case_dt1_x, aes(x='', label=paste0(round(proportion), "%"), y=y, fill=province))+
  geom_bar(stat="identity", width=1, color="black") +
  coord_polar("y") +
  geom_text(aes(label=paste0(round(proportion),"%")),position=position_stack(vjust=0.5))+
  ggtitle("Cases proportion by provinces")+
  scale_fill_manual(values = rev(c("#ededed", "#f2e6e2", "#f6dfd8", "#fbd8cd","#ffd1c2",
                                   "#f7cac9", "#dec2cb", "#c5b9cd", "#abb1cf","#92a8d1",
                                   "#7483c9","#8b74d4","#a166df","#b857ea","#ce48f5",
                                   "#800490","#70038c"))) +
  theme_void()
```



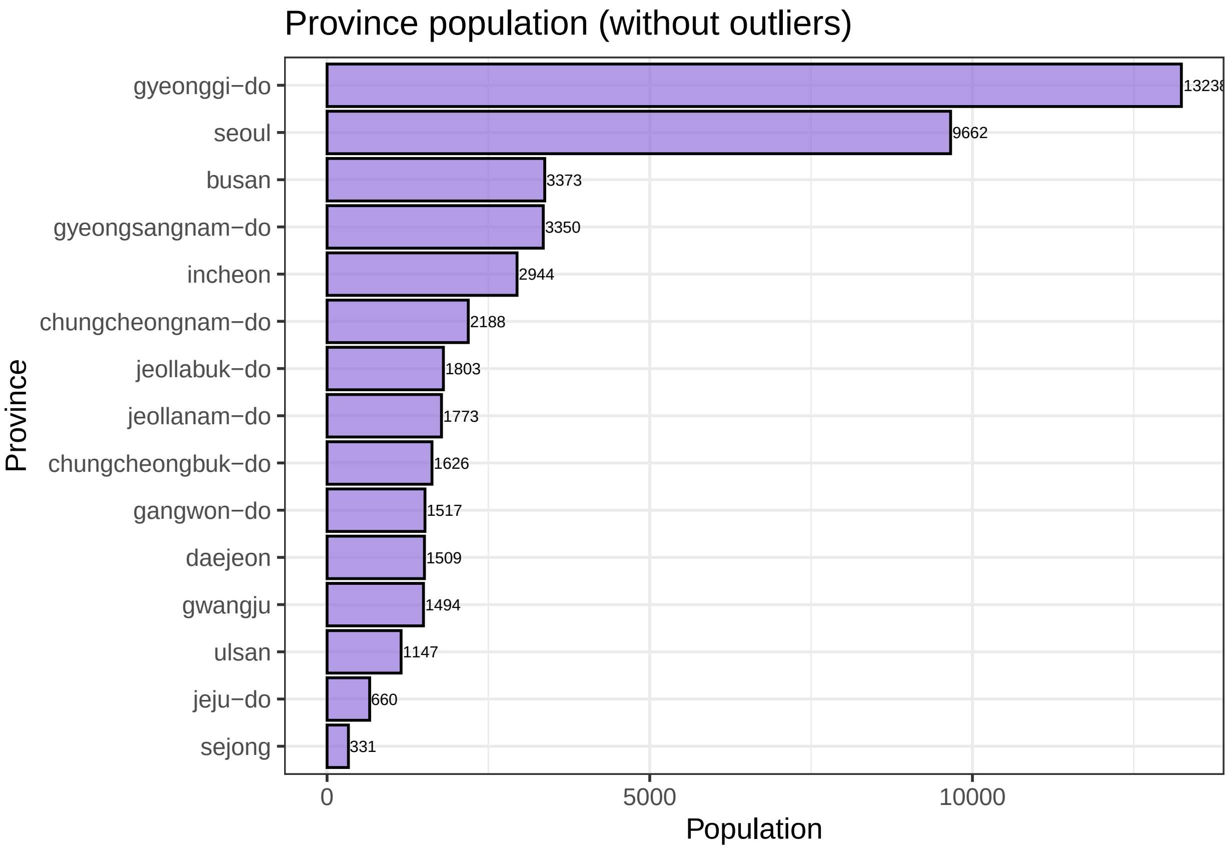
Plot Nr.5: Cases in provinces without 2 outliers

```
case_dt2<-case_dt1[!(case_dt1$province=="daegu" | case_dt1$province=="gyeongsangbuk-do"),]
plot5<-ggplot (case_dt2, aes(x=reorder(province, confirmed),y=confirmed))+
  geom_bar(stat="identity", color="black",fill= "orchid", alpha = 0.7)+
  coord_flip()+
  geom_text(aes(label=confirmed), hjust= - 0.1, size=2.5)+
  labs( x='Province', y='Total Number of Confirmed Cases') +
  ggtitle("Cases in provinces over time (without outliers)")+
  theme_bw()

plot5
```



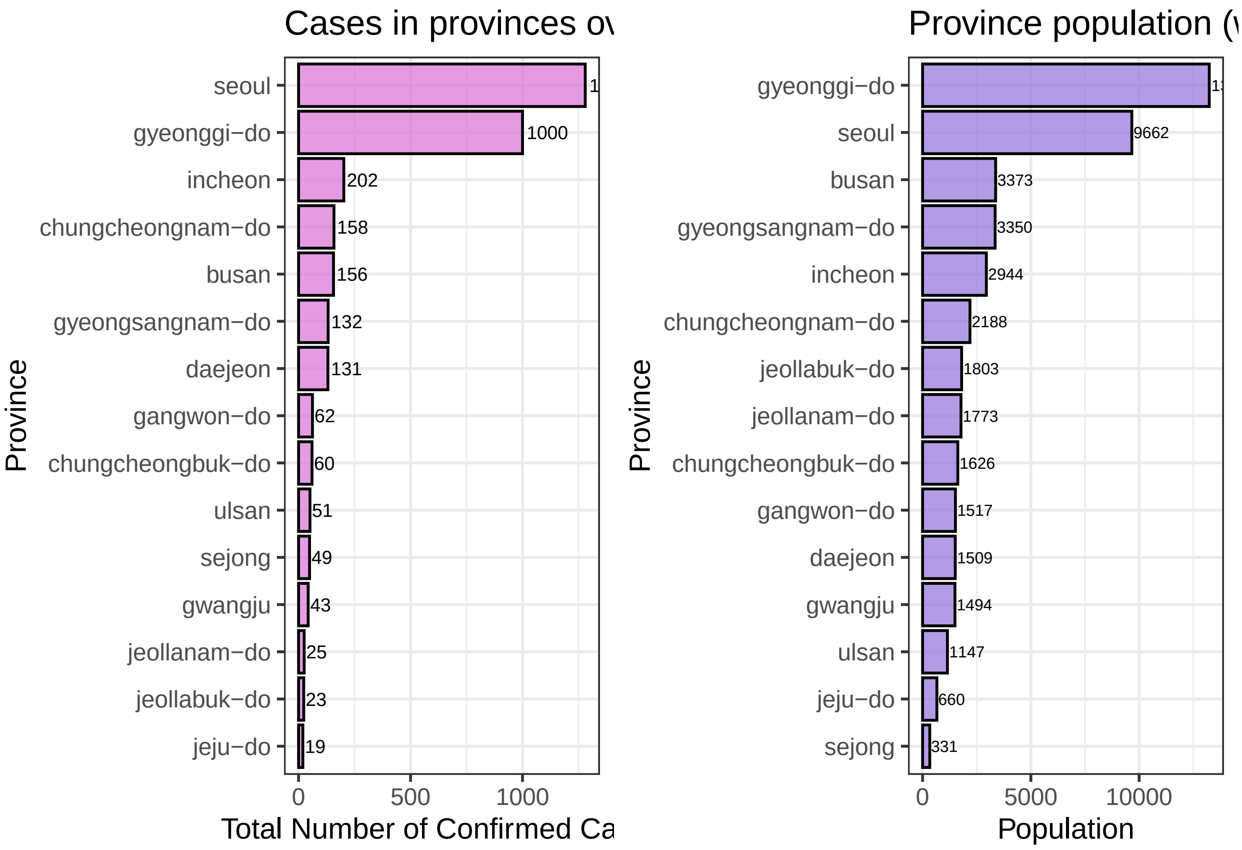
Plot Nr.6: Population in provinces without 2 outliers

```
case_dt1 <- case_dt1[order(case_dt1$province),] #sort in alph. order
case_dt1 <- case_dt1[, population:= as.integer(c("3373","1626","2188",
                                               "2432","1509","1517",
                                               "1494","13238","2665",
                                               "3350","2944","660",
                                               "1803","1773","331",
                                               "9662","1147"))]

plot6 <- ggplot (case_dt2, aes(x=reorder(province, population), y=population))+
  geom_bar(stat="identity", color="black",fill= "mediumpurple", alpha = 0.7)+
  coord_flip()+
  geom_text(aes(label=population), hjust=-0.05, size=2.2)+
  labs( x='Province', y='Population') +
  ggtitle("Province population (without outliers)")+
  theme_bw()

plot6
```



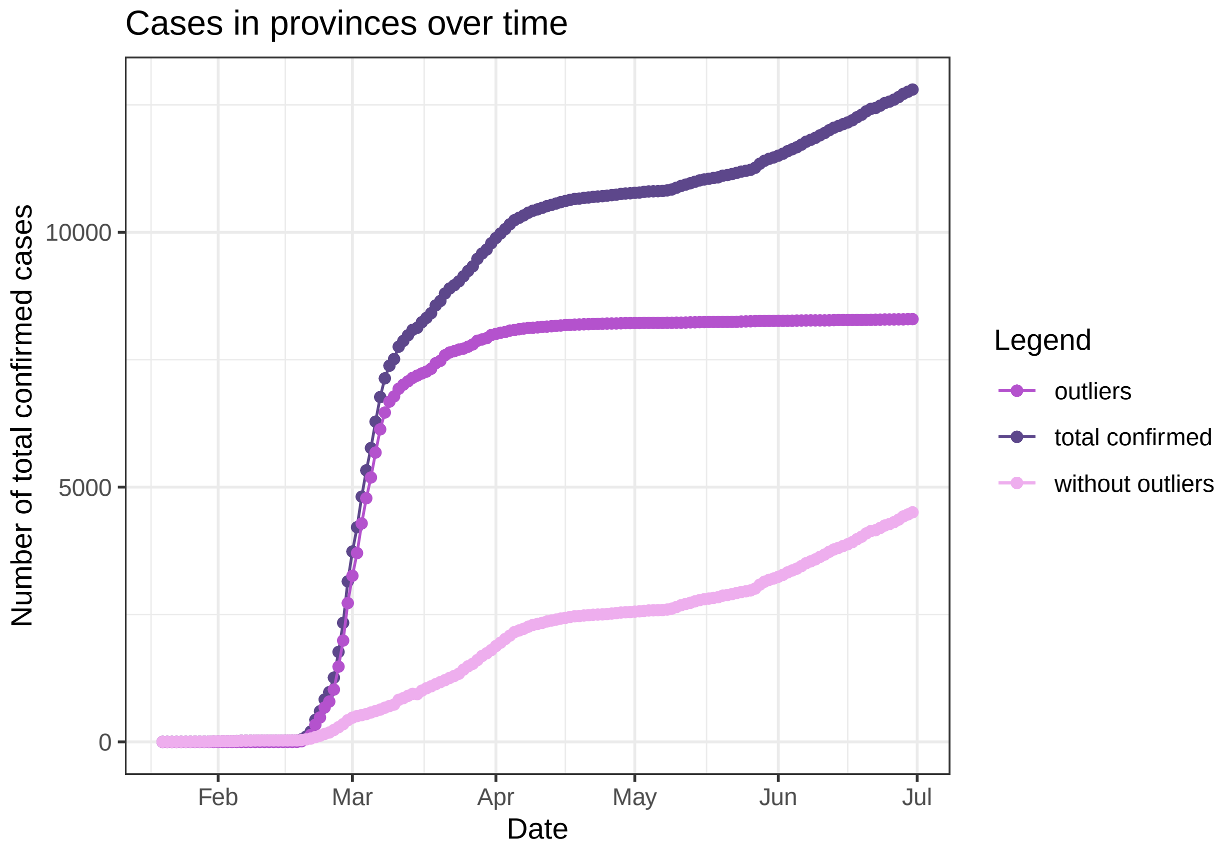
Plot Nr.7: Comparison of Plot Nr.1 and Plot Nr.4

```
require(gridExtra)
grid.arrange(plot5, plot6, ncol=2)
```



Plot Nr.8: General overview

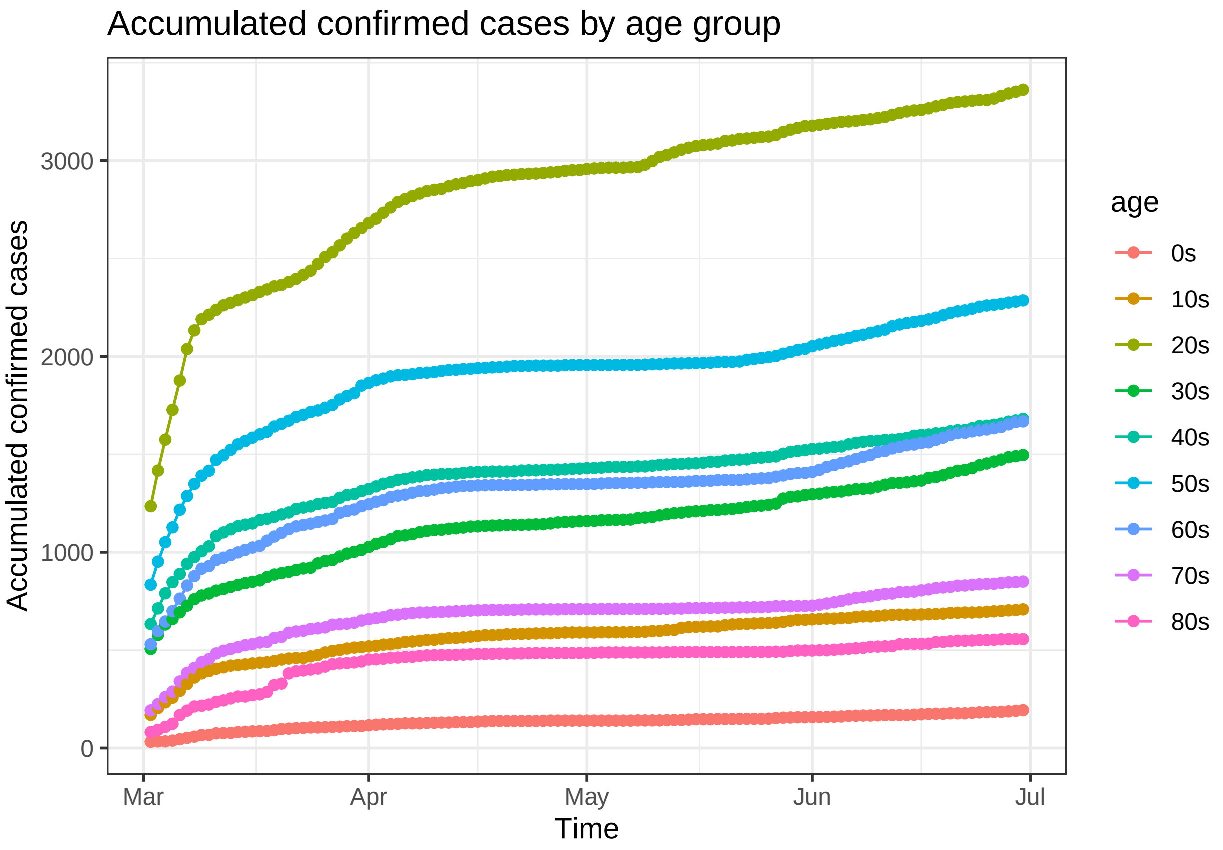
```
colors <- c("total confirmed" = "mediumpurple4",
            "outliers" = "mediumorchid3",
            "without outliers" = "plum2")

