COVID-19 Epidemic Information

Nick Lauerman

Contents

Set up

R Libraries

R is extended using packages or libraries. For this analysis the following packages are used.

```
library(tidyverse)
```

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

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  3.0.0     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

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

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

Note: the tidyverse package loads a set of packages that make up the tidy data universe. Use of the tidyverse package to load these packages is a convince instead of loading each individual backage.

Get data

Two R scripts are used to read in the data used. The source of this data is the WHO, for world wide data, and the CDC for US data. While the WHO data contains data for the US the CDC data is being used. The WHO data is a series of files scraped from their daily situtation reports and the CDC data is scraped from their web site for the CORANA virus. Due tothe WHO data being in 85 plus files the execution of these scripts is not shown to insure that the files is not overly large.

Create data set for Germany

The data for Germany is extracted from the WHO data for easy in displaying

```
germany <- WHO %>%
     filter(`Country Territory area` == "Germany")
```

Create data set for world wide data

This data set displays the daily total for world wide data.

```
world <- WHO %>%
     group_by(`date`) %>%
     summarise(`Number of new cases` = sum(`Confirmed new cases`),
               cum = sum(`total Confirmed Cases`))
world$`Week Number` <- week(world$date)
```

Charts and graphs

World Wide

```
epi <- ggplot(data = world)
```

```
epi + geom_point(aes(x = date,
               y = `Number of new cases`))+
     geom_smooth(aes(x = date,
               y = `Number of new cases`),
               color = "red",
               fill = "blue") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the WHO")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
epi + geom_col(aes(x=date,
                   y=`Number of new cases`),
                fill= "blue") +
     geom_smooth(aes(x=date,
                   y=`Number of new cases`),
                 color = "red") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the WHO")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

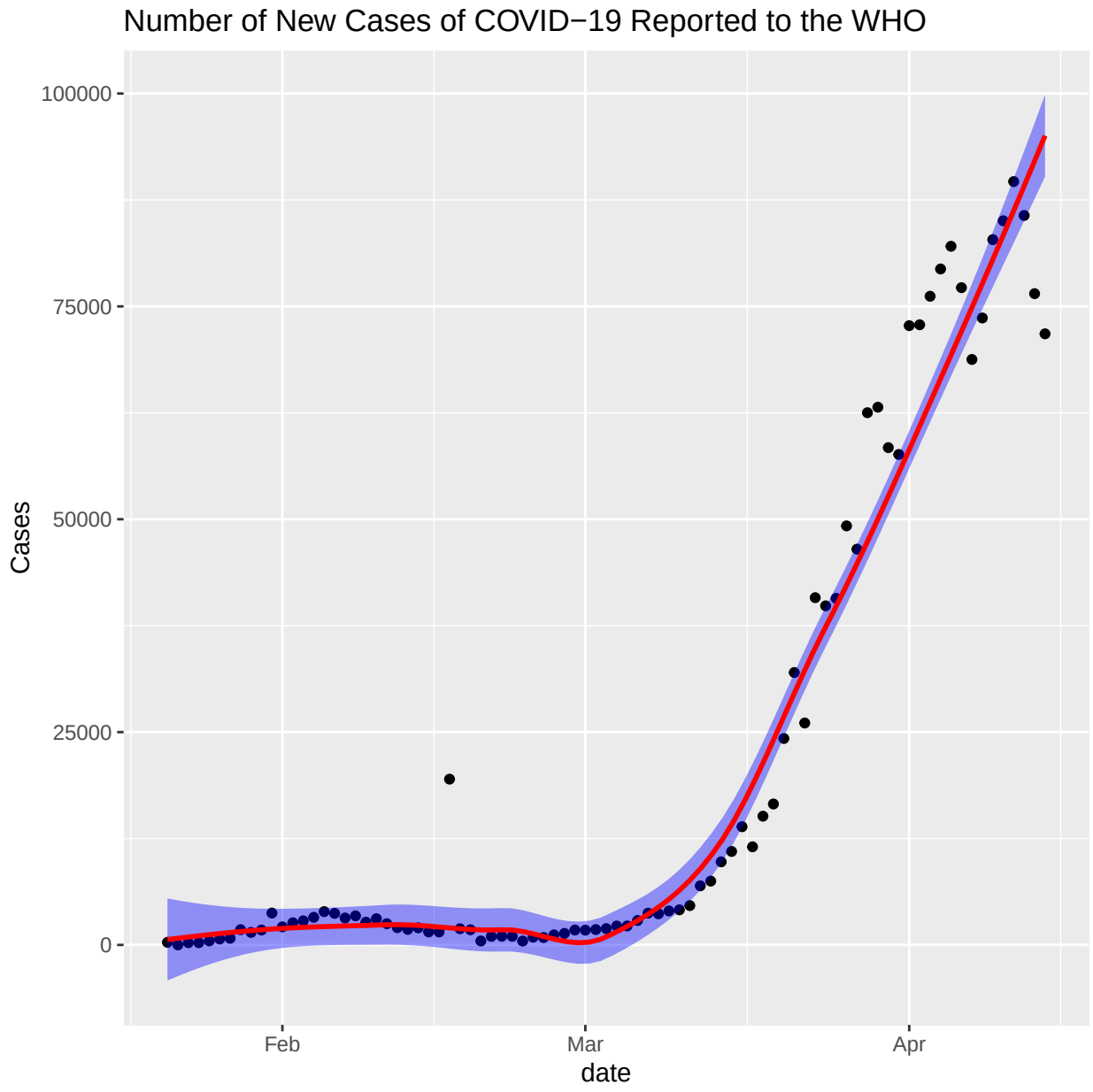

Figure 1: Epi curve 1

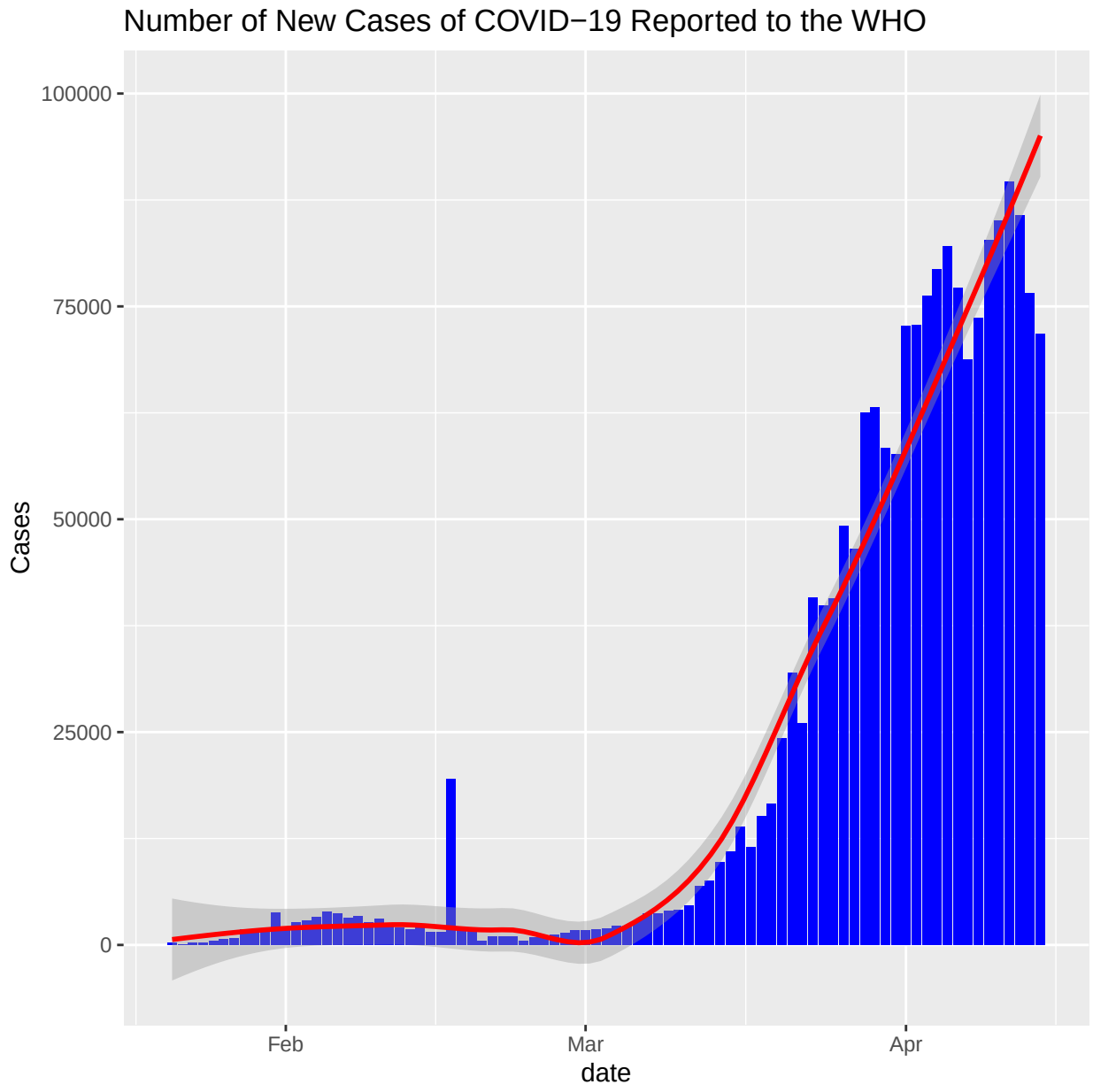

Figure 2: Epi curve 2, traditional

```
epi + geom_point(aes(x = date,
                     y = cum))+
     geom_smooth(aes(x = date,
                     y = `cum`),
                     color = "red",
                     fill = "blue") +
     labs(y = "Cumulative number of cases",
          title = "Cumulative Number of Cases of COVID-19 Reported to the WHO")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