ggplot(time_dt, aes(x=date)) +
  geom_line(aes(y=confirmed, color="total confirmed")) +
  geom_point(aes(y=confirmed, color="total confirmed")) +
  geom_line(aes(y=confirmed_outliers, color="outliers")) +
  geom_point(aes(y=confirmed_outliers, color="outliers")) +
  geom_line(aes(y=no_outliers, color="without outliers")) +
  geom_point(aes(y=no_outliers, color="without outliers"))+
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5, hjust=1)) +
  labs(x='Date', y='Number of total confirmed cases', color="Legend") +
  ggtitle("Cases in provinces over time")+
  scale_color_manual(values = colors)+
  theme_bw()
```



ANALYSIS ON GENDER AND AGE

Plot Nr.9: Accumulated confirmed cases by age group

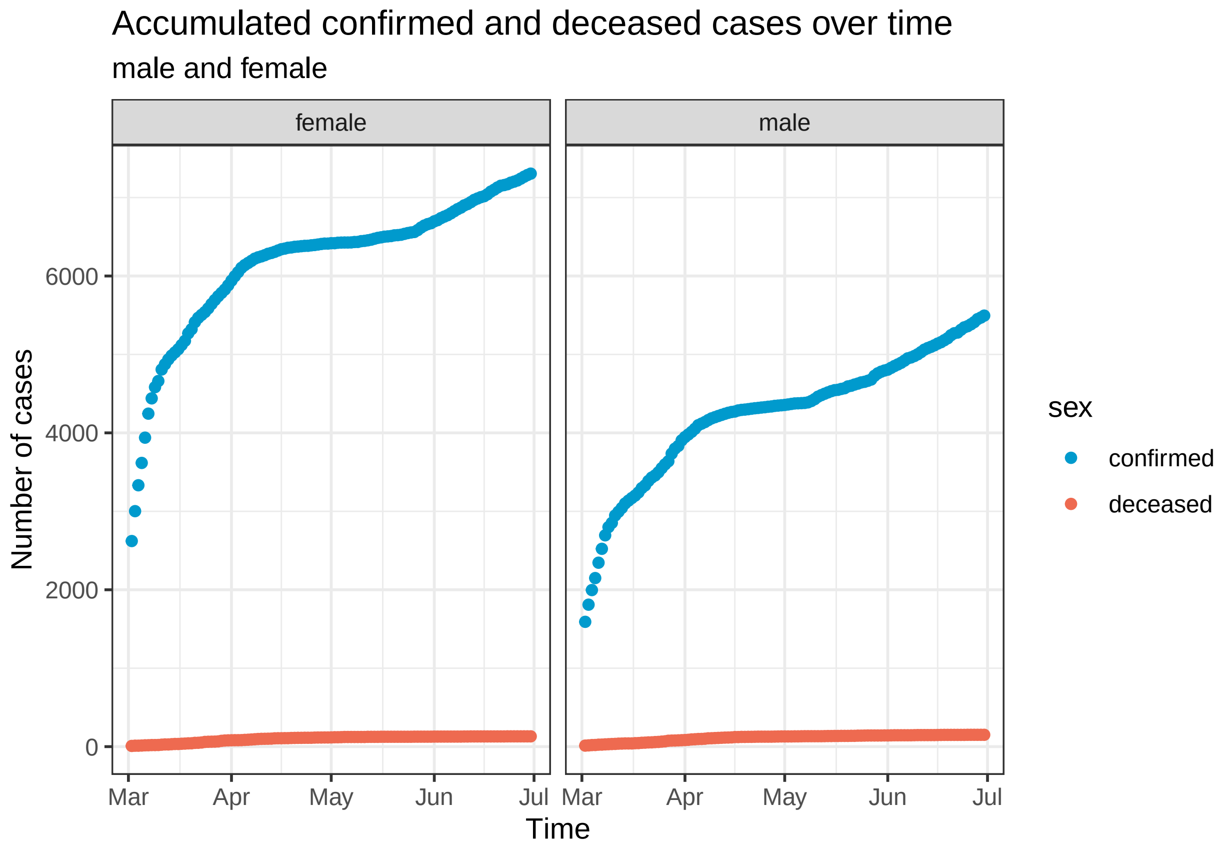
```
TimeAge[, date := as.Date(date, "%Y-%m-%d")]
ggplot(TimeAge, aes(
  x = date,
  y = confirmed,
  color = age,
  group = age
)) +
  geom_line() + geom_point() +
  ggtitle("Accumulated confirmed cases by age group") +
  theme(axis.text.x = element_text(angle = 30, hjust = 1)) +
  labs(x = "Time", y = "Accumulated confirmed cases") + theme_bw()
```



```
TimeGender[, date := as.Date(date, "%Y-%m-%d")]
gender_reshape <- melt(TimeGender,
                       id.vars = c("date", "sex", "time"),
                       measure.vars = c("confirmed", "deceased"))
female_time <- gender_reshape[sex == "female"]
male_time <- gender_reshape[sex == "male"]
```

Plot Nr.10: Accumulated confirmed and deceased cases over time

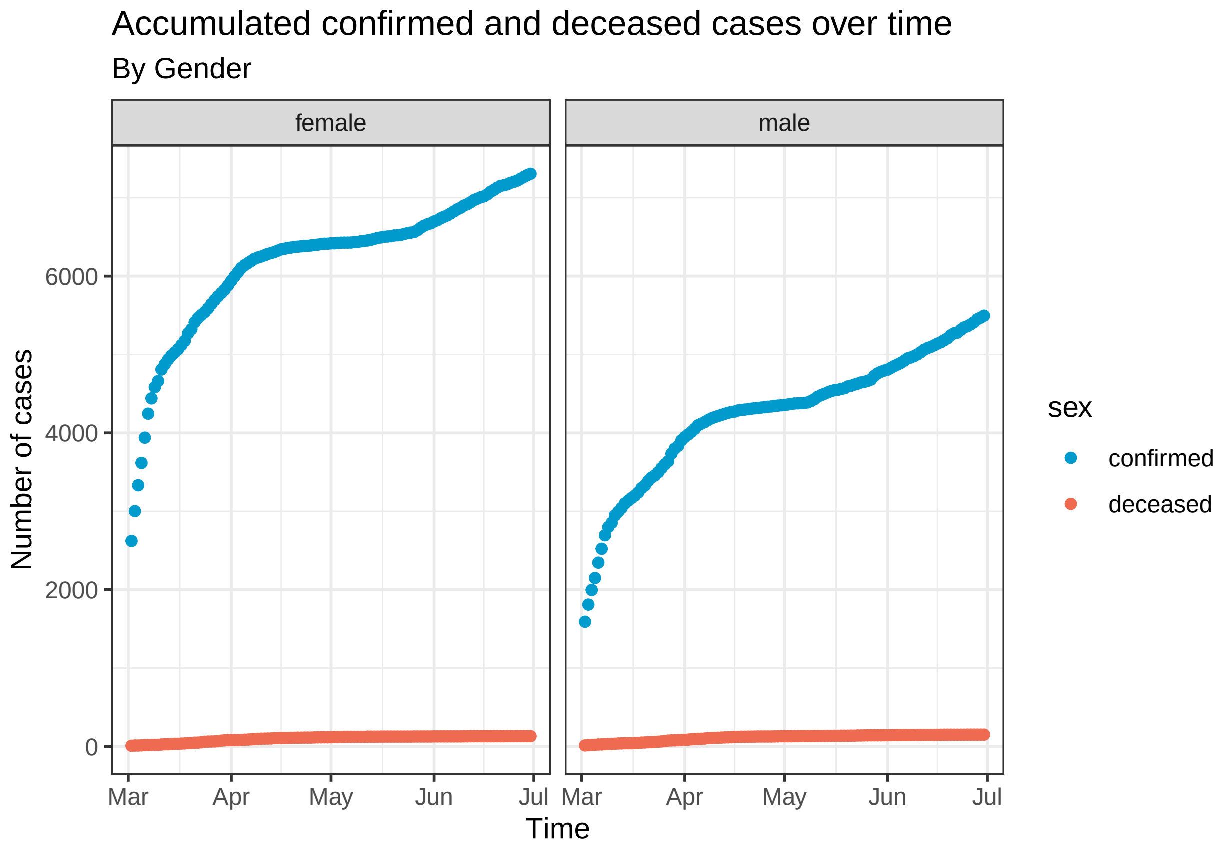
```
ggplot(gender_reshape, aes(x = date, y = value, col = variable)) +
  geom_point() +
  labs(x = "Time", y = "Number of cases",
       title = "Accumulated confirmed and deceased cases over time",
       subtitle = "male and female") +
  scale_color_manual("sex", values = c("deceased" = "coral2",
                                       "confirmed" = "deepskyblue3")) +
  facet_wrap( ~ sex) +
  theme_bw()
```



Plot Nr.11: Accumulated confirmed and deceased cases over time

By gender

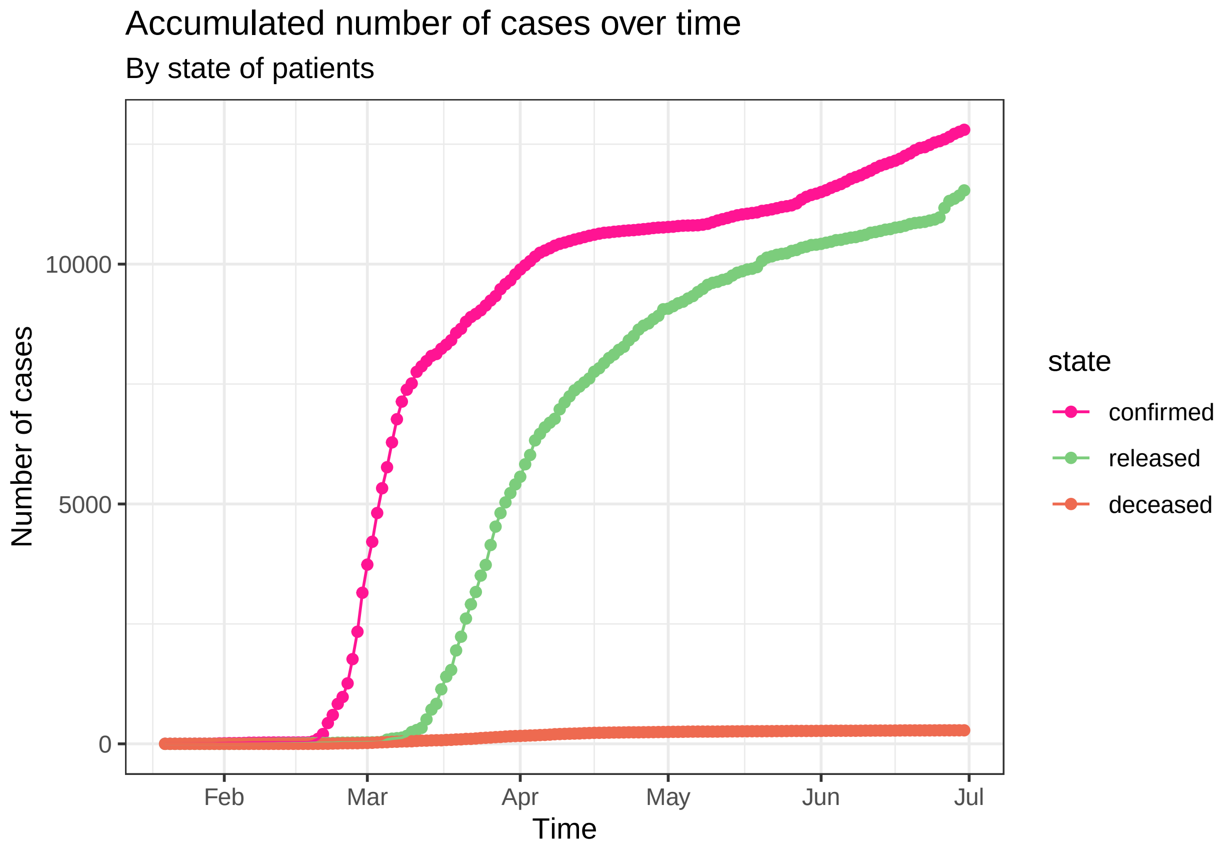
```
ggplot(gender_reshape, aes(x = date, y = value, col = variable)) + geom_point() +
  labs(
    x = "Time",
    y = "Number of cases",
    title = "Accumulated confirmed and deceased cases over time",
    subtitle = "By Gender"
  ) +
  scale_color_manual("sex",
                     values = c("deceased" = "coral2", "confirmed" = "deepskyblue3")) +
  facet_wrap(~ sex) + theme_bw()
```



Plot Nr.12: Accumulated number of cases over time

By state of patients

```
# reshape time data
time_reshape <- melt(Time,
                     id.vars = "date",
                     measure.vars = c("confirmed", "released", "deceased"))

ggplot(time_reshape, aes(x = date, y = value, col = variable)) +
  geom_point() + geom_line() +
  labs(x = "Time",
       y = "Number of cases",
       title = "Accumulated number of cases over time",
       subtitle = "By state of patients") +
  scale_color_manual("state",
                     values = c("deceased" = "coral2",
                                "confirmed" = "deeppink",
                                "released" = "palegreen3")) +
  theme_bw()
```



Working on PatientInfo

```
PatientInfo[sex == "female", .N]
```

```
## [1] 2218
```

```
PatientInfo[sex == "male", .N]
```

```
## [1] 1825
```

```
PatientInfo$age <- factor(PatientInfo$age,
                          levels = c("0s","10s","20s","30s",
                                     "40s","50s","60s","70s",
                                     "80s", "90s","100s"))

patient_state_sex <- PatientInfo[sex != "", .(sex, age, confirmed_date, state)]
patient_state_sex[, state := as.factor(state)]
patient_state_sex[, sex := as.factor(sex)]
sapply(patient_state_sex, class)
```

```
## $sex
## [1] "factor"
##
## $age
## [1] "factor"
##
## $confirmed_date
## [1] "IDate" "Date"
```

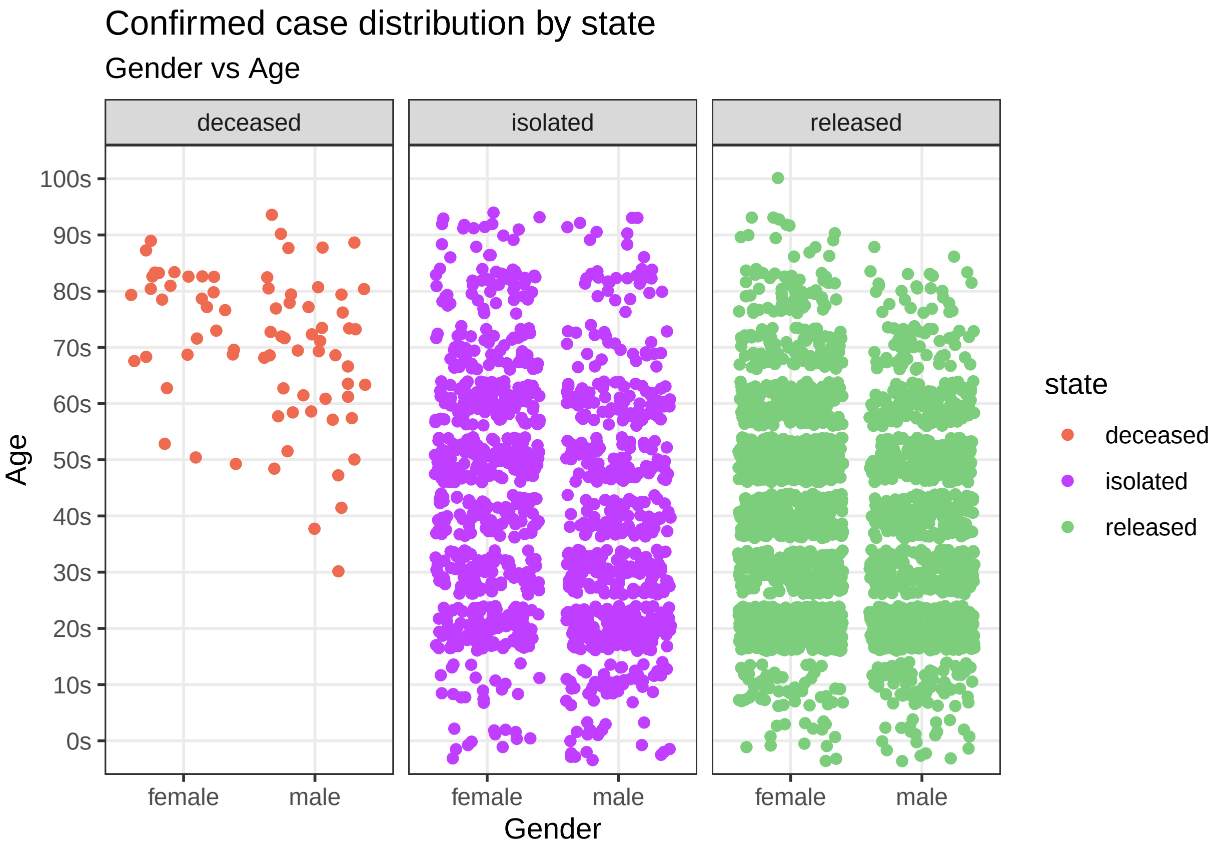
```
##
## $state
## [1] "factor"
```