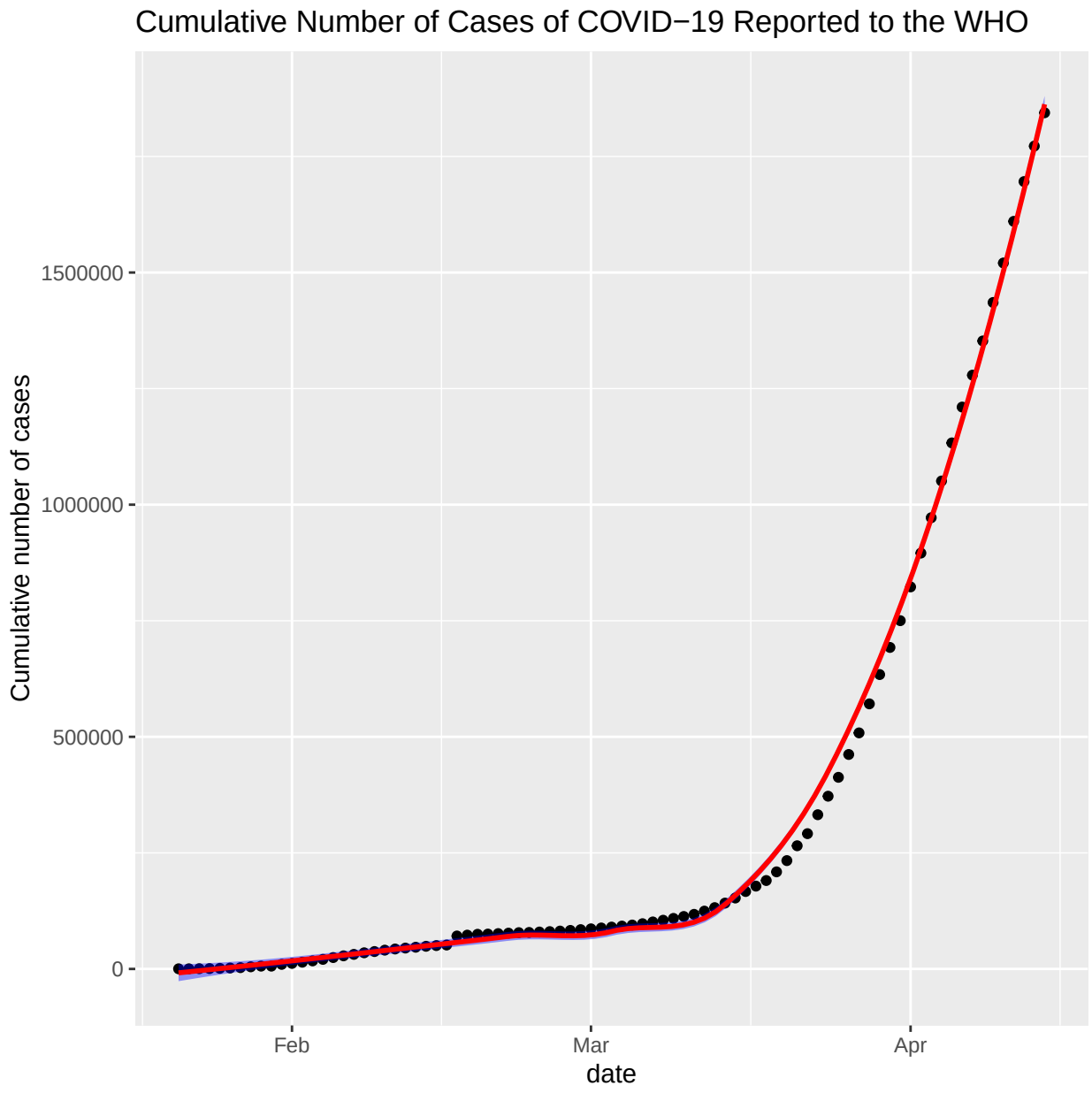

Figure 3: Cumulative cases

```
gr <- world %>%
  filter(`Number of new cases` != 0)

ggplot(data = gr) +
  geom_line(aes(y = `Number of new cases`,
                x = cum)) +
  scale_x_log10() +
  scale_y_log10() +
  geom_smooth(aes(y = `Number of new cases`,
                x = cum)) +
  labs(title = "Daily Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
gr <- world %>%
     group_by(`Week Number`) %>%
     summarise(cases = sum(`Number of new cases`),
               cs = sum(cum),
               count = n())

ggplot(data = gr) +
     geom_line(aes(y = `cases`,
                   x = cs)) +
     geom_point(aes(y = `cases`,
                    x = cs)) +
     scale_x_log10() +
     scale_y_log10() +
     geom_smooth(aes(y = `cases`,
                     x = cs)) +
  labs(title = "Weekly Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Germany