```
dead_patients_bysex <- patient_state_sex[state == "deceased"]
```

Plot Nr.13: Confirmed case distribution by state

Gender vs Age

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  ggplot(., aes(x = sex, y = age, col = state)) +
  geom_jitter() +
  labs(
    title = "Confirmed case distribution by state",
    subtitle = "Gender vs Age",
    y = "Age",
    x = "Gender"
  ) + facet_wrap( ~ state) +
  scale_color_manual(
    "state",
    values = c(
      "deceased" = "coral2",
      "isolated" = "darkorchid1",
      "released" = "palegreen3"
    )
  ) +
  theme_bw()
```

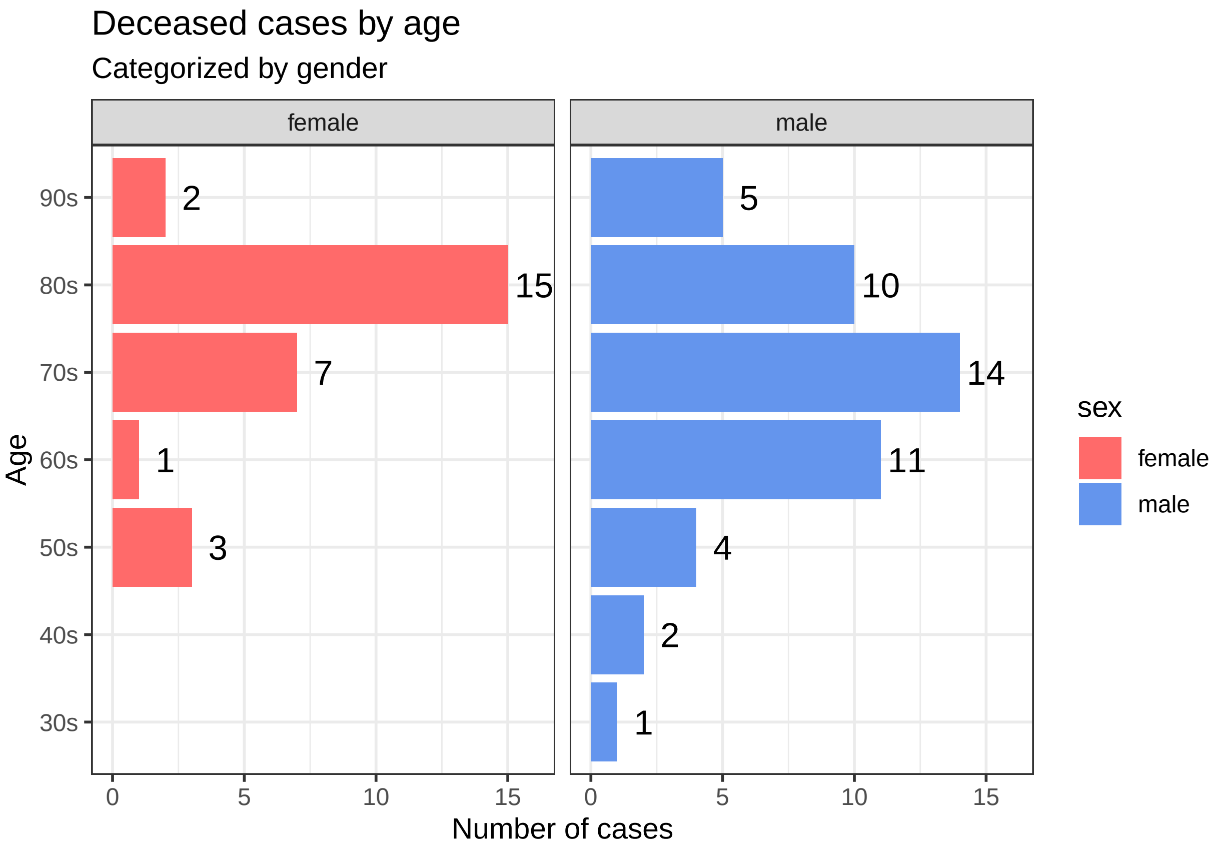


Plot Nr.14: Deceased cases by age

Categorized by gender

```
dead_patients_bysex %>%
  group_by(sex, age) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = age, y = count, fill = sex)) + geom_bar(stat = "identity") +
  geom_text(aes(label = count, y = count + 1), size = 4.5) +
  labs(
    x = "Age",
    y = "Number of cases",
    title = "Deceased cases by age",
    subtitle = "Categorized by gender"
  ) +
  scale_fill_manual("sex",
                    values = c("female" = "indianred1", "male" = "cornflowerblue")) +
  facet_wrap( ~ sex) + coord_flip() + theme_bw()
```

```
## `summarise()` regrouping output by 'sex' (override with `.groups` argument)
```

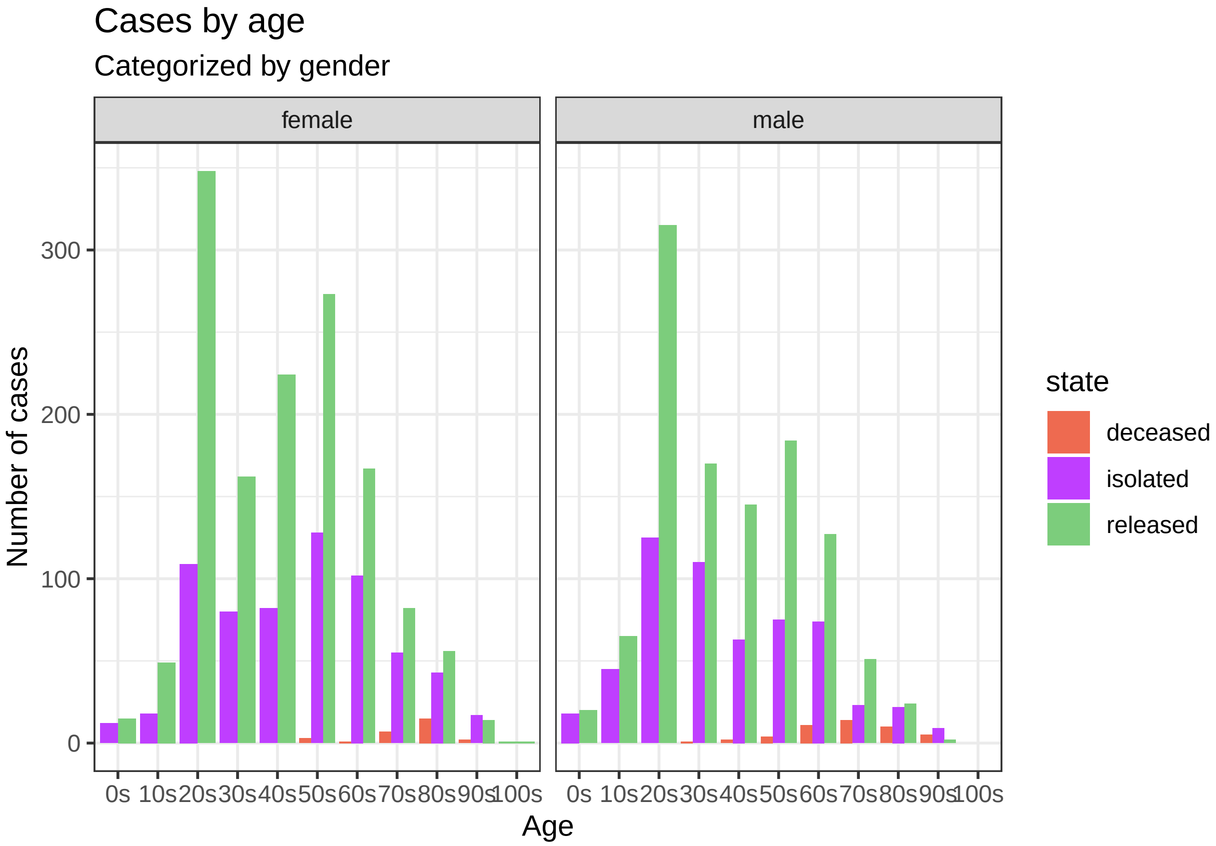


Plot Nr.15: Cases by age

Categorized by gender

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  group_by(sex, age, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = age, y = count, fill = state)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    x = "Age",
    y = "Number of cases",
    title = "Cases by age",
    subtitle = "Categorized by gender"
  ) +
  scale_fill_manual(
    "state",
    values = c(
      "deceased" = "coral2",
      "isolated" = "darkorchid1",
      "released" = "palegreen3"
    )
  )  +
  facet_wrap( ~ sex) + theme_bw()
```

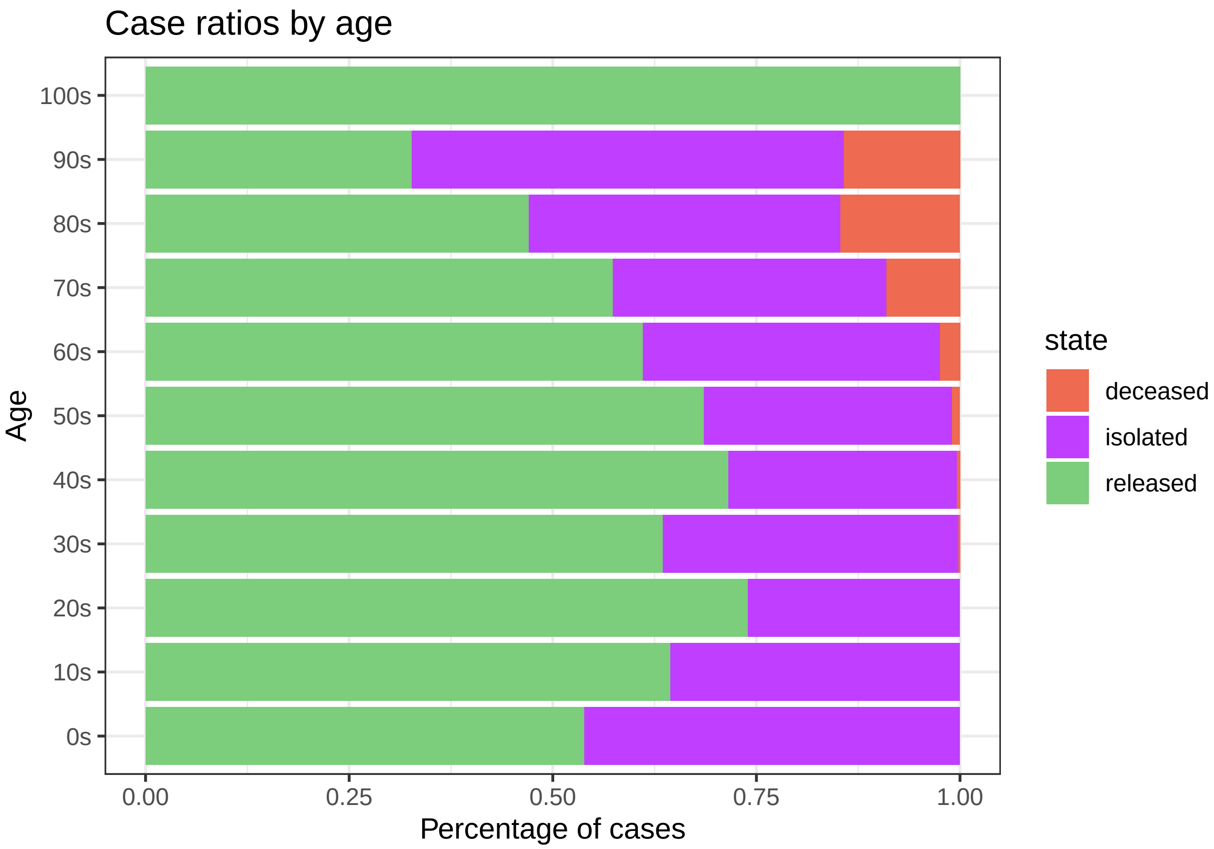
```
## `summarise()` regrouping output by 'sex', 'age' (override with `.groups` argument)
```



Plot Nr.16: Case ratios by age

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  group_by(sex, age, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = age, y = count, fill = state)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(x = "Age", y = "Percentage of cases", title = "Case ratios by age") +
  scale_fill_manual(
    "state",
    values = c(
      "deceased" = "coral2",
      "isolated" = "darkorchid1",
      "released" = "palegreen3"
    )
  )  + coord_flip() + theme_bw()
```

```
## `summarise()` regrouping output by 'sex', 'age' (override with `.groups` argument)
```