```
epi <- ggplot(data = germany)
```

```
epi + geom_point(aes(x = date,
               y = `Confirmed new cases`))+
     geom_smooth(aes(x = date,
               y = `Confirmed new cases`),
               color = "red",
               fill = "blue") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
epi + geom_col(aes(x=date,
                   y=`Confirmed new cases`),
                fill= "blue") +
```

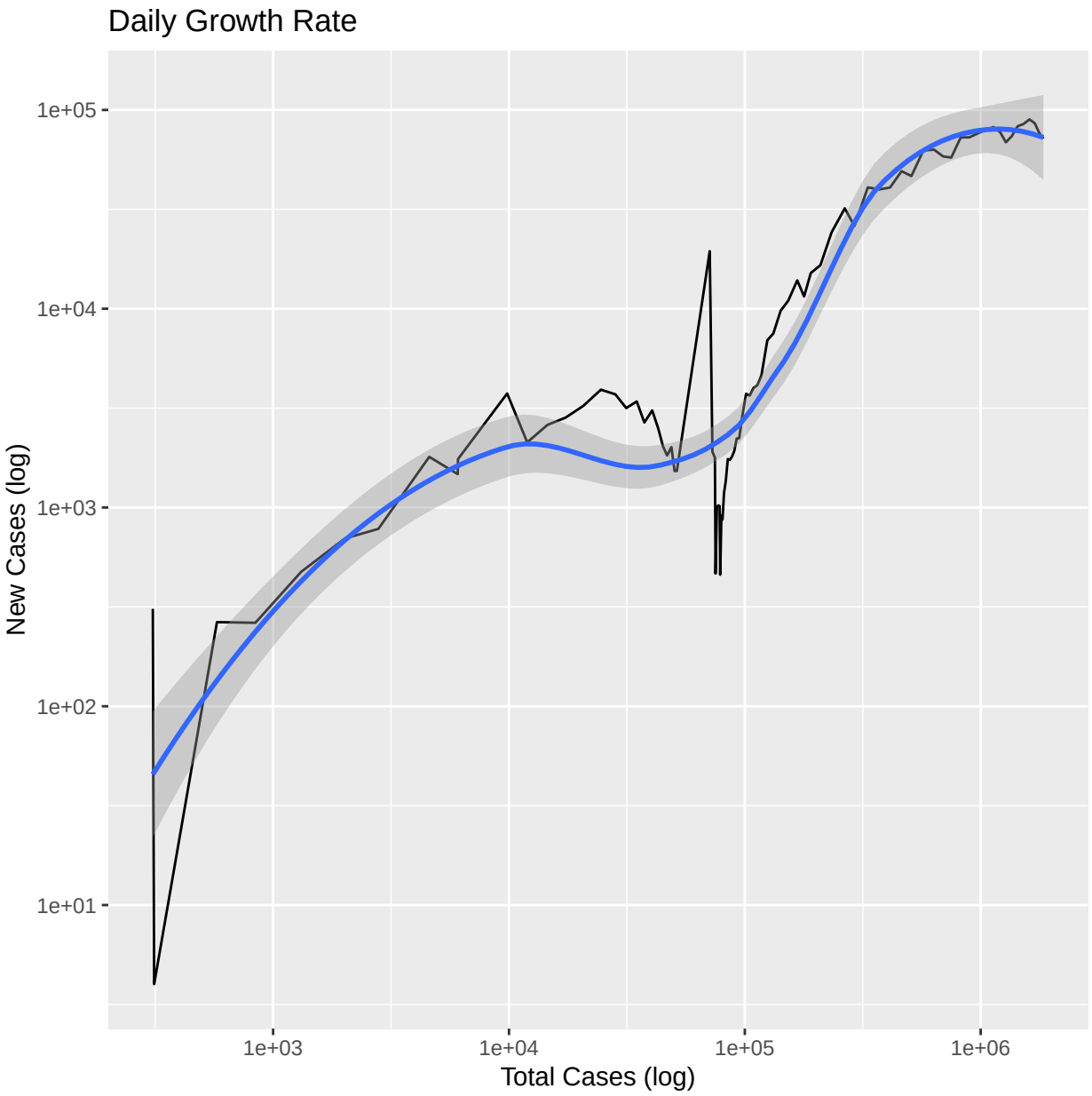

Figure 4: Growth Rate

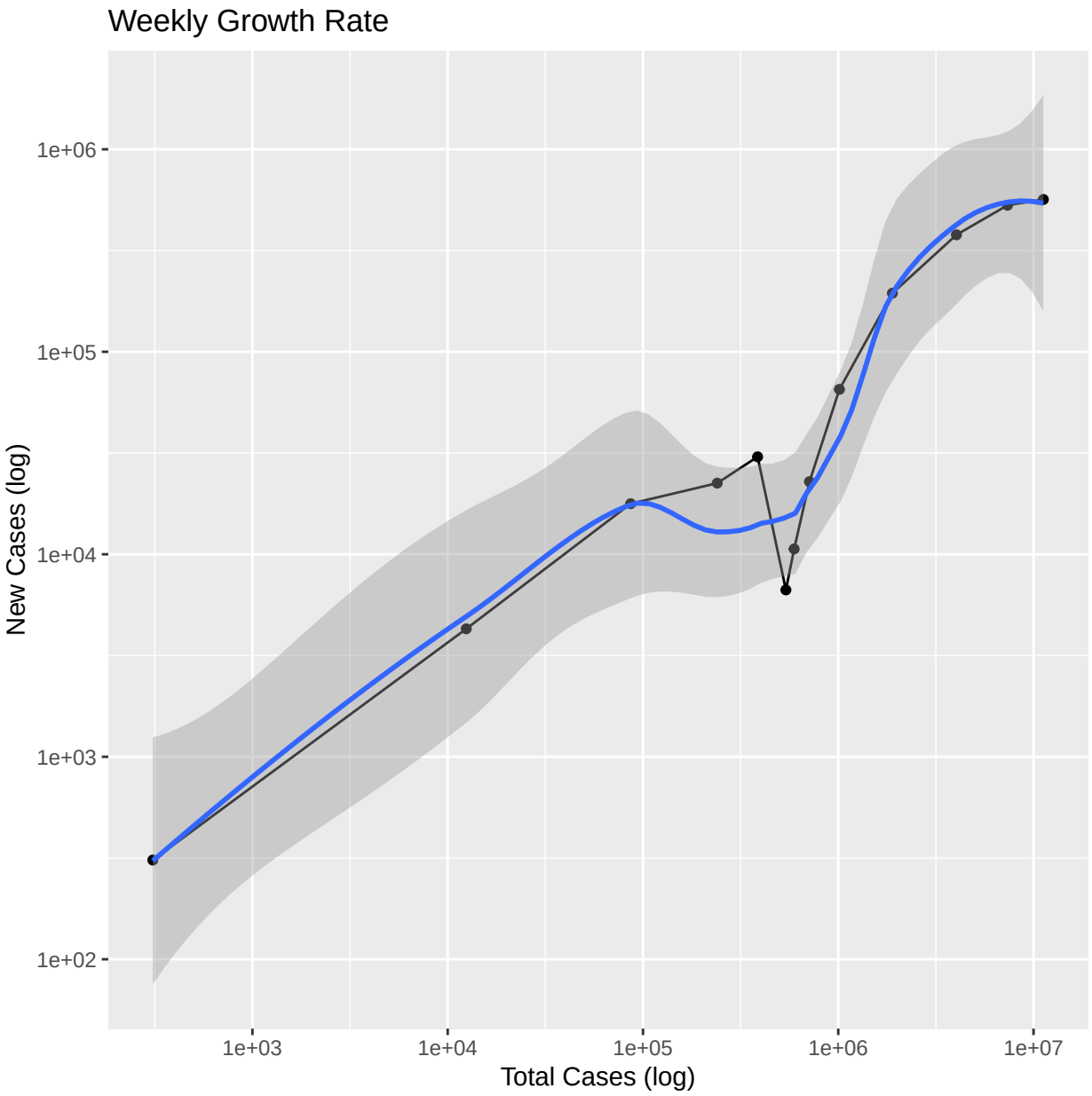

Figure 5: weekly growth rate

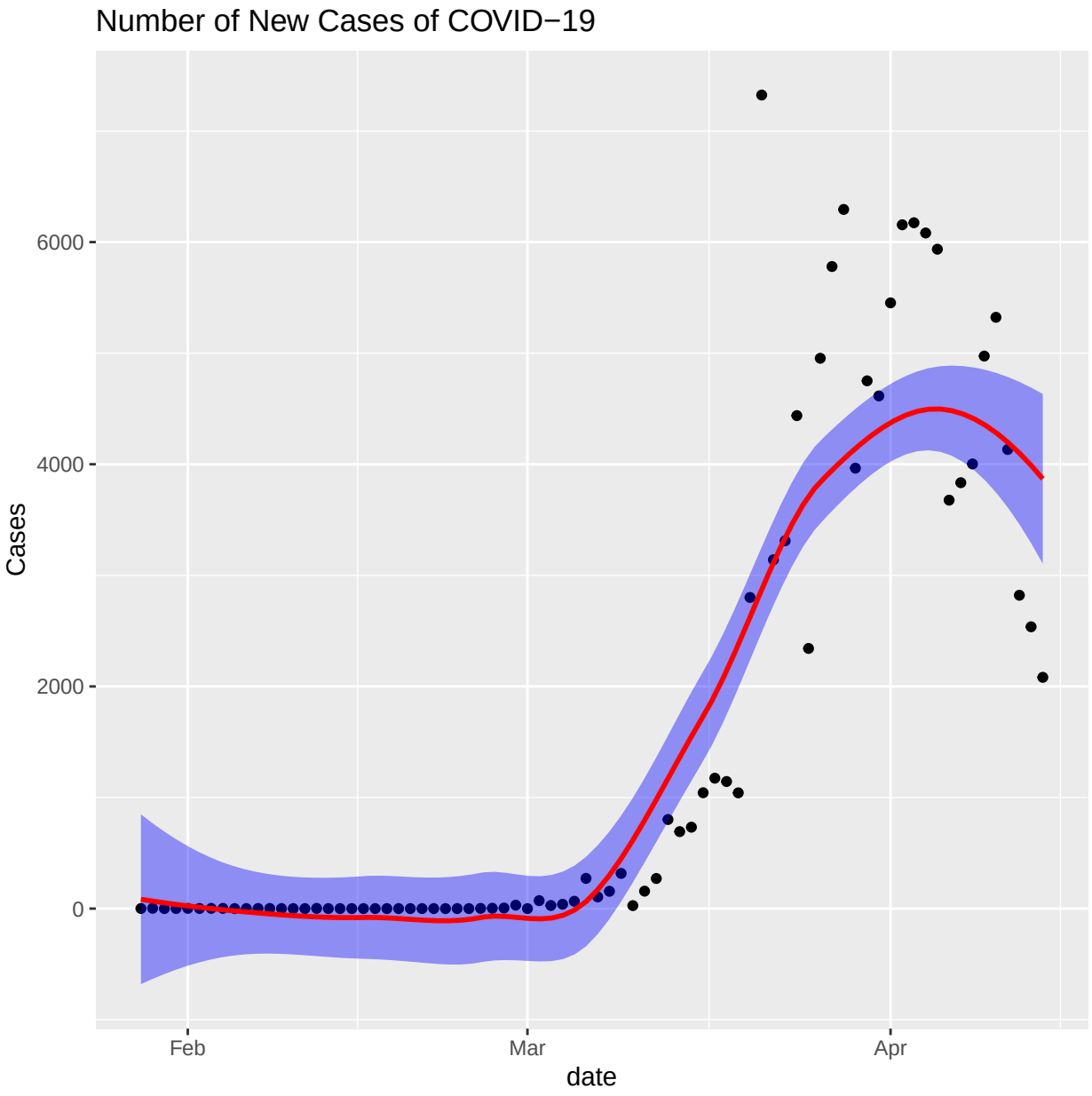

Figure 6: Epi curve 1

```
     geom_smooth(aes(x=date,
                     y=`Confirmed new cases`),
                 color = "red") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

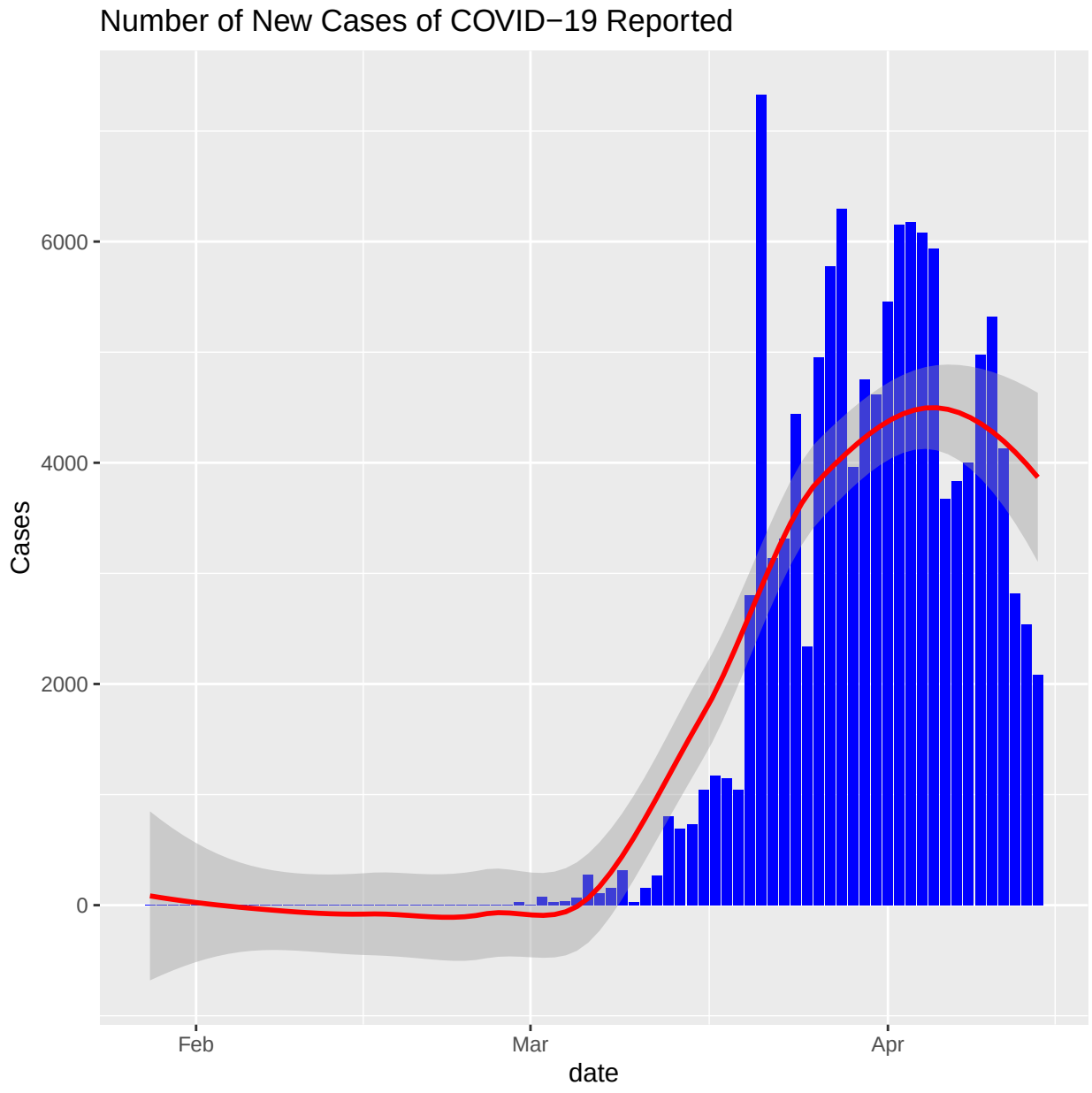

Figure 7: Epi curve 2, traditional

```
epi + geom_point(aes(x = date,
                     y = `total Confirmed Cases`))+
     geom_smooth(aes(x = date,
                     y = `total Confirmed Cases`),
```

```
              color = "red",
              fill = "blue") +
     labs(y = "Cumulative number of cases",
          title = "Cumulative Number of Cases of COVID-19 Reported")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

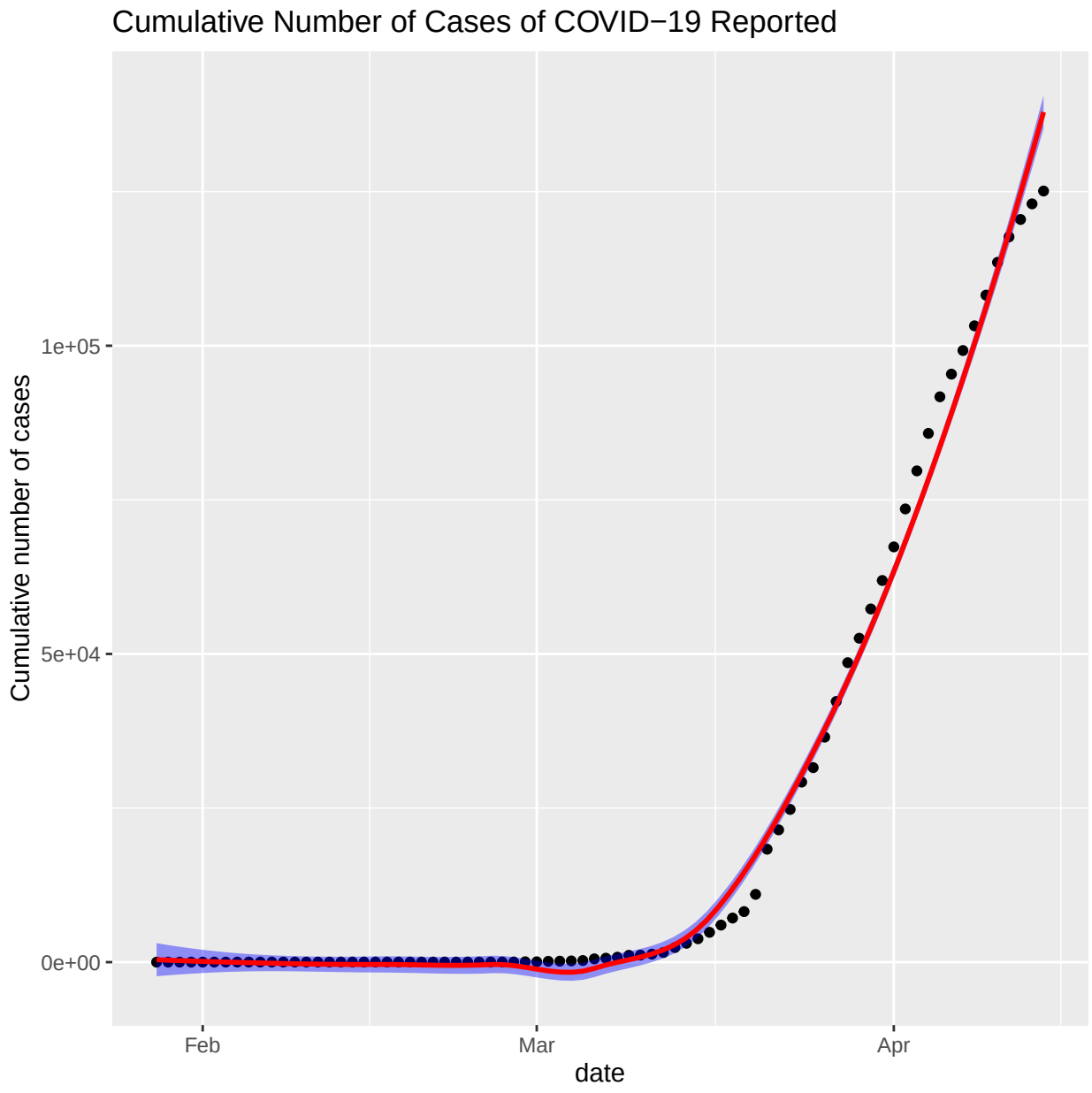

Figure 8: Cumulative cases

```
gr <- germany %>%
  filter(`Confirmed new cases` != 0)

ggplot(data = gr) +
  geom_line(aes(y = `Confirmed new cases`,
```

```
                  x = `total Confirmed Cases`)) +
  scale_x_log10() +
  scale_y_log10() +
  geom_smooth(aes(y = `Confirmed new cases`,
                  x = `total Confirmed Cases`)) +
  labs(title = "Daily Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

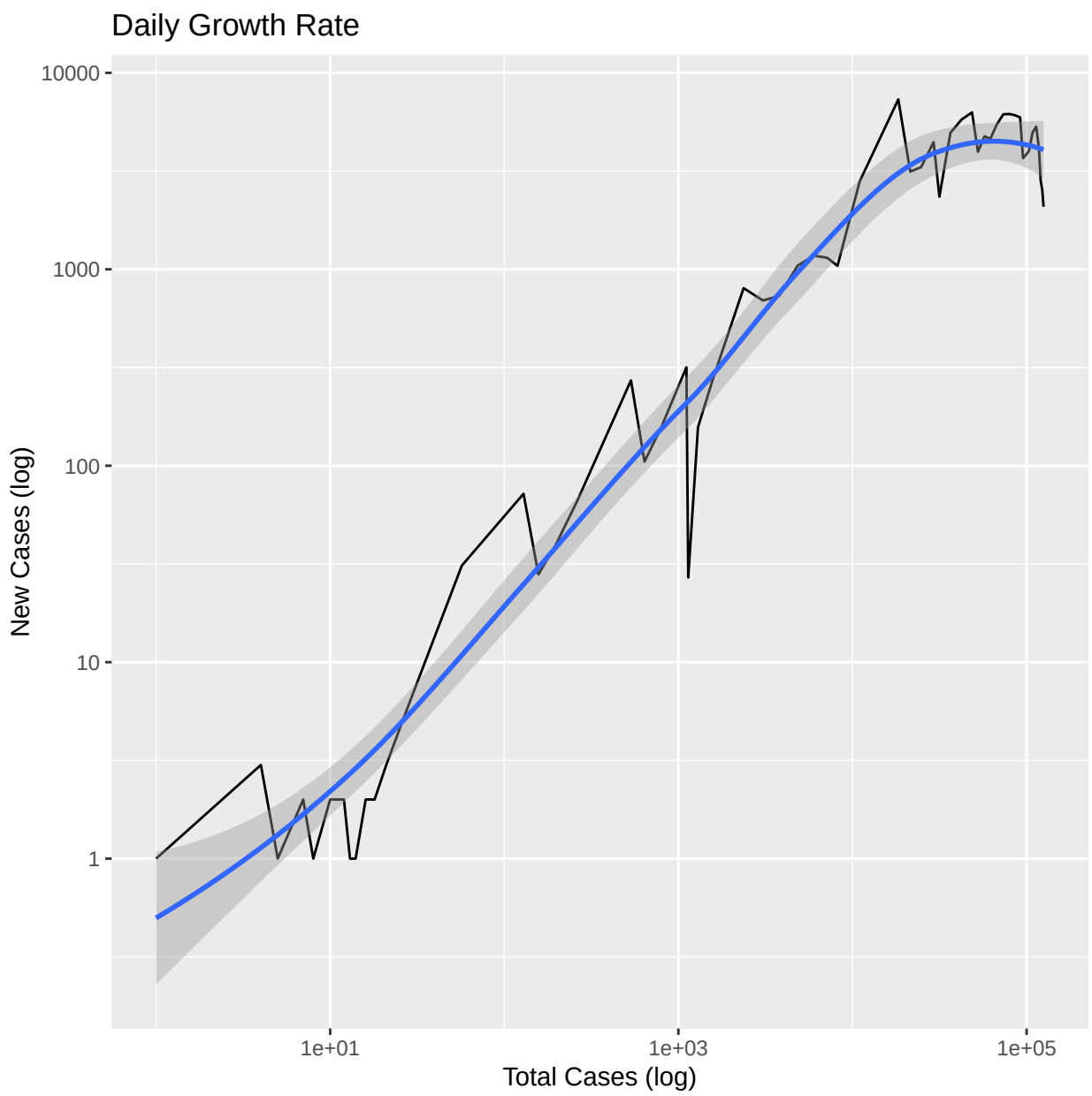

Figure 9: Growth Rate

```
gr <- germany %>%
     group_by(`Week Number`) %>%
     summarise(cases = sum(`Confirmed new cases`),
               cs = sum(`total Confirmed Cases`),
               count = n()) %>%
     filter(cases != 0)

ggplot(data = gr) +
     geom_line(aes(y = `cases`,
                   x = cs)) +
     geom_point(aes(y = `cases`,
                    x = cs)) +
     scale_x_log10() +
     scale_y_log10() +
     geom_smooth(aes(y = `cases`,
                     x = cs)) +
  labs(title = "Weekly Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

US

Complete data set

CDC designates a date to which they consider the data "good". Reported data after that date is considered incomplete due to delays in reporting. This intial set of plots uses the complete data set