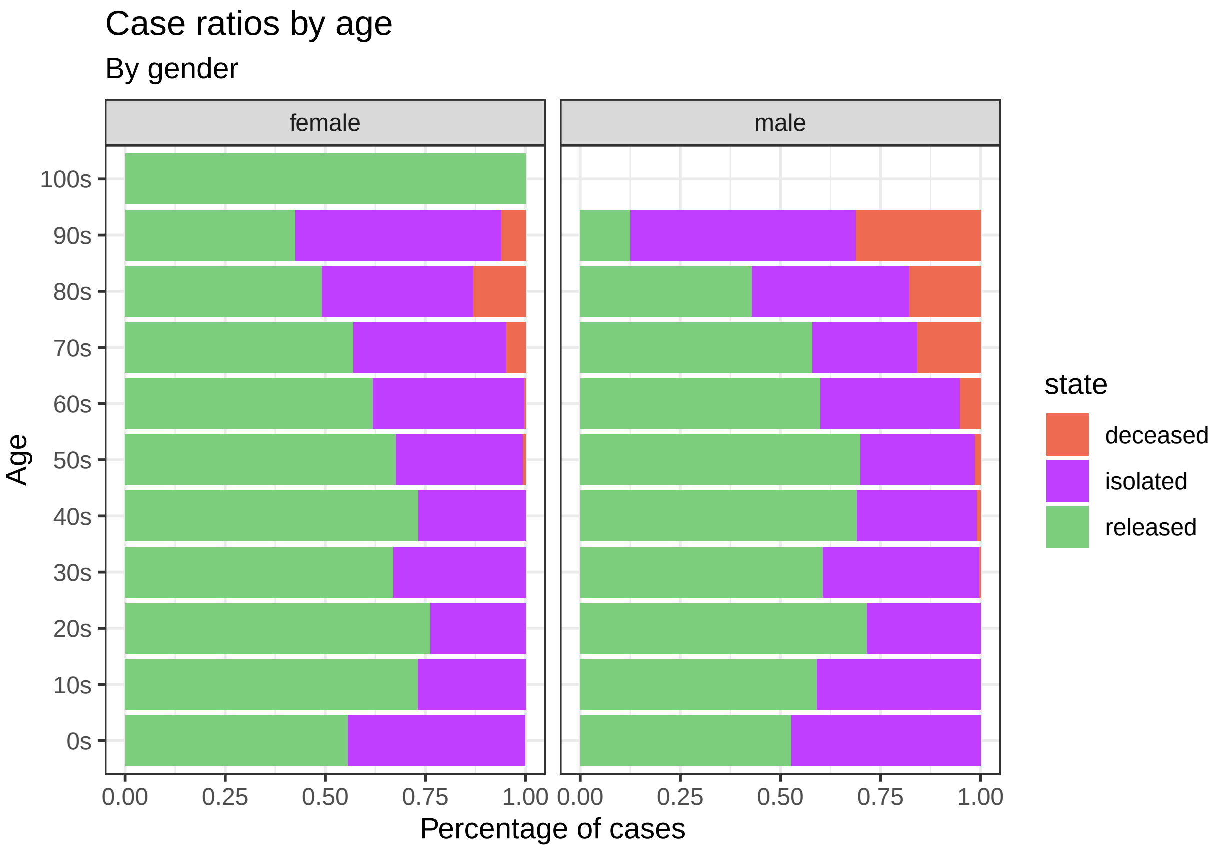


Plot Nr.17: Case ratios by age

By gender

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  group_by(sex, age, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = age, y = count, fill = state)) +
    geom_bar(stat = "identity", position = "fill") +
    labs(x = "Age",
         y = "Percentage of cases",
         title = "Case ratios by age",
         subtitle = "By gender") +
    scale_fill_manual("state", values = c("deceased" = "coral2",
                                          "isolated" = "darkorchid1",
                                          "released" = "palegreen3")) +
    facet_wrap( ~ sex) +
    coord_flip() +
  theme_bw()
```

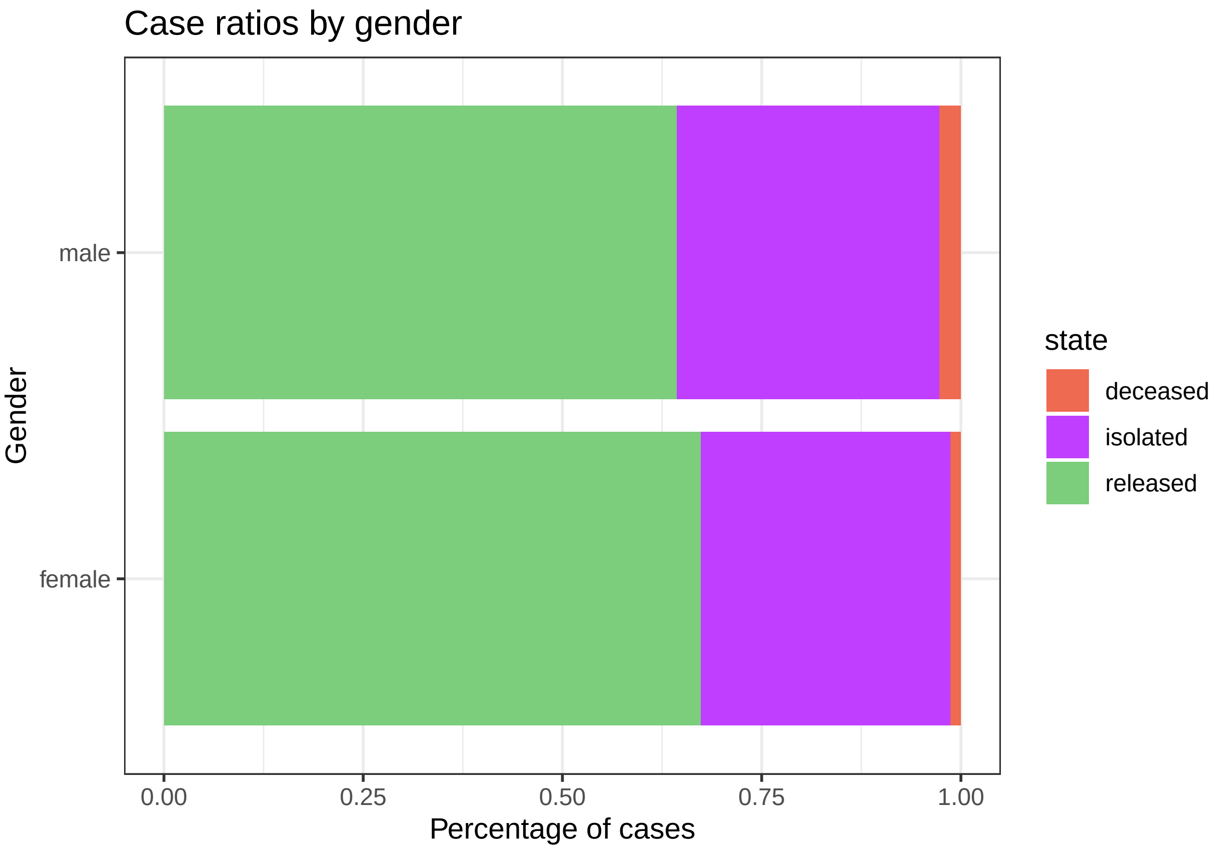
```
## `summarise()` regrouping output by 'sex', 'age' (override with `.groups` argument)
```



Plot Nr.18: Case ratios by gender

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  group_by(sex, age, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = sex, y = count, fill = state)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(x = "Gender", y = "Percentage of cases", title = "Case ratios by gender") +
  scale_fill_manual("state", values = c("deceased" = "coral2",
                                        "isolated" = "darkorchid1",
                                        "released" = "palegreen3")) +
   coord_flip() +
   theme_bw()
```

```
## `summarise()` regrouping output by 'sex', 'age' (override with `.groups` argument)
```

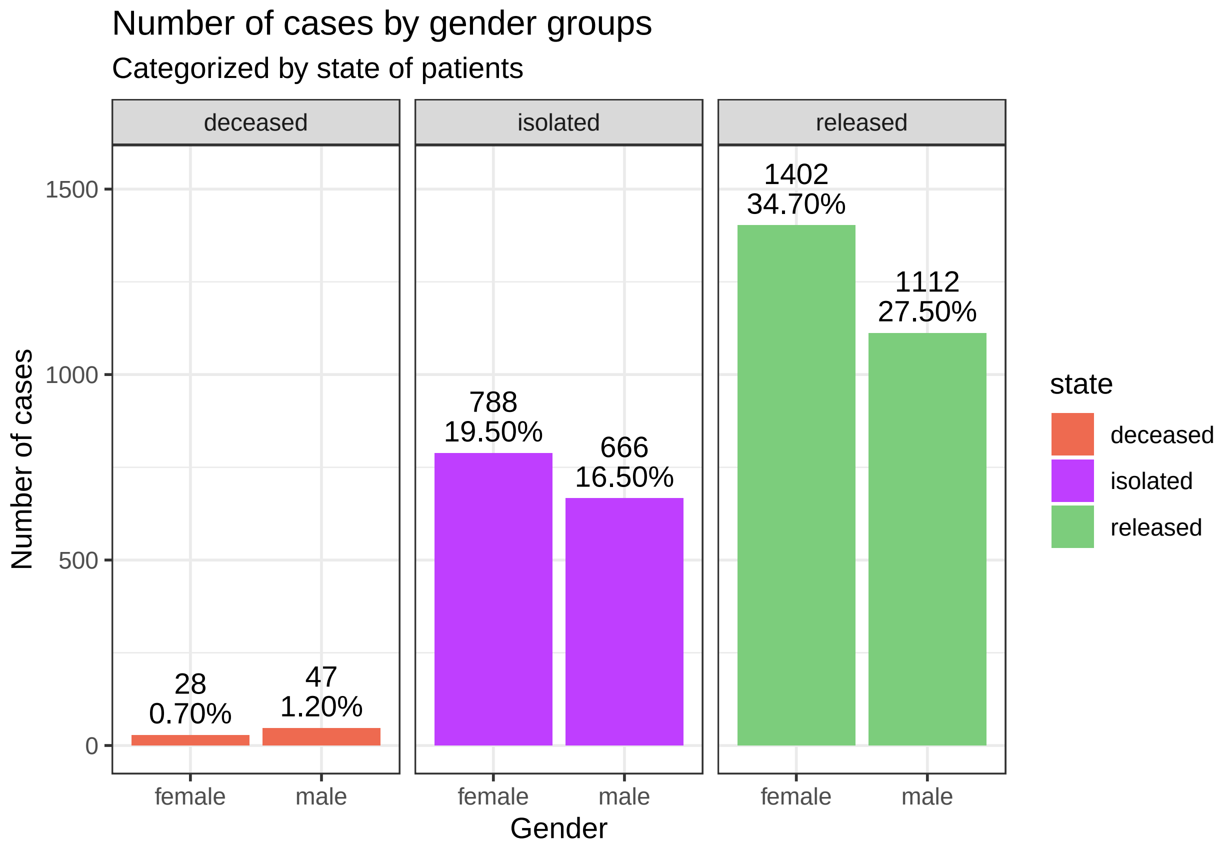


Plot Nr.19: Number of cases by gender groups

Categorized by state of patients

```
patient_state_sex %>%
  group_by(sex, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = sex, y = count, fill = state)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = scales::percent(round((count) / sum(count), 3)),
              y = count + 60), size = 4) +
  geom_text(aes(label = count, y = count + 140), size = 4) +
  labs(x = "Gender", y = "Number of cases",
       title = "Number of cases by gender groups",
       subtitle = "Categorized by state of patients") +
  facet_wrap( ~ state) +
  scale_fill_manual("state", values = c("deceased" = "coral2",
                                        "isolated" = "darkorchid1",
                                        "released" = "palegreen3")) +
  theme_bw()
```

```
## `summarise()` regrouping output by 'sex' (override with `.groups` argument)
```

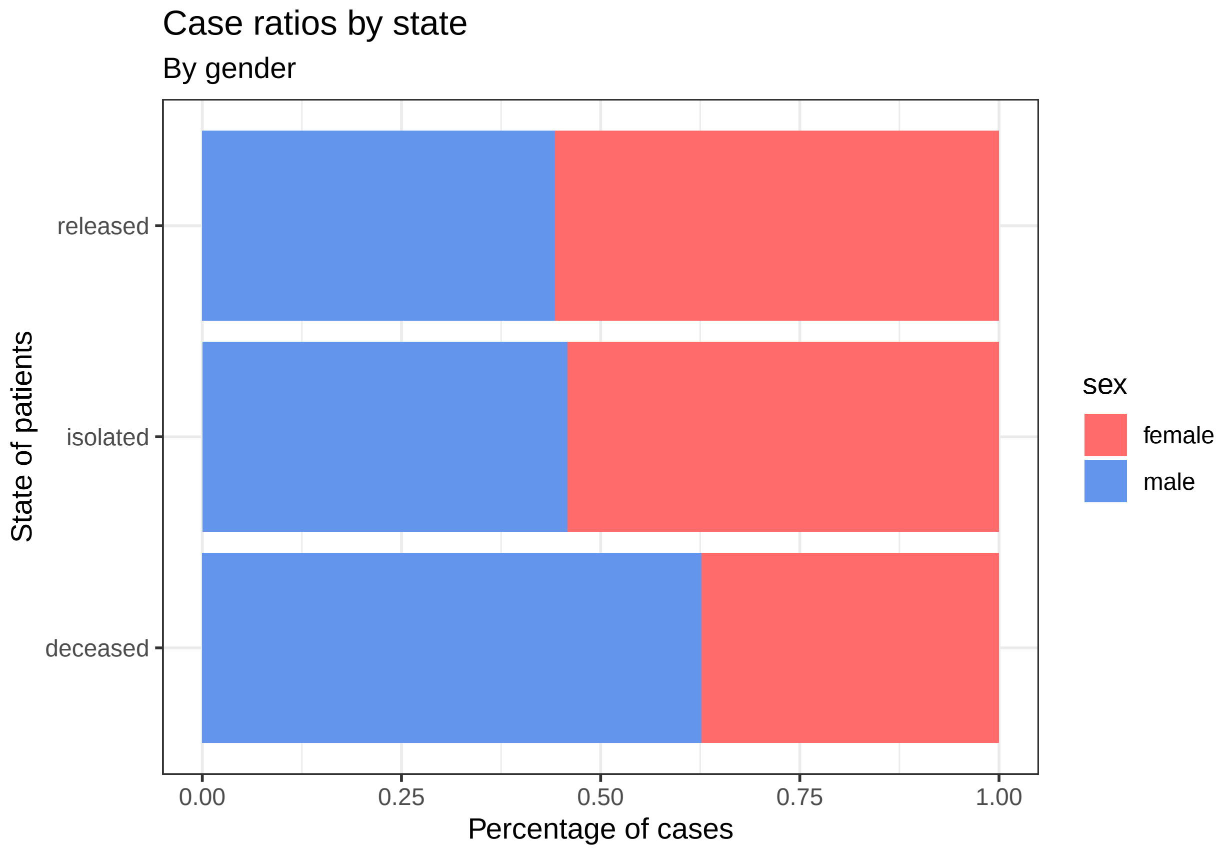


Plot Nr.20: Case ratios by state

By gender

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  group_by(sex, state) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = state, y = count, fill = sex)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(x = "State of patients", y = "Percentage of cases",
       title = "Case ratios by state",
       subtitle = "By gender") +
  scale_fill_manual("sex", values = c("female" = "indianred1",
                                      "male" = "cornflowerblue")) +
  coord_flip() +
  theme_bw()
```

```
## `summarise()` regrouping output by 'sex' (override with `.groups` argument)
```

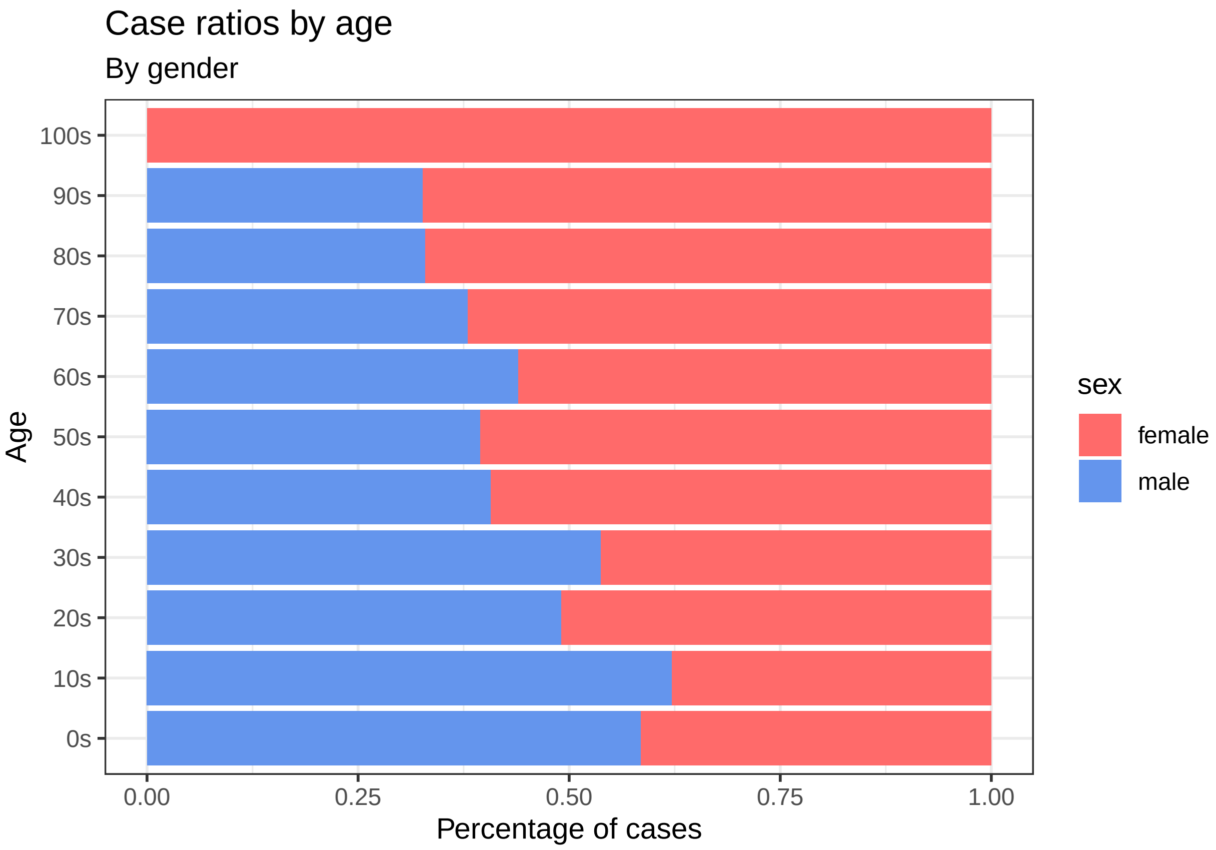


Plot Nr.21: Case ratios by sge

By gender

```
patient_state_sex %>%
  filter(is.na(confirmed_date) == F) %>%
  filter(is.na(age) == F) %>%
  group_by(sex, state, age) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = age, y = count, fill = sex)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(x = "Age", y = "Percentage of cases",
       title = "Case ratios by age",
       subtitle = "By gender") +
  scale_fill_manual("sex", values = c("female" = "indianred1",
                                      "male" = "cornflowerblue")) +
  coord_flip() +
  theme_bw()
```