```
epi <- ggplot(data = cdc)
```

```
epi + geom_point(aes(x = Date,
               y = `Number of new cases`))+
     geom_smooth(aes(x = Date,
               y = `Number of new cases`),
               color = "red",
               fill = "blue") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the CDC")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
epi + geom_col(aes(x=Date,
                   y=`Number of new cases`),
               fill= "blue") +
     geom_smooth(aes(x=Date,
                   y=`Number of new cases`),
                 color = "red") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the CDC")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
epi + geom_point(aes(x = Date,
               y = cum))+
    # geom_line(aes(x = Date,
```

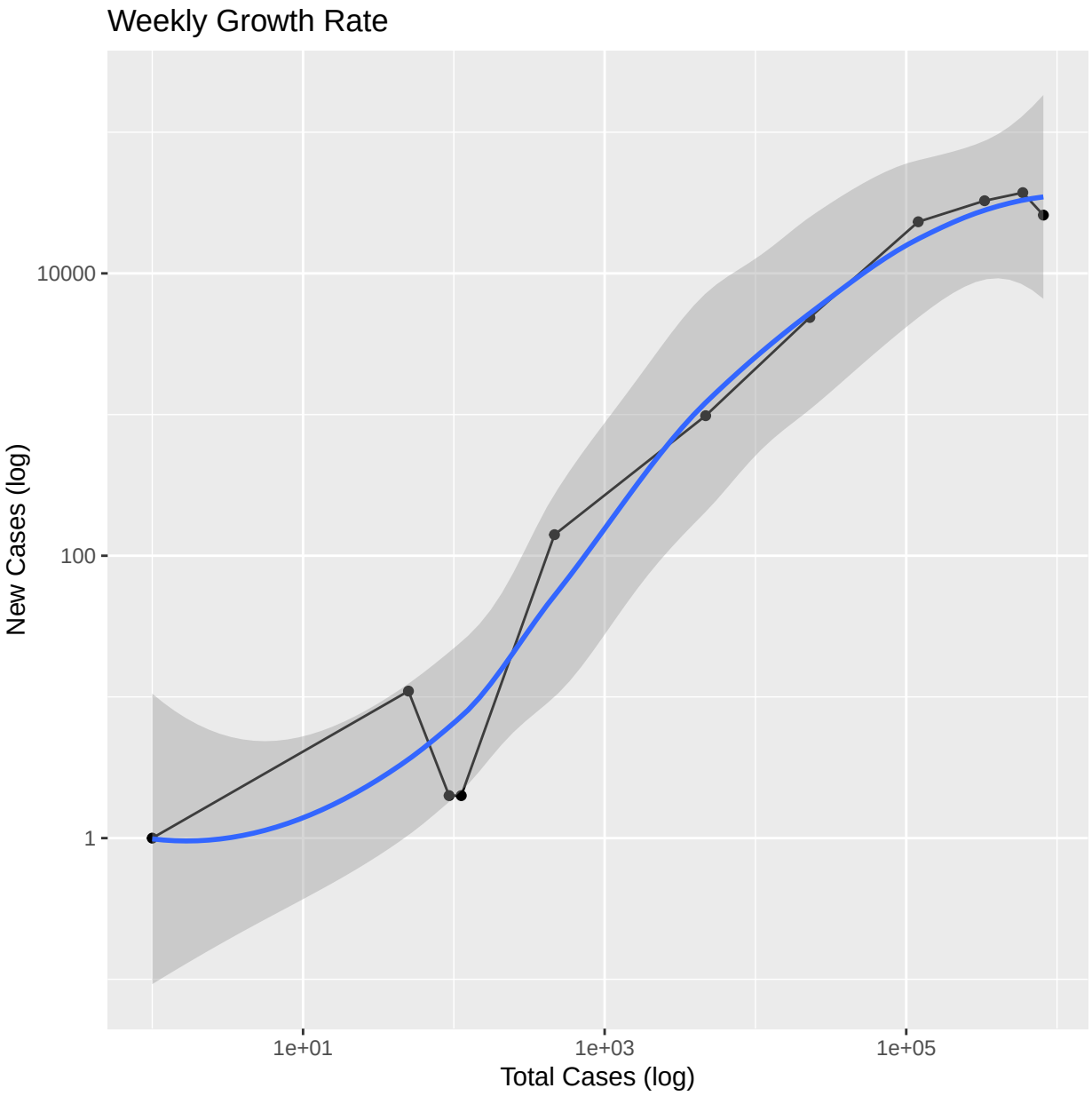

Figure 10: stable weekly growth rate

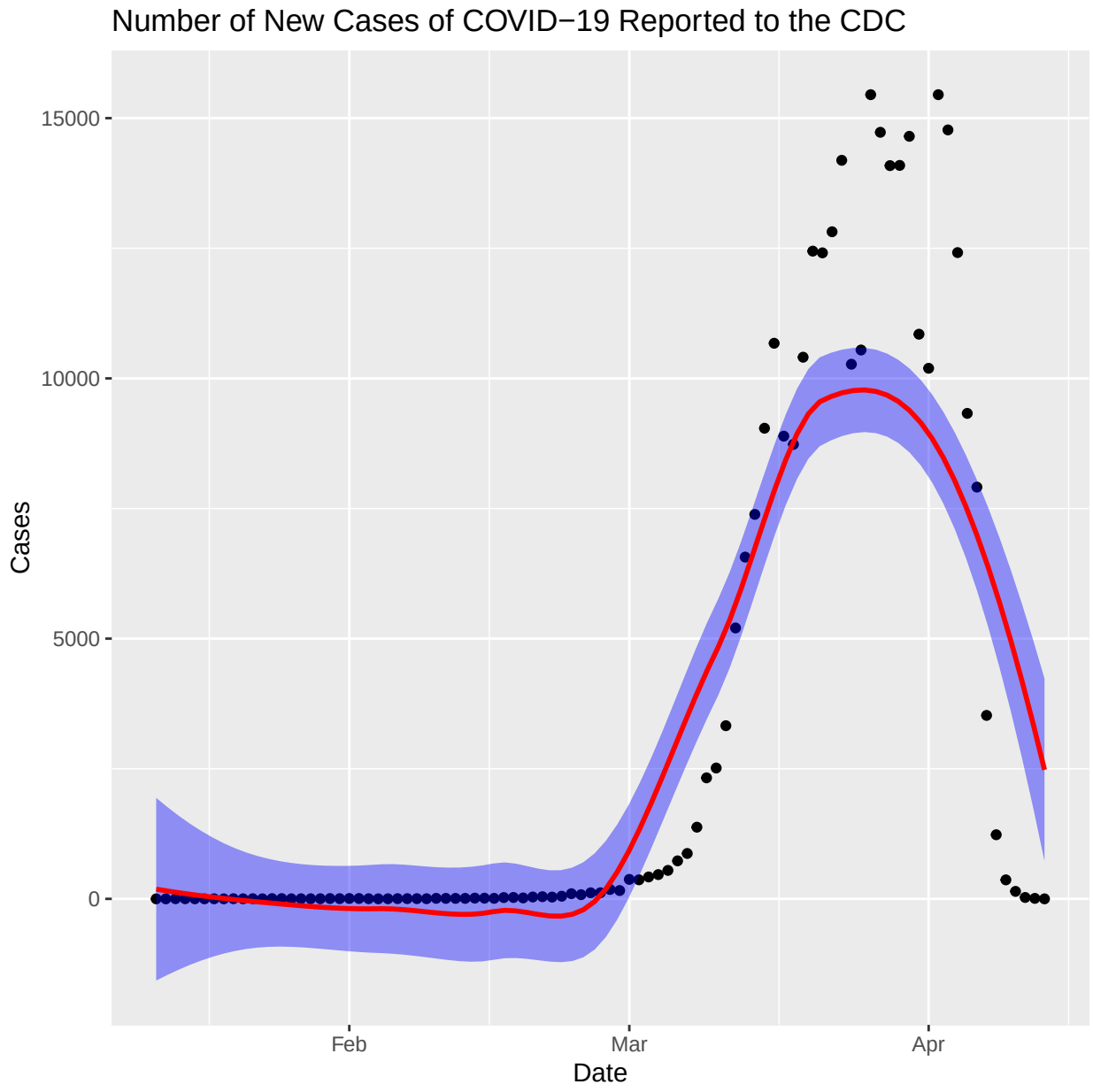

Figure 11: Epi curve 1

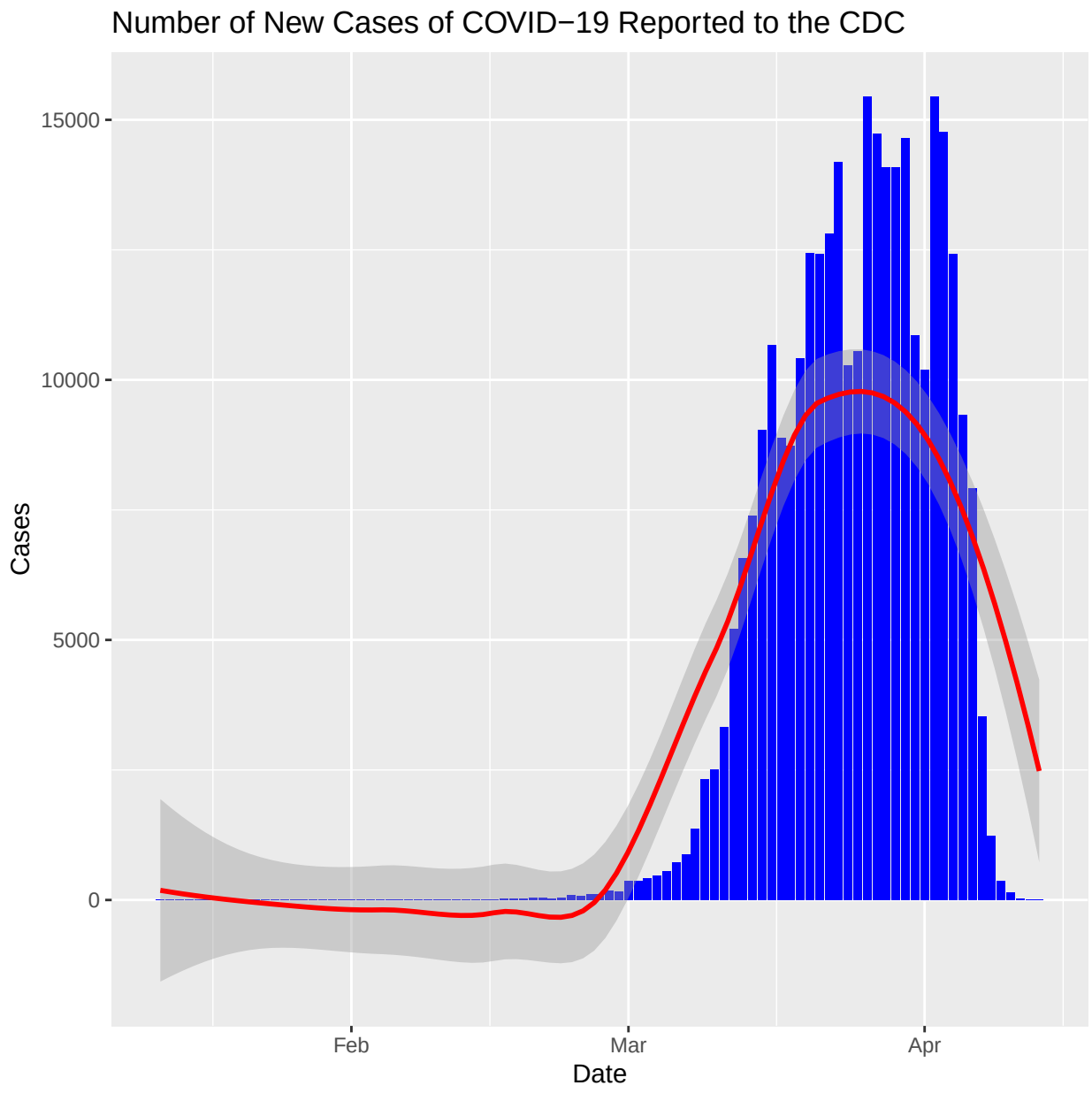

Figure 12: Epi curve 2, traditional