```
## `summarise()` regrouping output by 'sex', 'state' (override with `.groups` argument)
```

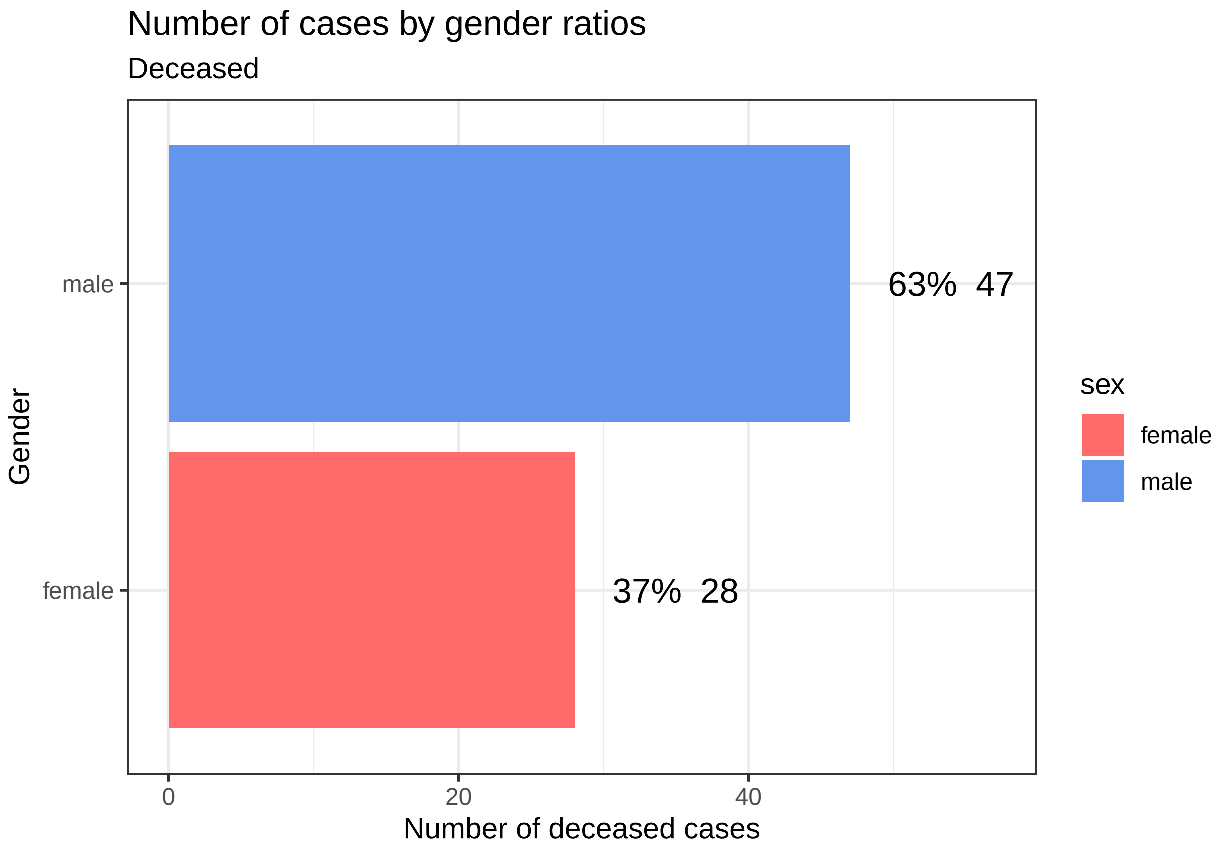


Plot Nr.21: Number of cases by gender ratios

Deceased

```
dead_patients_bysex %>%
  group_by(sex) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = sex, y = count, fill = sex)) +
  geom_bar(stat = "identity") +
  labs(x = "Gender", y = "Number of deceased cases",
       title = "Number of cases by gender ratios",
       subtitle = "Deceased") +
  geom_text(aes(label = scales::percent(round((count) / sum(count), 3)),
                y = count + 5), size = 4.5) +
  geom_text(aes(label = count, y = count + 10), size = 4.5) +
  coord_flip() +
  scale_fill_manual("sex", values = c("female" = "indianred1",
                                      "male" = "cornflowerblue")) +
  theme_bw()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

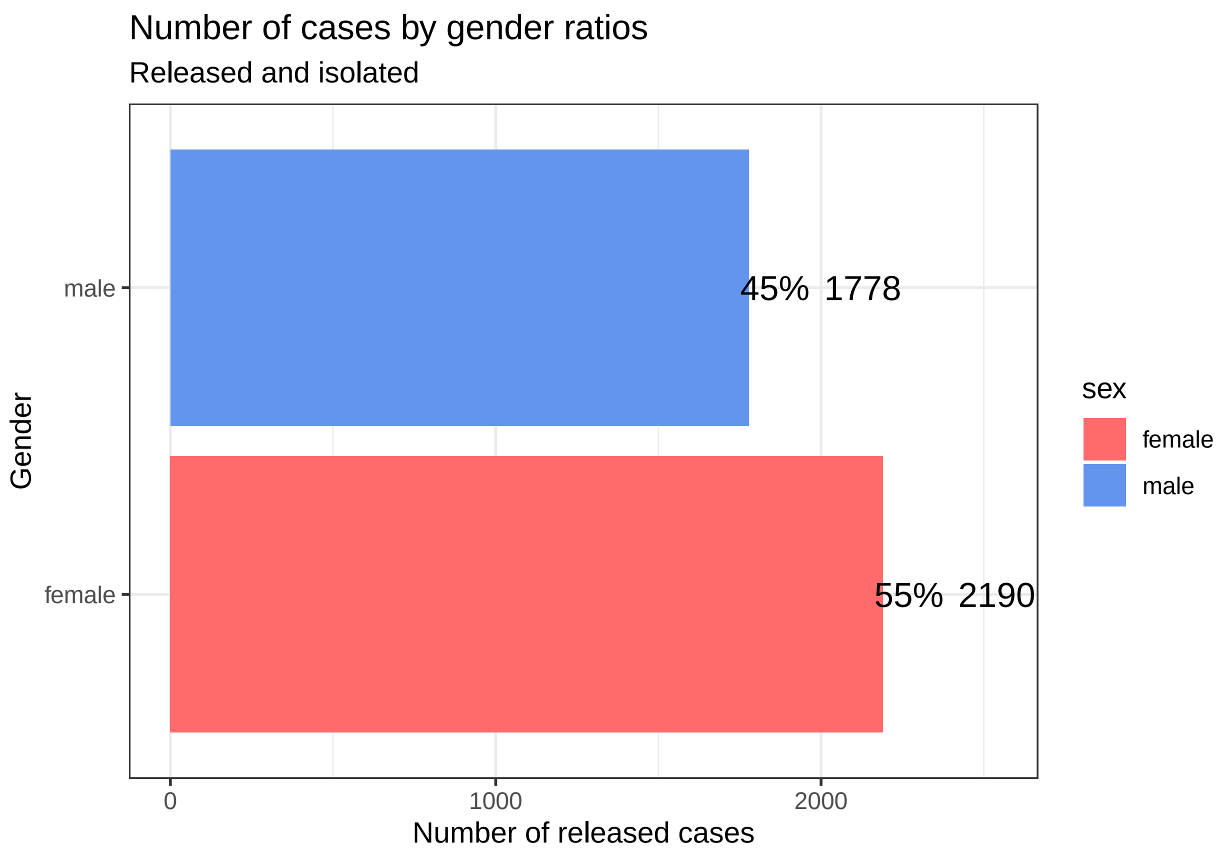


Plot Nr.22: Number of cases by gender ratios

Released and isolated

```
patient_state_sex %>%
  filter(state != "deceased") %>%
  group_by(sex) %>%
  summarise(count = n()) %>%
  ggplot(., aes(x = sex, y = count, fill = sex)) +
  geom_bar(stat = "identity") +
  labs(x = "Gender", y = "Number of released cases",
       title = "Number of cases by gender ratios",
       subtitle = "Released and isolated") +
  geom_text(aes(label = scales::percent(round((count) / sum(count), 3)),
                y = count + 80), size = 4.5) +
  geom_text(aes(label = count, y = count + 350), size = 4.5) +
  coord_flip() +
  scale_fill_manual("sex", values = c("female" = "indianred1",
                                    "male" = "cornflowerblue")) +
  theme_bw()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

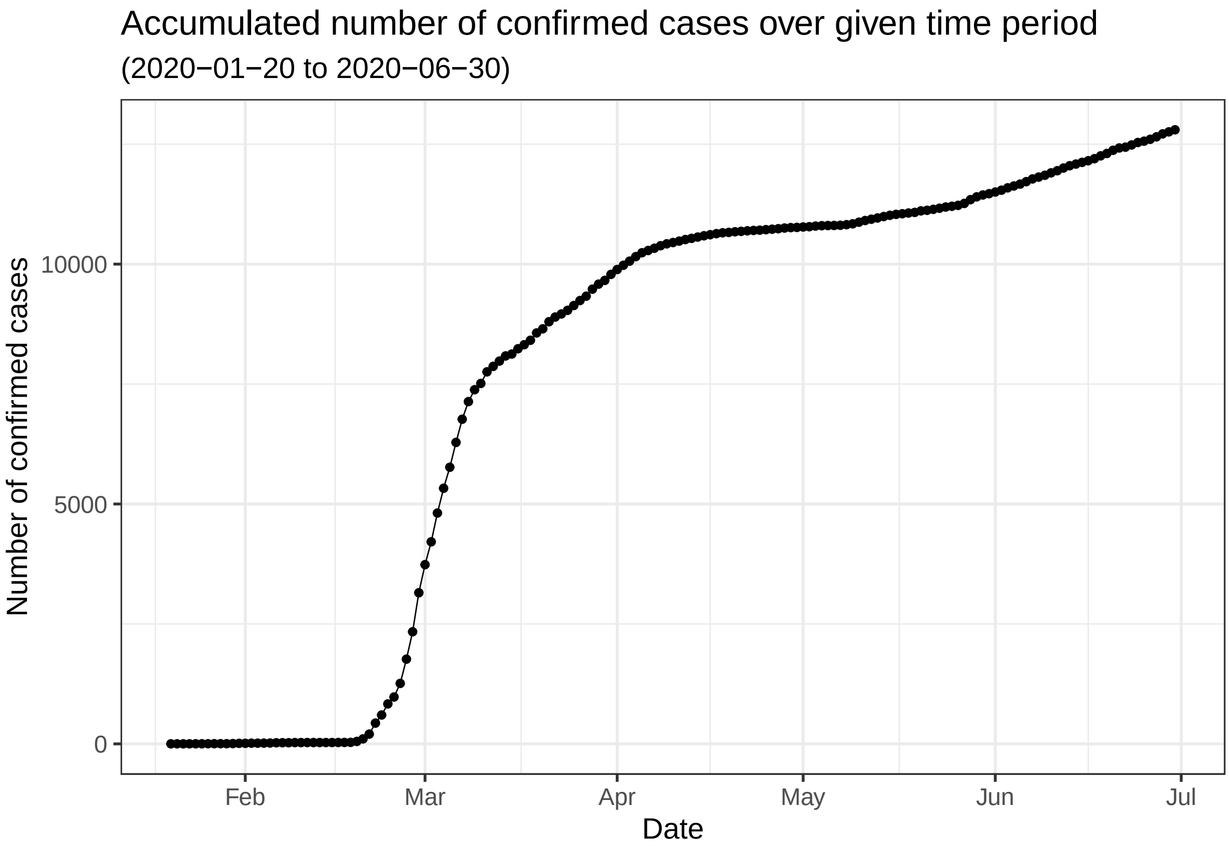


ANALYSIS ON GOVERNMENT POLICIES

```r
#value for zero row
Time[, confirmed_per_day := confirmed - shift(confirmed, fill = 0, type = 'lag')]
```

Plot Nr.23: Accumulated number of confirmed cases over given time period (2020-01-20 to 2020-06-30)

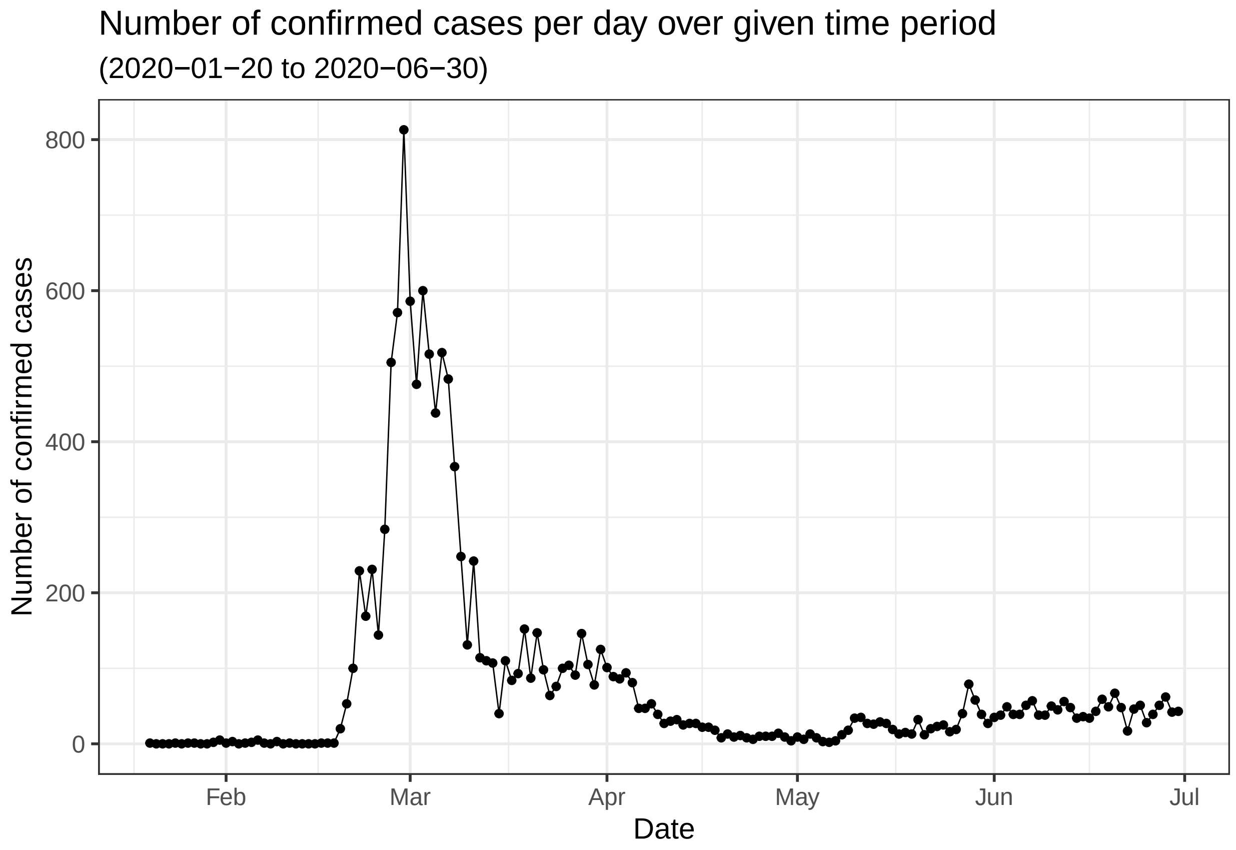
```r
plot_1 <- ggplot(Time, aes(date, confirmed)) +
  geom_point(size = 1) +
  geom_line(size = 0.25) +
  scale_color_brewer(palette='Set3') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases',
       title = 'Accumulated number of confirmed cases over given time period',
       subtitle = '(2020-01-20 to 2020-06-30)') +
  theme_bw()

plot_1
```



Plot Nr.24: Number of confirmed cases per day over given time period

(2020-01-20 to 2020-06-30)