```
            #y = `cum`)) +
  geom_smooth(aes(x = Date,
            y = `cum`),
            color = "red",
            fill = "blue") +
  labs(y = "Cumulative number of cases",
       title = "Cumulative Number of Cases of COVID-19 Reported to the CDC")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

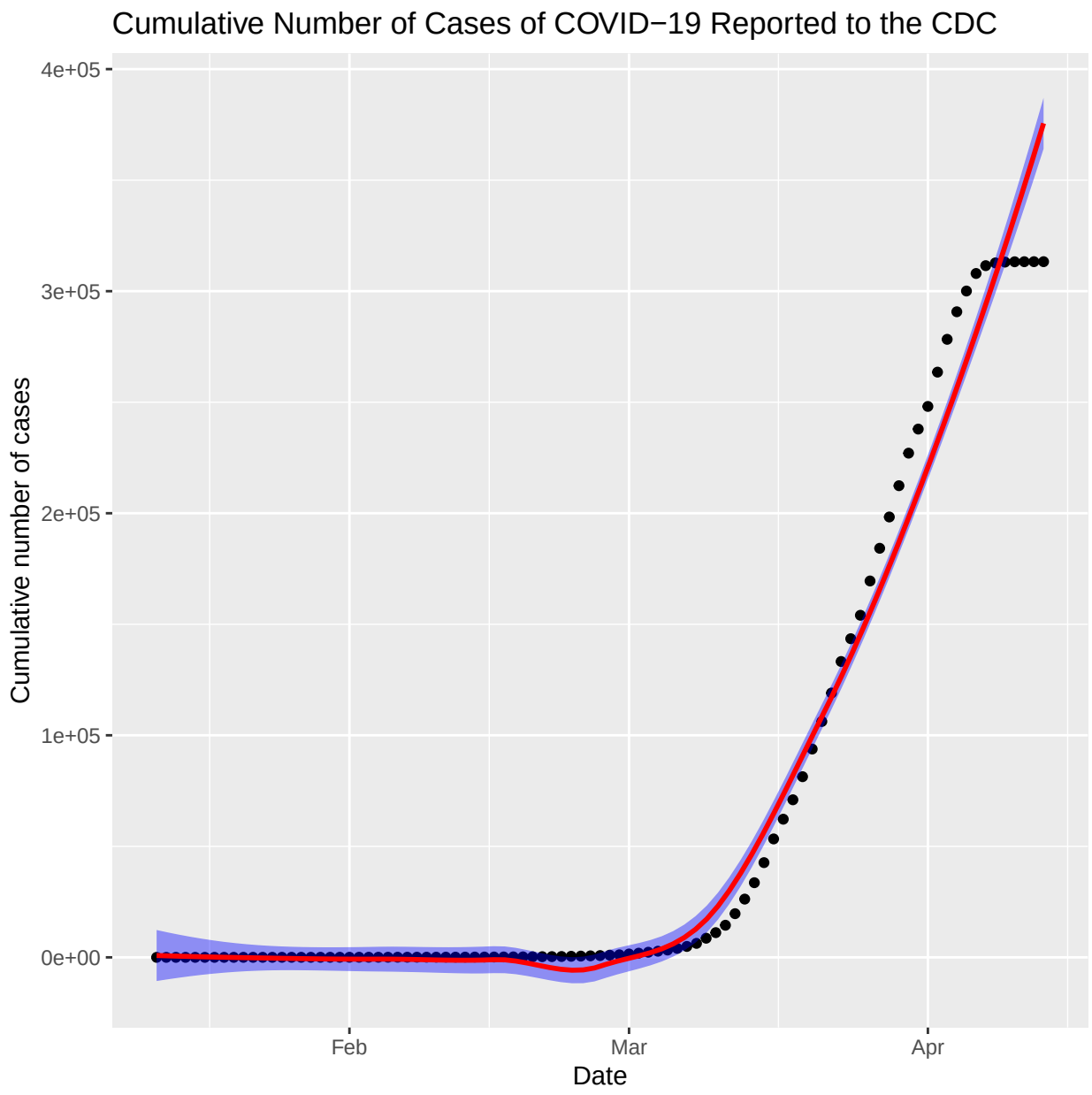

Figure 13: Cumulative cases

```
gr <- cdc %>%
  filter(`Number of new cases` != 0)

ggplot(data = gr) +
  geom_line(aes(y = `Number of new cases`,
                x = cum)) +
  scale_x_log10() +
  scale_y_log10() +
  geom_smooth(aes(y = `Number of new cases`,
                x = cum)) +
  labs(title = "Daily Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
gr <- cdc %>%
     group_by(`Week Number`) %>%
     summarise(cases = sum(`Number of new cases`),
               cs = sum(cum),
               count = n())

ggplot(data = gr) +
     geom_line(aes(y = `cases`,
                   x = cs)) +
     geom_point(aes(y = `cases`,
                    x = cs)) +
     scale_x_log10() +
     scale_y_log10() +
     geom_smooth(aes(y = `cases`,
                     x = cs)) +
  labs(title = "Weekly Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Filtered to use only "complete data"

Remove dates on or after 3 April as this data may not be completely reported

```
cdc <- cdc %>%                                                    # update with g
     filter(Date < as.Date("2020-04-03")) %>%
  filter(`Number of new cases` != 0)
```

```
epi <- ggplot(data = cdc)
```

```
epi + geom_point(aes(x = Date,
               y = `Number of new cases`))+
     geom_smooth(aes(x = Date,
               y = `Number of new cases`),
               color = "red",
               fill = "blue") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the CDC")
```

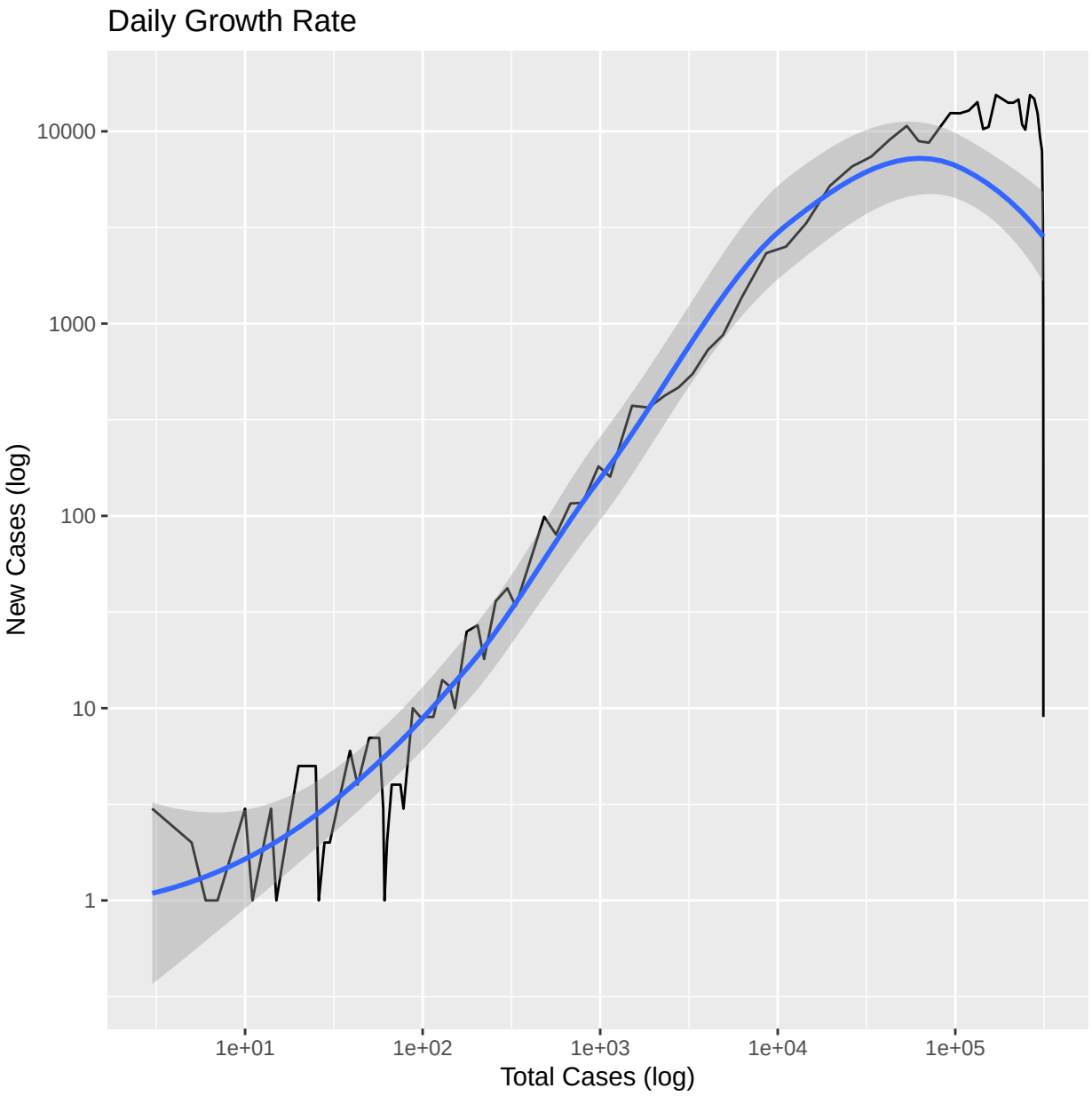

Figure 14: Growth Rate

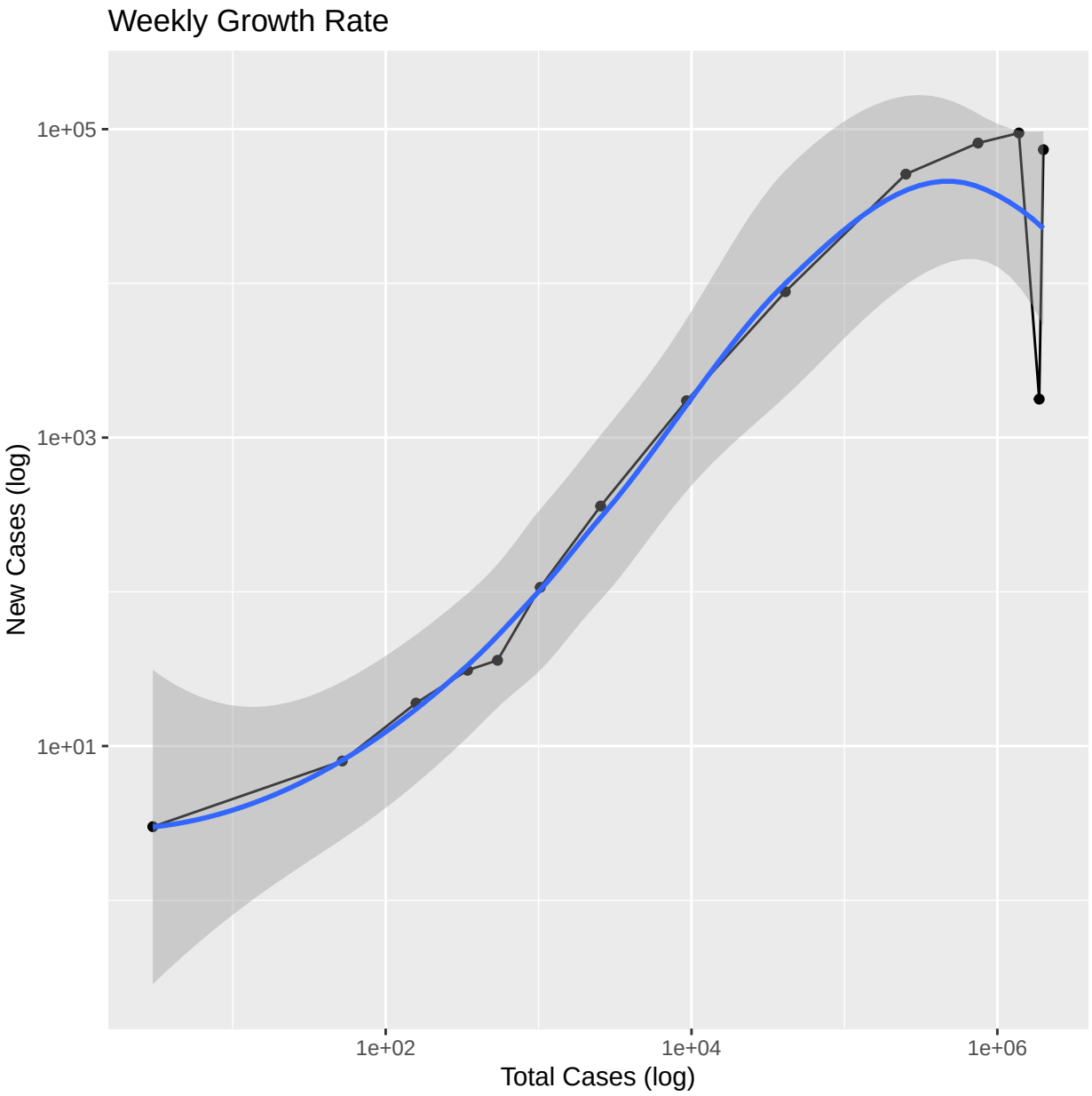

Figure 15: weekly growth rate

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

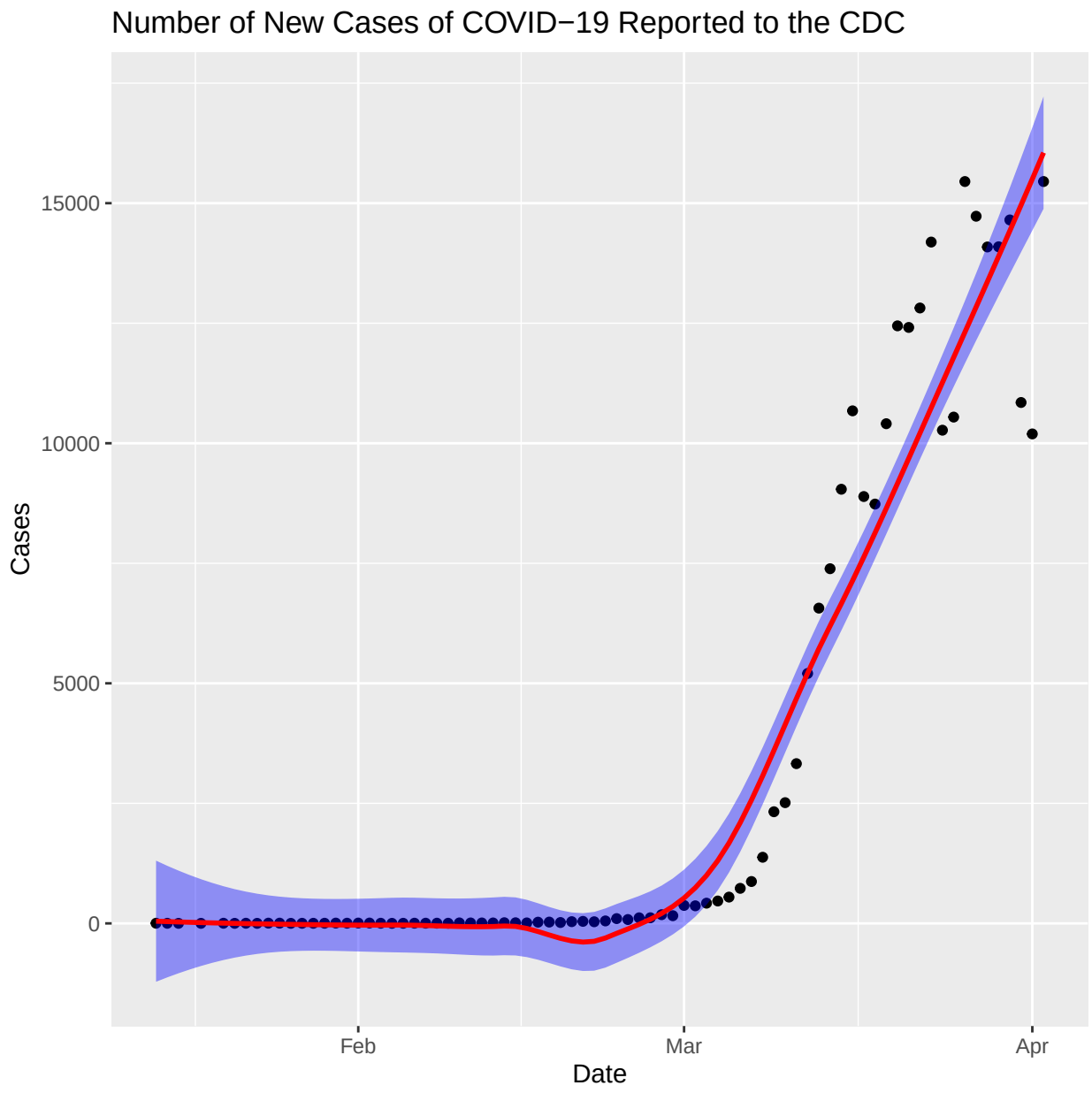

Figure 16: stable Epi curve