```
plot_2 <- ggplot(Time, aes(date, confirmed_per_day)) +
  geom_point(size = 1) +
  geom_line(size = 0.25) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases',
       title = 'Number of confirmed cases per day over given time period',
       subtitle = '(2020-01-20 to 2020-06-30)') +
  theme_bw()

plot_2
```



Working on Policy

Rearranging and merging data

```
policy_reshaped <- Policy[, date := start_date] #variable date to identify observations
policy_merged <- merge(Time, policy_reshaped, by='date', group_by='date',all=TRUE) %>%
                 as.data.table(policy_merged)

policy_sum <- Policy[, .(count = .N), by=date]
policy_sum[, count_accumulated := cumsum(count)]
```

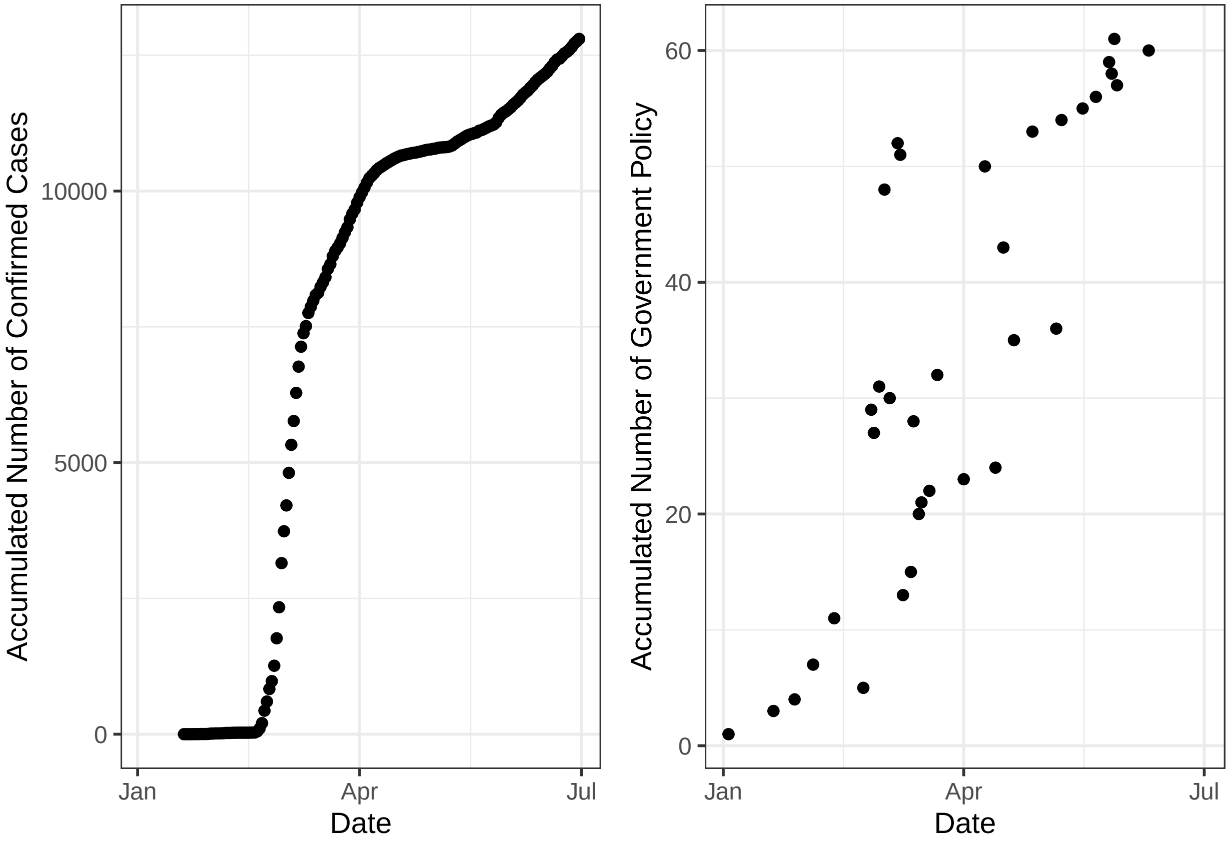
```
policy_sum <- merge(policy_sum, Time, by='date', group_by='date', all=TRUE)
```

Plot Nr.25: Comparison of Accumulated Confirmed Cases and

Accumulated Number of Government Policy ### (2020-01-20 to 2020-06-30)

```
plot1.1 <- ggplot(policy_sum, aes(date, confirmed)) +
           geom_point() +
           labs(x = 'Date', y = 'Accumulated Number of Confirmed Cases') +
           theme_bw()

plot3.1 <- ggplot(policy_sum, aes(date, count_accumulated)) +
           geom_point() +
           labs(x = 'Date', y = 'Accumulated Number of Government Policy') +
           theme_bw()

grid.arrange(plot1.1, plot3.1, ncol=2)
```

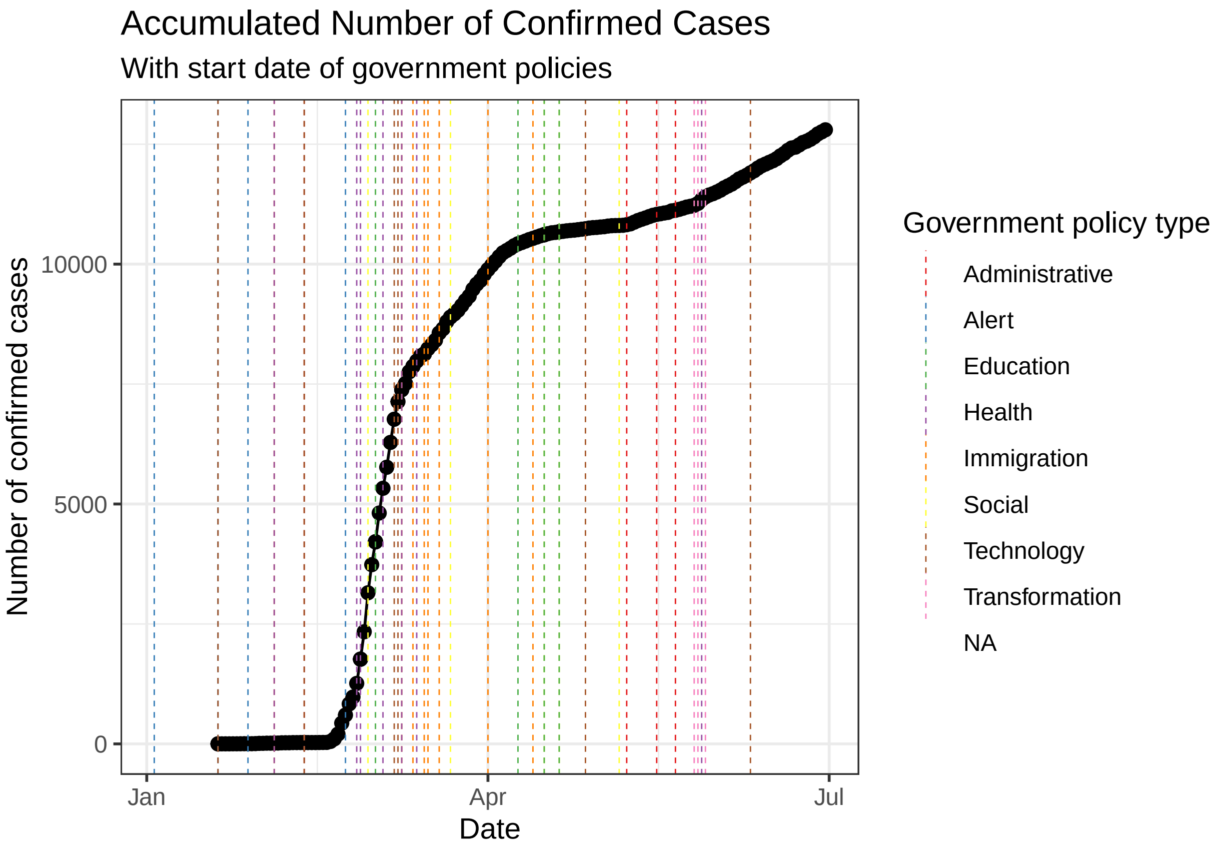
```
## Warning: Removed 1 rows containing missing values (geom_point).

## Warning: Removed 129 rows containing missing values (geom_point).
```



Plot Nr.26: Show events of government policies

```
ggplot(policy_merged, aes(date, confirmed)) +
  geom_line() +
  geom_point(size=2) +
  geom_vline(aes(xintercept = date, color = type), size = 0.25, linetype = 'dashed') +
  labs(color = 'Government policy type') +
  scale_color_brewer(palette='Set1') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases',
       title = "Accumulated Number of Confirmed Cases",
       subtitle = "With start date of government policies") +
  theme_bw()
```

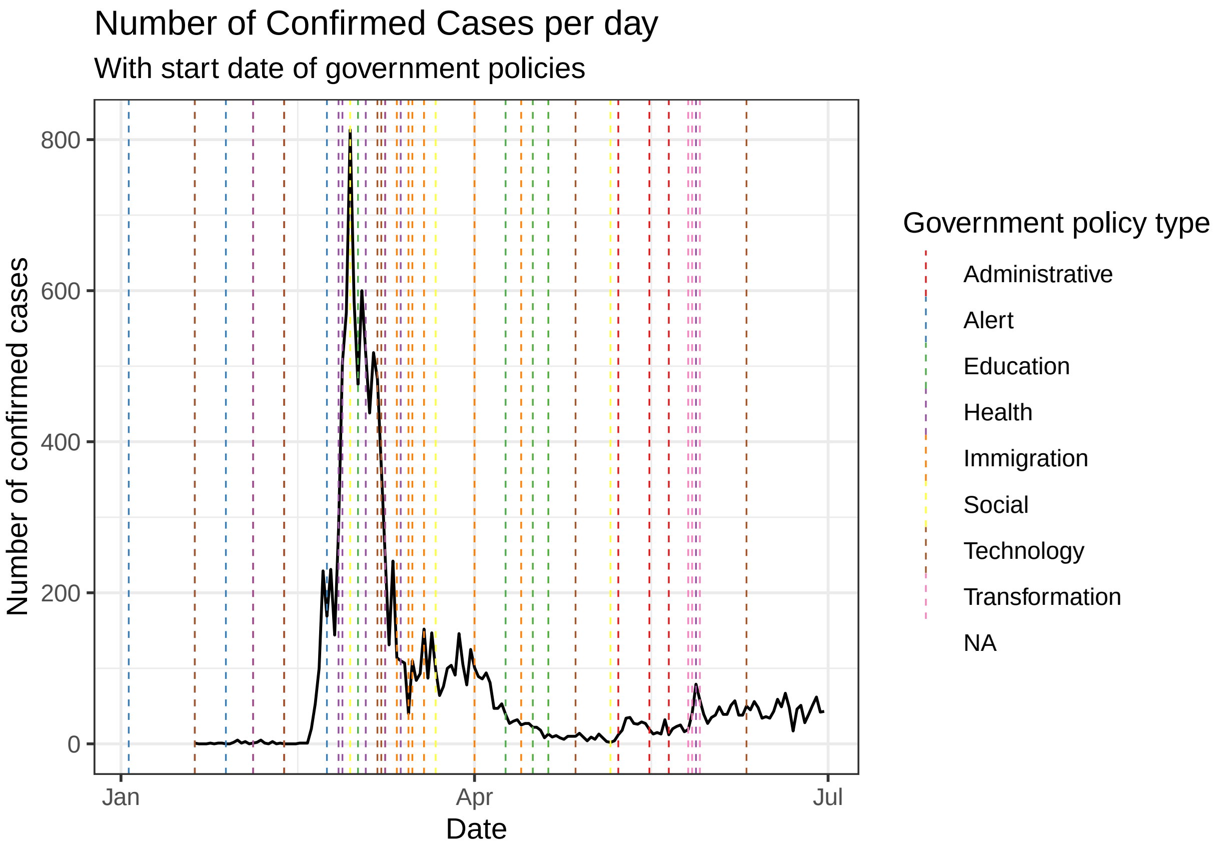
```
## Warning: Removed 1 row(s) containing missing values (geom_path).

## Warning: Removed 1 rows containing missing values (geom_point).
```



Plot Nr.27: Show events of government policies

```
plot_3 <- ggplot(policy_merged, aes(date, confirmed_per_day)) +
          geom_line() +
          geom_vline(aes(xintercept=date, color=type), size=0.3, linetype = 'dashed') +
          labs(color = 'Government policy type') +
          scale_color_brewer(palette='Set1') +
          theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
          labs(x = 'Date', y = 'Number of confirmed cases',
               title = "Number of Confirmed Cases per day",
               subtitle = "With start date of government policies") +
          theme_bw()

plot_3
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```

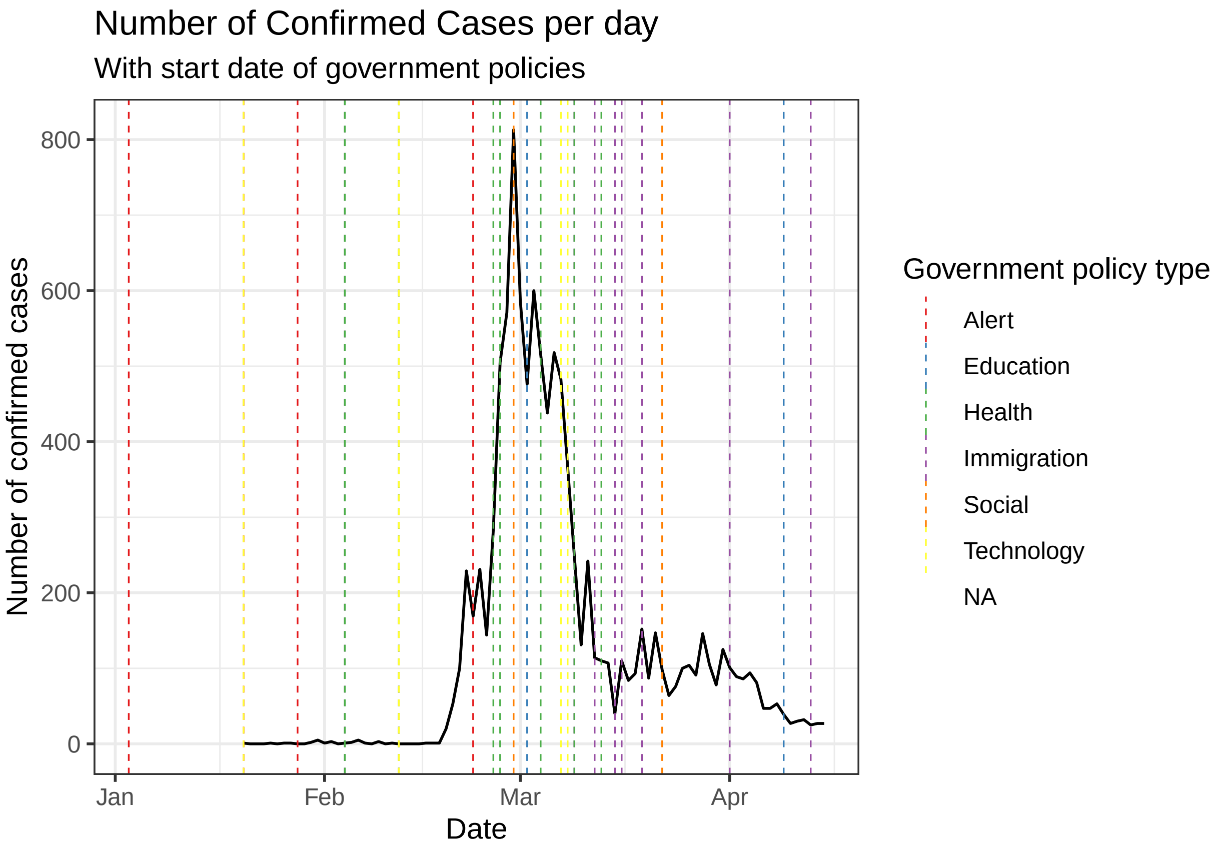


Plot Nr.28: Show events of government policies

For the first big peak

```
policy_merged_reshaped1 <- policy_merged[date >= as.Date("2020-01-03") &
                                          date <= as.Date("2020-04-15"),
                                          .(date,confirmed_per_day,
                                            type,detail,start_date)]
#for a better overview of the data
ggplot(policy_merged_reshaped1, aes(date, confirmed_per_day)) +
  geom_line() +
  geom_vline(aes(xintercept = date, color = type), size = 0.3, linetype = 'dashed') +
  labs(color = 'Government policy type') +
  scale_color_brewer(palette='Set1') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases',
       title = "Number of Confirmed Cases per day",
       subtitle = "With start date of government policies") +
  theme_bw()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



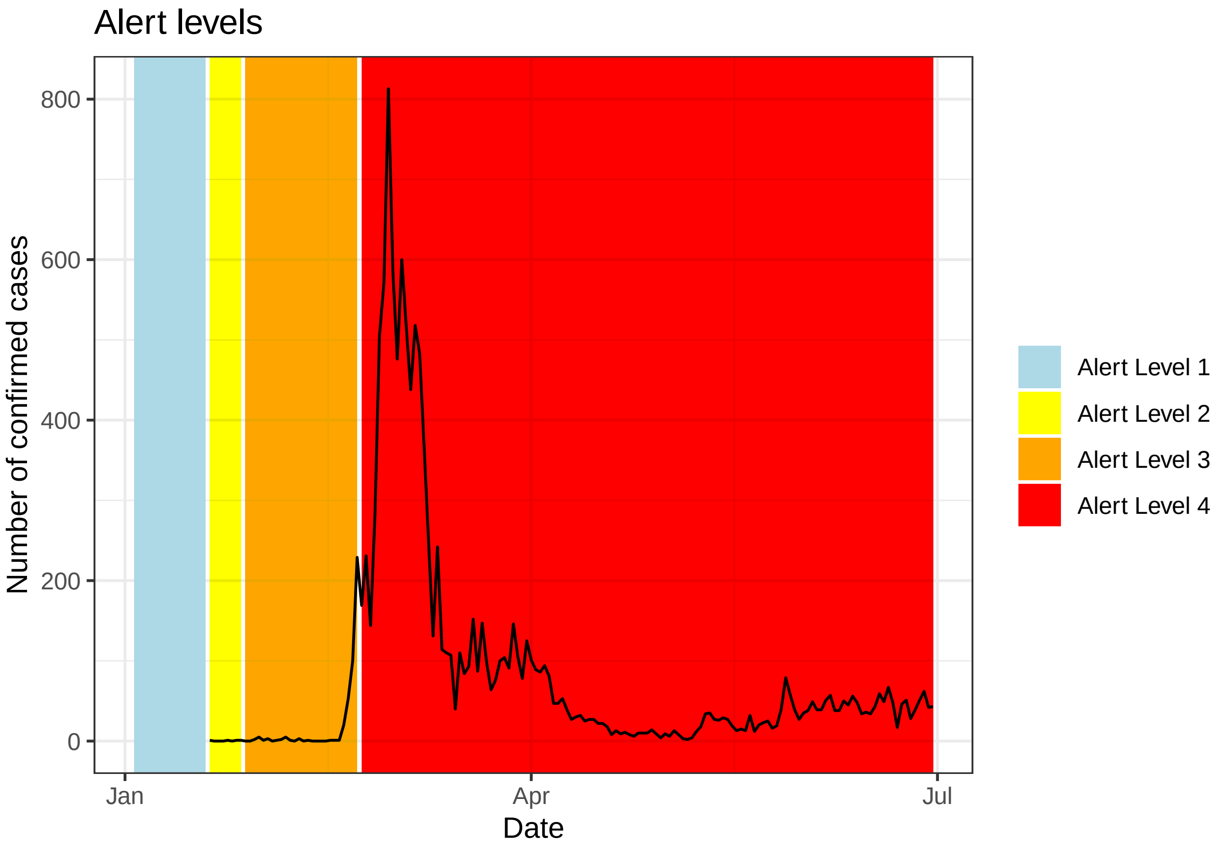
Plot Nr.29: Alert Level of Policies

```
alert_level <- Policy[type == 'Alert']
alert_level[,date] #View start_date of each Alert Level
```

```
## [1] "2020-01-03" "2020-01-20" "2020-01-28" "2020-02-23"
```

```
plot_4 <- ggplot(policy_merged, aes(date, confirmed_per_day)) +
  #Geom_rect for Level 1
        geom_rect(aes(xmin = as.Date("2020-01-03", "%Y-%m-%d"),
                      xmax = as.Date("2020-01-19",  "%Y-%m-%d"),
                      ymin = -Inf, ymax = Inf, fill = 'Alert Level 1'),
                  alpha = 0.03) +
        #Geom_rect for Level 2
        geom_rect(aes(xmin = as.Date("2020-01-20", "%Y-%m-%d"),
                      xmax = as.Date("2020-01-27",  "%Y-%m-%d"),
                      ymin = -Inf, ymax = Inf, fill = 'Alert Level 2'),
                  alpha = 0.03) +
        #Geom_rect for Level 3
        geom_rect(aes(xmin = as.Date("2020-01-28", "%Y-%m-%d"),
                      xmax = as.Date("2020-02-22",  "%Y-%m-%d"),
                      ymin = -Inf, ymax = Inf, fill = 'Alert Level 3'),
                  alpha = 0.03) +
        #Geom_rect for Level 4
        geom_rect(aes(xmin = as.Date("2020-02-23", "%Y-%m-%d"),
                      xmax = as.Date("2020-06-30",  "%Y-%m-%d"),
                      ymin = -Inf, ymax = Inf, fill = 'Alert Level 4'),
                    alpha = 0.03) +
          geom_line() +
          scale_fill_manual(name ='',
                            values = c('lightblue','yellow', 'orange','red'),
                            guide = guide_legend(override.aes = list(alpha = 1))) +
          theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
          labs(x = 'Date', y = 'Number of confirmed cases') +
          ggtitle('Alert levels')+ theme_bw()

plot_4
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



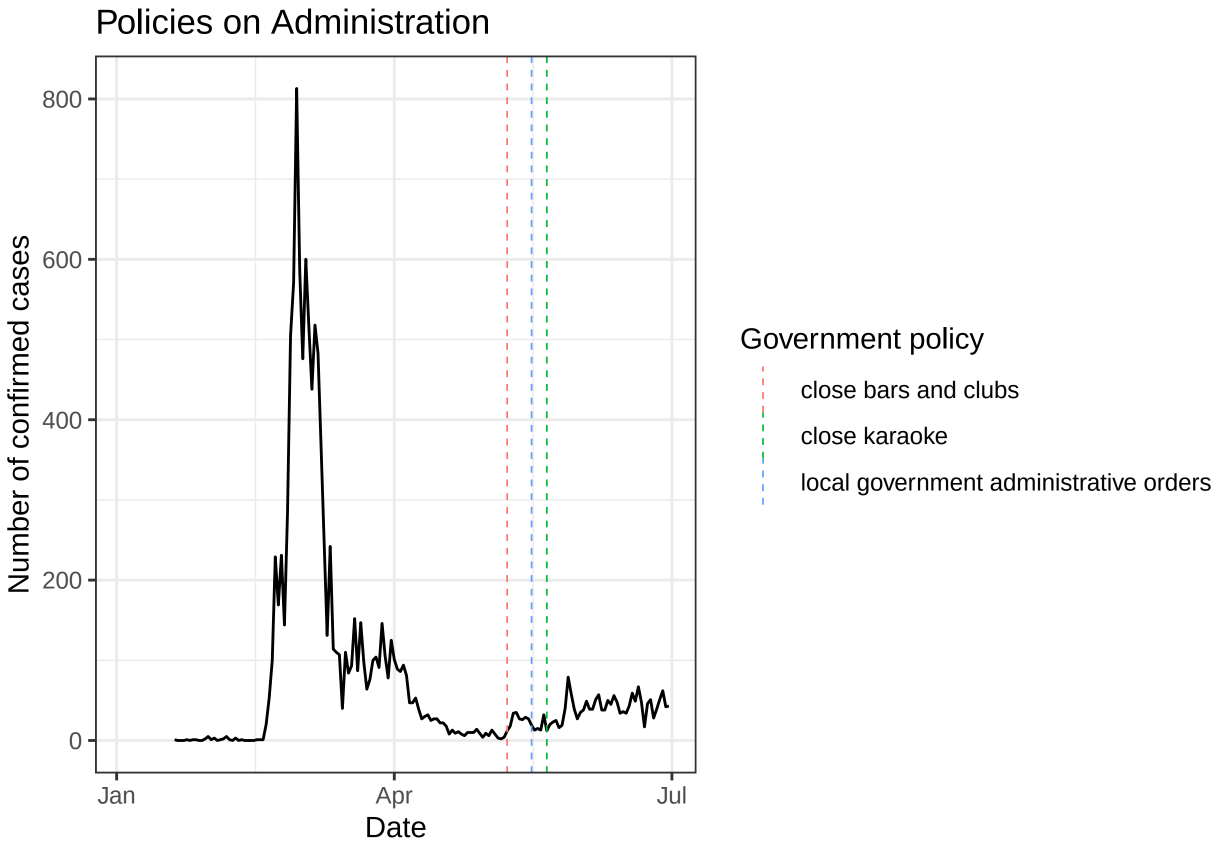
Plot Nr.30: Policies on Administration

```
Policy[type=='Administrative', 4:6] # view all policy titles on administration
```

```
##                                 gov_policy
## 1:                   Close bars and clubs
## 2: local government Administrative orders
## 3:                          Close karaoke
##
## 1:
## 2: Gathering limited administrative orders for nursing homes, nursing homes, elderly day care center
## 3:                                                                                     Ban on gather
##     start_date
## 1: 2020-05-08
## 2: 2020-05-16
## 3: 2020-05-21
```

```
ggplot(policy_merged, aes(date, confirmed_per_day)) +
  geom_line() +
  geom_vline(data = subset(policy_merged, type == 'Administrative'), # filter data source
             aes(xintercept = start_date,
                 color = c('close bars and clubs',
                           'local government administrative orders',
                           'close karaoke')), size = 0.3, linetype = 'dashed') +
  labs(color = 'Government policy') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases') +
  ggtitle('Policies on Administration') +
  theme_bw()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



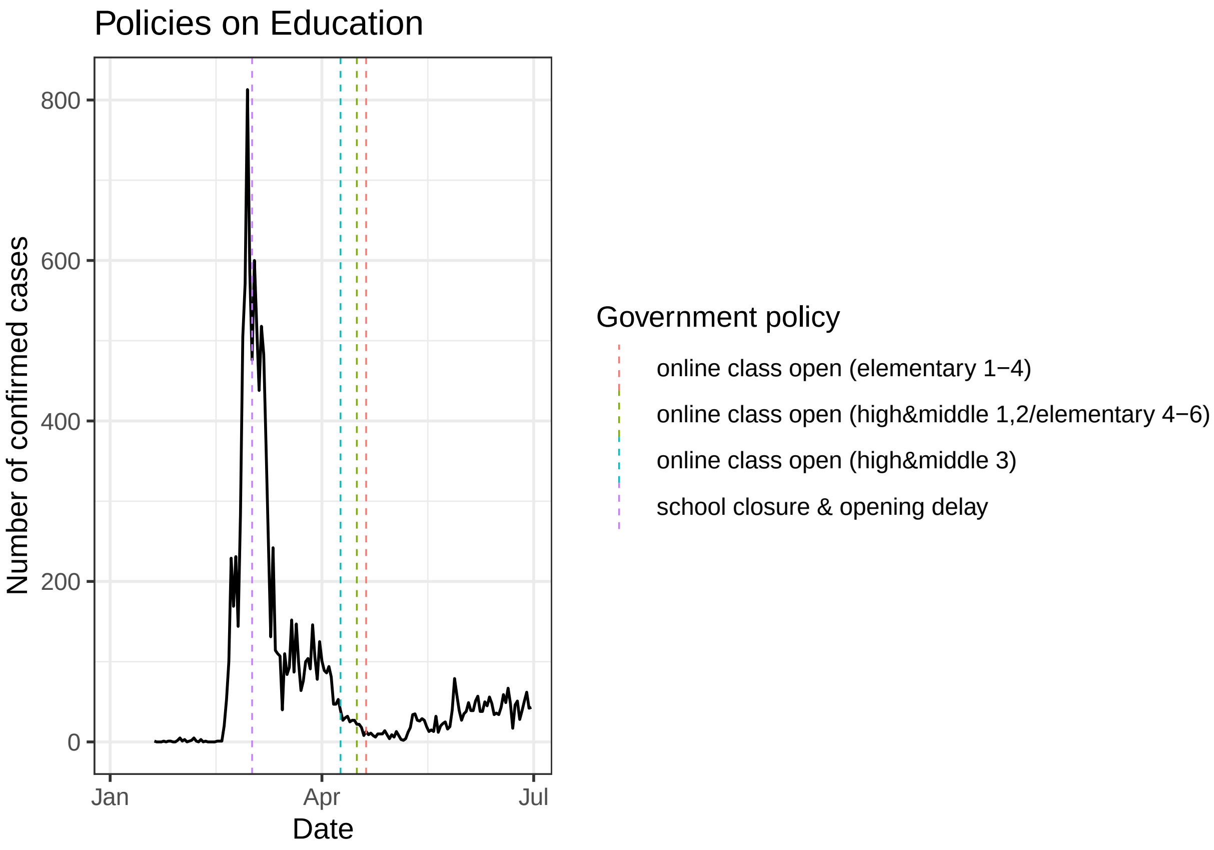
Plot Nr.31: Policies on Education

```
Policy[type=='Education', 4:6] %>%
  group_by(start_date) #view all policy titles on education
```

```
## # A tibble: 15 x 3
## # Groups:   start_date [4]
##    gov_policy                       detail                           start_date
##    <chr>                            <chr>                            <date>
##  1 School Closure                   Daycare Center for Children      2020-03-02
##  2 School Opening Delay            Kindergarten                      2020-03-02
##  3 School Opening Delay            High School                       2020-03-02
##  4 School Opening Delay            Middle School                     2020-03-02
##  5 School Opening Delay            Elementary School                 2020-03-02
##  6 School Opening with Online Class High School (3rd grade)           2020-04-09
##  7 School Opening with Online Class High School (2nd grade)           2020-04-16
##  8 School Opening with Online Class High School (1st grade)           2020-04-16
##  9 School Opening with Online Class Middle School (3rd grade)         2020-04-09
## 10 School Opening with Online Class Middle School (2nd grade)         2020-04-16
## 11 School Opening with Online Class Middle School (1st grade)         2020-04-16
## 12 School Opening with Online Class Elementary School (5th ~ 6th gra~ 2020-04-16
## 13 School Opening with Online Class Elementary School (4th grade)     2020-04-16
## 14 School Opening with Online Class Elementary School (3rd grade)     2020-04-20
## 15 School Opening with Online Class Elementary School (1st ~ 2nd gra~ 2020-04-20
```

```
#summarize policy titles for each start_date
policy_education <- Policy[type=='Education', .(date=unique(start_date))]
policy_education[,gov_policy_grouped := c('school closure & opening delay',
                                          'online class open (high&middle 3)',
                                          'online class open (high&middle
                                          1,2/elementary 4-6)',
                                          'online class open (elementary 1-4)')]

ggplot(policy_merged, aes(date, confirmed_per_day)) +
  geom_line() +
  geom_vline(data = subset(policy_education, date==date), # filter data source
             aes(xintercept = date,
                 color=c('school closure & opening delay',
                         'online class open (high&middle 3)',
                         'online class open (high&middle 1,2/elementary 4-6)',
                         'online class open (elementary 1-4)')), size = 0.3,
             linetype = 'dashed') +
  labs(color = 'Government policy') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases') +
  ggtitle('Policies on Education') +
  theme_bw()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



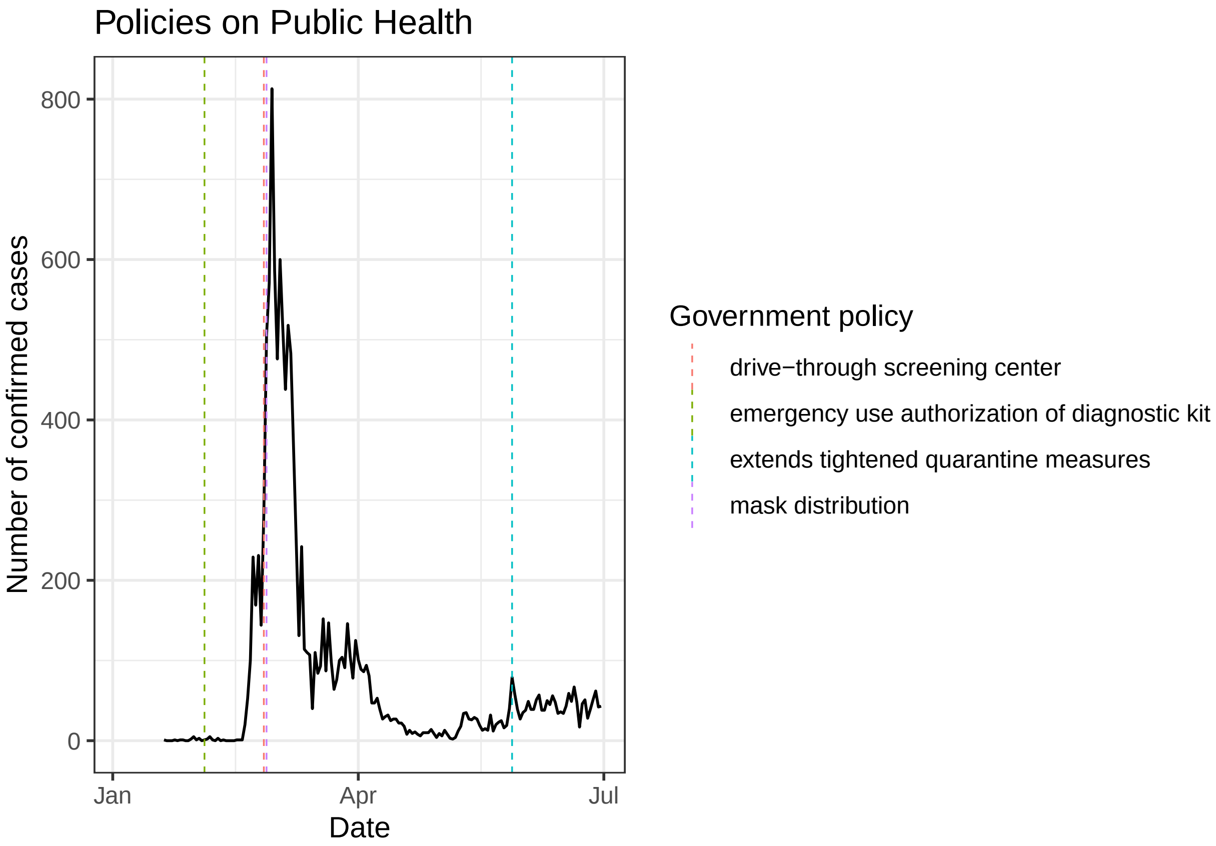
Plot Nr.32: Policies on Public Health

```
Policy[type=='Health', 4:6] %>% group_by(start_date) #view all policy titles on health
```

```
## # A tibble: 10 x 3
## # Groups:   start_date [8]
##    gov_policy                      detail                                start_date
##    <chr>                           <chr>                                 <date>
##  1 Emergency Use Authorization ~ 1st EUA                               2020-02-04
##  2 Emergency Use Authorization ~ 2nd EUA                               2020-02-12
##  3 Emergency Use Authorization ~ 3rd EUA                               2020-02-27
##  4 Emergency Use Authorization ~ 4th EUA                               2020-02-27
##  5 Emergency Use Authorization ~ 5th EUA                               2020-03-13
##  6 Drive-Through Screening Cent~ by Local Government                   2020-02-26
##  7 Drive-Through Screening Cent~ Standard Operating Procedures         2020-03-04
##  8 Mask Distribution             Public-Sale                           2020-02-27
##  9 Mask Distribution             5-day Rotation System                 2020-03-09
## 10 Extends Tightened Quarantine~ Gov't Extends Tightened Quarantine ~ 2020-05-28
```

```
policy_health <- Policy[type=='Health',.(start_date=first(start_date)),
                        by=gov_policy] #only first policy implementation

ggplot(policy_merged, aes(date, confirmed_per_day)) +
  geom_line() +
  geom_vline(data = policy_health,
             aes(xintercept = start_date,
                 color = c('emergency use authorization of diagnostic kit',
                             'drive-through screening center', 'mask distribution',
                             'extends tightened quarantine measures' )), size = 0.3,
              linetype = 'dashed') +
  labs(color = 'Government policy') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases') +
  ggtitle('Policies on Public Health') +
  theme_bw()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



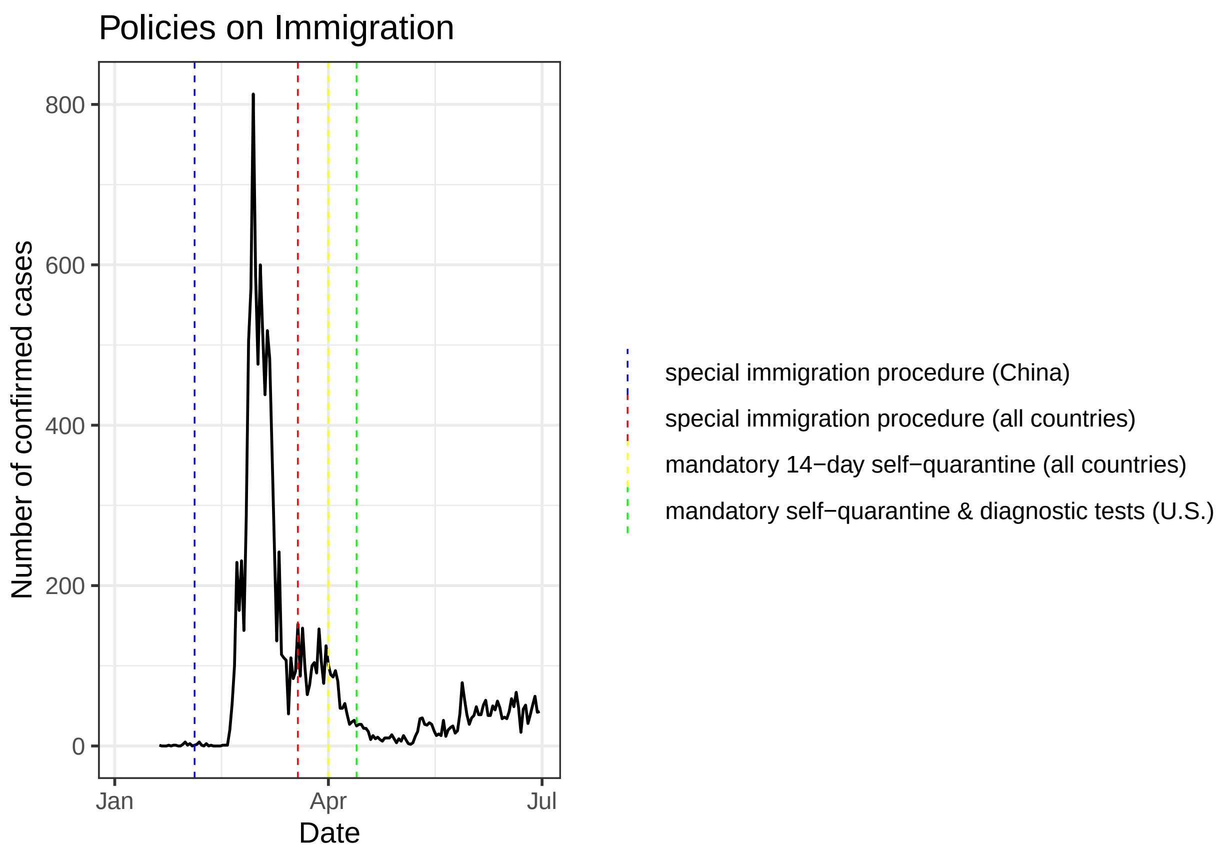
Plot Nr.33: Policies on Immigration

```
Policy[type=='Immigration', 4:6] %>%
  group_by(start_date) #view all policy titles on health
```

```
## # A tibble: 15 x 3
## # Groups:   start_date [9]
##    gov_policy                             detail              start_date
##    <chr>                                  <chr>               <date>
##  1 Special Immigration Procedure          from China          2020-02-04
##  2 Special Immigration Procedure          from Hong Kong      2020-02-12
##  3 Special Immigration Procedure          from Macau          2020-02-12
##  4 Special Immigration Procedure          from Japan          2020-03-09
##  5 Special Immigration Procedure          from Italy          2020-03-12
##  6 Special Immigration Procedure          from Iran           2020-03-12
##  7 Special Immigration Procedure          from France         2020-03-15
##  8 Special Immigration Procedure               from Germany          2020-03-15
##  9 Special Immigration Procedure               from Spain            2020-03-15
## 10 Special Immigration Procedure               from U.K.             2020-03-15
## 11 Special Immigration Procedure               from Netherlands      2020-03-15
## 12 Special Immigration Procedure               from Europe           2020-03-16
## 13 Special Immigration Procedure               from all the countri~ 2020-03-19
## 14 Mandatory 14-day Self-Quarantine            from all the countri~ 2020-04-01
## 15 Mandatory Self-Quarantine & Diagonostic Tes~ from U.S.            2020-04-13
```

```
policy_immigration <- Policy[detail=='from China'|detail=='from all the countries'|
                               detail=='from U.S.',4:6] #select representative policies

ggplot(policy_merged, aes(date, confirmed_per_day)) +
  geom_line() +
  geom_vline(data = policy_immigration,
             aes(xintercept = start_date, color = c('blue','red','yellow','green')),
             size = 0.3, linetype = 'dashed') +
  scale_color_identity(name='',
                       breaks=c('blue','red','yellow','green'),
                       labels=c('special immigration procedure (China)',
                                'special immigration procedure (all countries)',
                                'mandatory 14-day self-quarantine (all countries)',
                                'mandatory self-quarantine & diagnostic tests (U.S.)'),
                       guide = 'legend') +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'Number of confirmed cases') +
  ggtitle('Policies on Immigration') +
  theme_bw()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



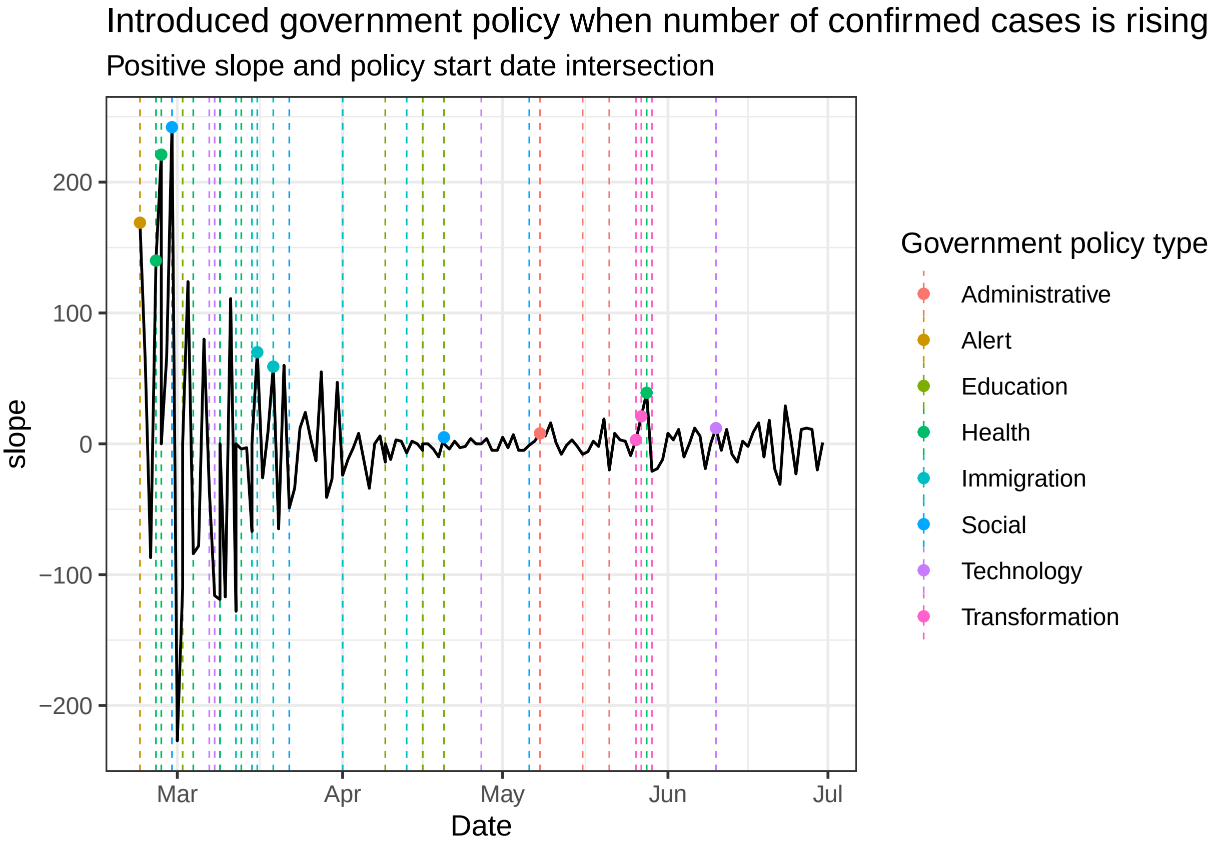
Time of Execution

```
policy_merged_reshaped <- policy_merged[date >= as.Date("2020-02-23") &
                                        date <= as.Date("2020-06-30"),
                                        .(date,confirmed_per_day,type,detail,start_date)]
#for a better overview of the data

time_slope <- policy_merged_reshaped %>%
              mutate(slope = confirmed_per_day - shift(confirmed_per_day,
                                                       fill = 0, type = 'lag'))
```

Plot Nr.34: Slope of the case numbers and policy start date intersections

```
plot_5 <- ggplot(time_slope, aes(x=date, y=slope)) +
  geom_vline(data = subset(Policy, start_date == date), # filter data source
             aes(xintercept = date, color = type),
             size = 0.3, linetype = 'dashed') +
  geom_line(aes(x=date, y=slope)) +
  labs(color = 'Government policy type') +
  geom_point(data = subset(time_slope, start_date == date & slope>0),
             aes(x=date, y=slope, color=type )) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = 'Date', y = 'slope',
       subtitle='Positive slope and policy start date intersection') +
  ggtitle('Introduced government policy when number of confirmed cases is rising') +
  theme_bw()
```

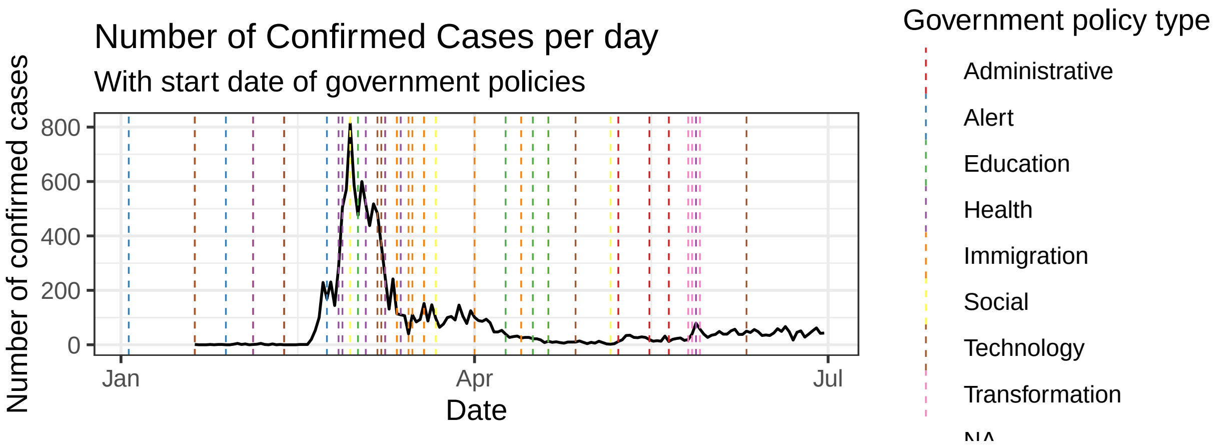
```
plot_5
```

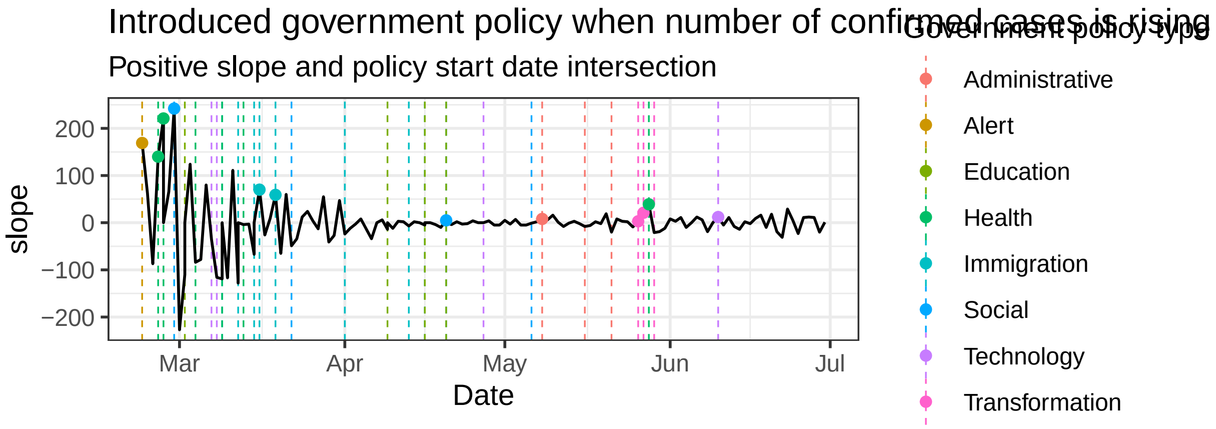


Plot Nr.35: Comparison of Number of Confirmed Cases and Slope with policy dates

```
require(gridExtra)
grid.arrange(plot_3, plot_5, nrow=2)
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```


Number of Confirmed Cases per day
With start date of government policies
Number of confirmed cases
800
600
400
200
0
Jan
Apr
Jul
Date
Government policy type
Administrative
Alert
Education
Health
Immigration
Social
Technology
Transformation
NA


Introduced government policy when number of confirmed cases is rising
Positive slope and policy start date intersection
slope
200
100
0
−100
−200
Mar
Apr
May
Jun
Jul
Date
Government policy type
Administrative
Alert
Education
Health
Immigration
Social
Technology
Transformation