```
epi + geom_col(aes(x=Date,
                   y=`Number of new cases`),
               fill= "blue") +
     geom_smooth(aes(x=Date,
                   y=`Number of new cases`),
                 color = "red") +
     labs(y = "Cases",
          title = "Number of New Cases of COVID-19 Reported to the CDC")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

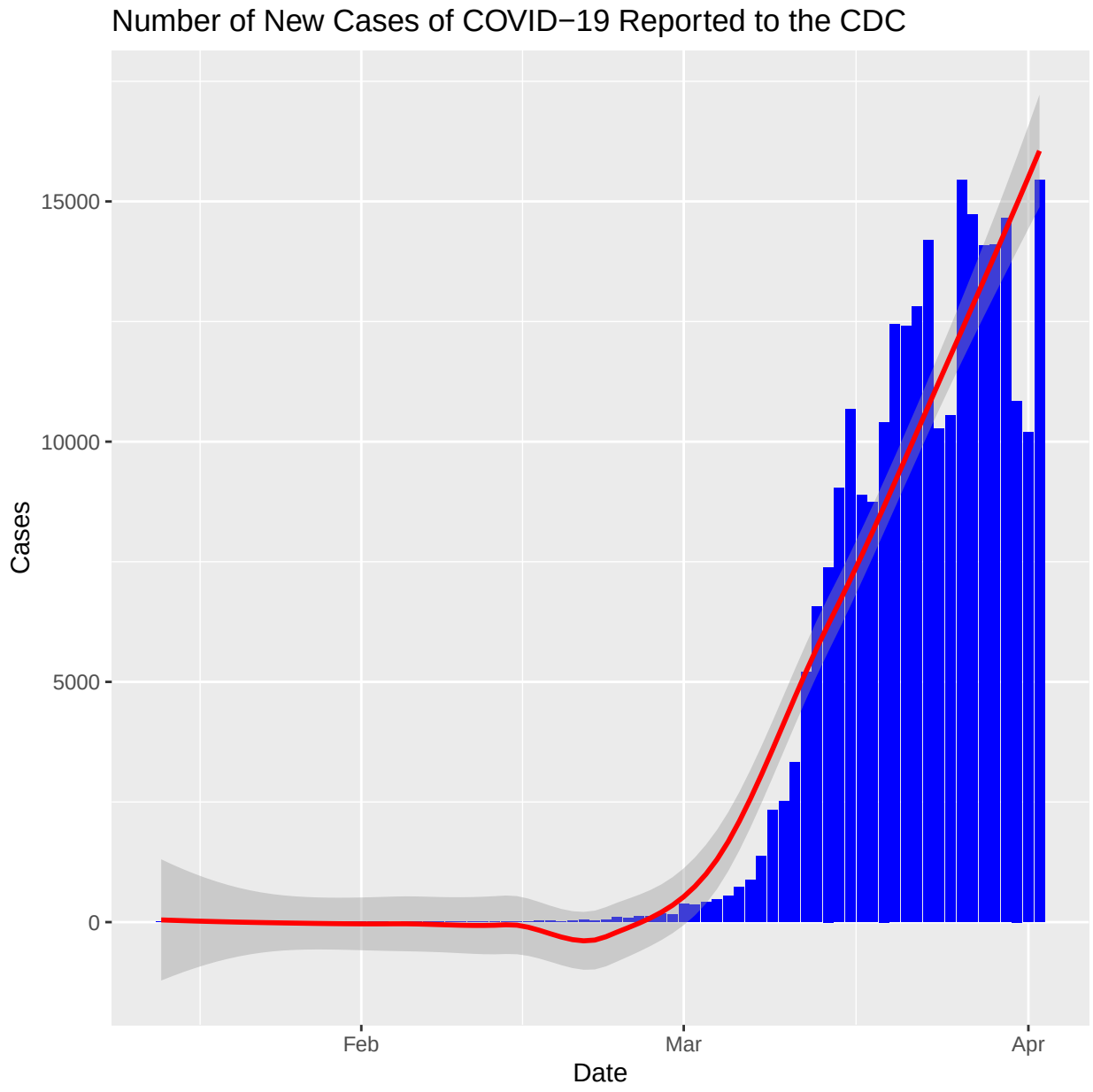

Figure 17: stable Epi curve, traditional

```
epi + geom_point(aes(x = Date,
                     y = cum))+
  geom_smooth(aes(x = Date,
                  y = `cum`),
              color = "red",
              fill = "blue") +
  labs(y = "Cumulative number of cases",
       title = "Cumulative Number of Cases of COVID-19 Reported to the CDC") +
  geom_hline(yintercept = mean(cdc$cum)) +
  geom_hline(yintercept = median(cdc$cum),
             lty = 2)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
gr <- cdc %>%
  filter(`Number of new cases` != 0)

ggplot(data = gr) +
  geom_line(aes(y = `Number of new cases`,
                x = cum)) +
  scale_x_log10() +
  scale_y_log10() +
  geom_smooth(aes(y = `Number of new cases`,
                  x = cum)) +
  labs(title = "Daily Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
gr <- cdc %>%
     group_by(`Week Number`) %>%
     summarise(cases = sum(`Number of new cases`),
               cs = sum(cum),
               count = n())

ggplot(data = gr) +
     geom_line(aes(y = `cases`,
                   x = cs)) +
     geom_point(aes(y = `cases`,
                    x = cs)) +
     scale_x_log10() +
     scale_y_log10() +
     geom_smooth(aes(y = `cases`,
                     x = cs)) +
  labs(title = "Weekly Growth Rate",
       x = "Total Cases (log)",
       y = "New Cases (log)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

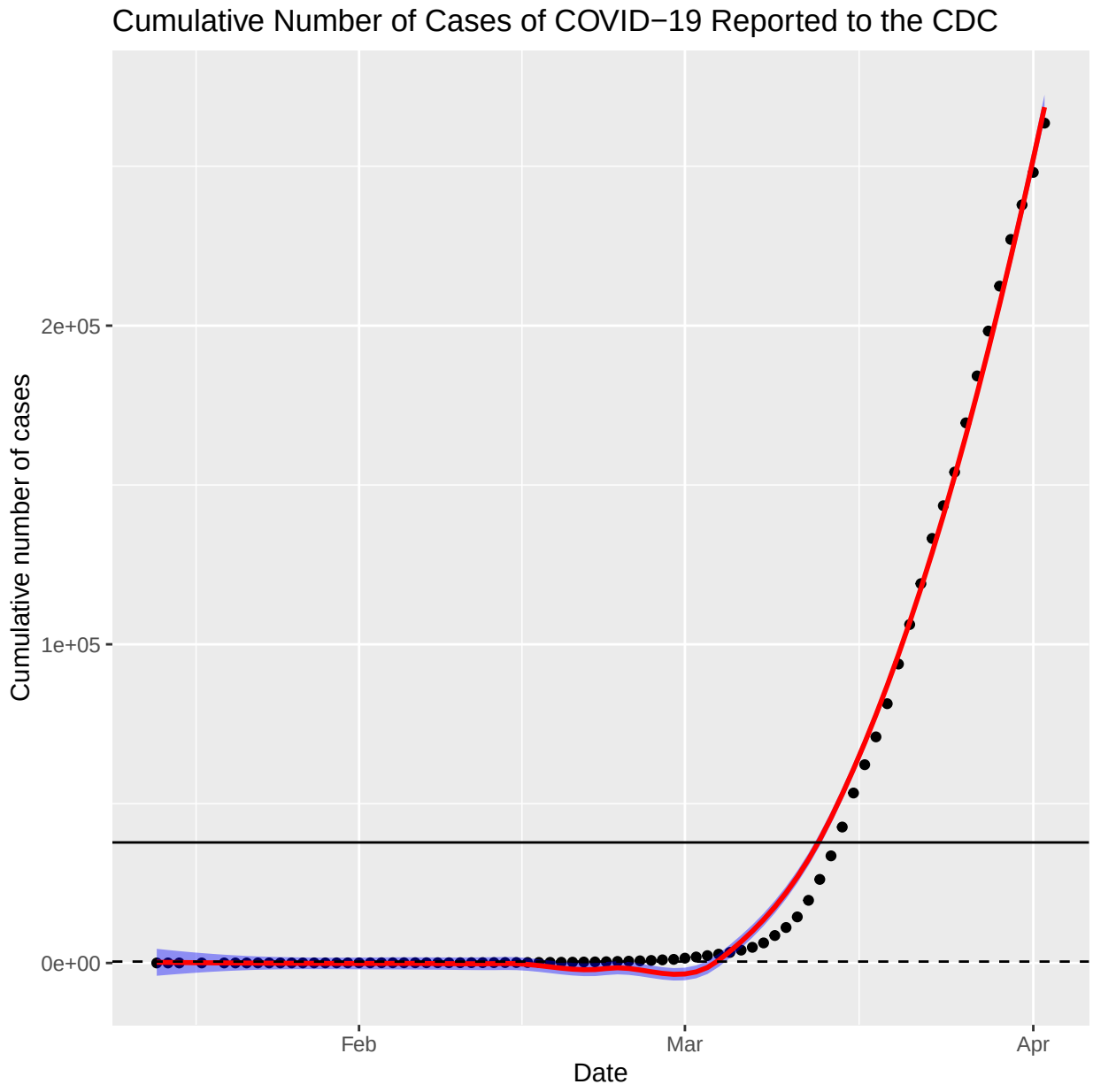

Figure 18: stable Cumulative cases

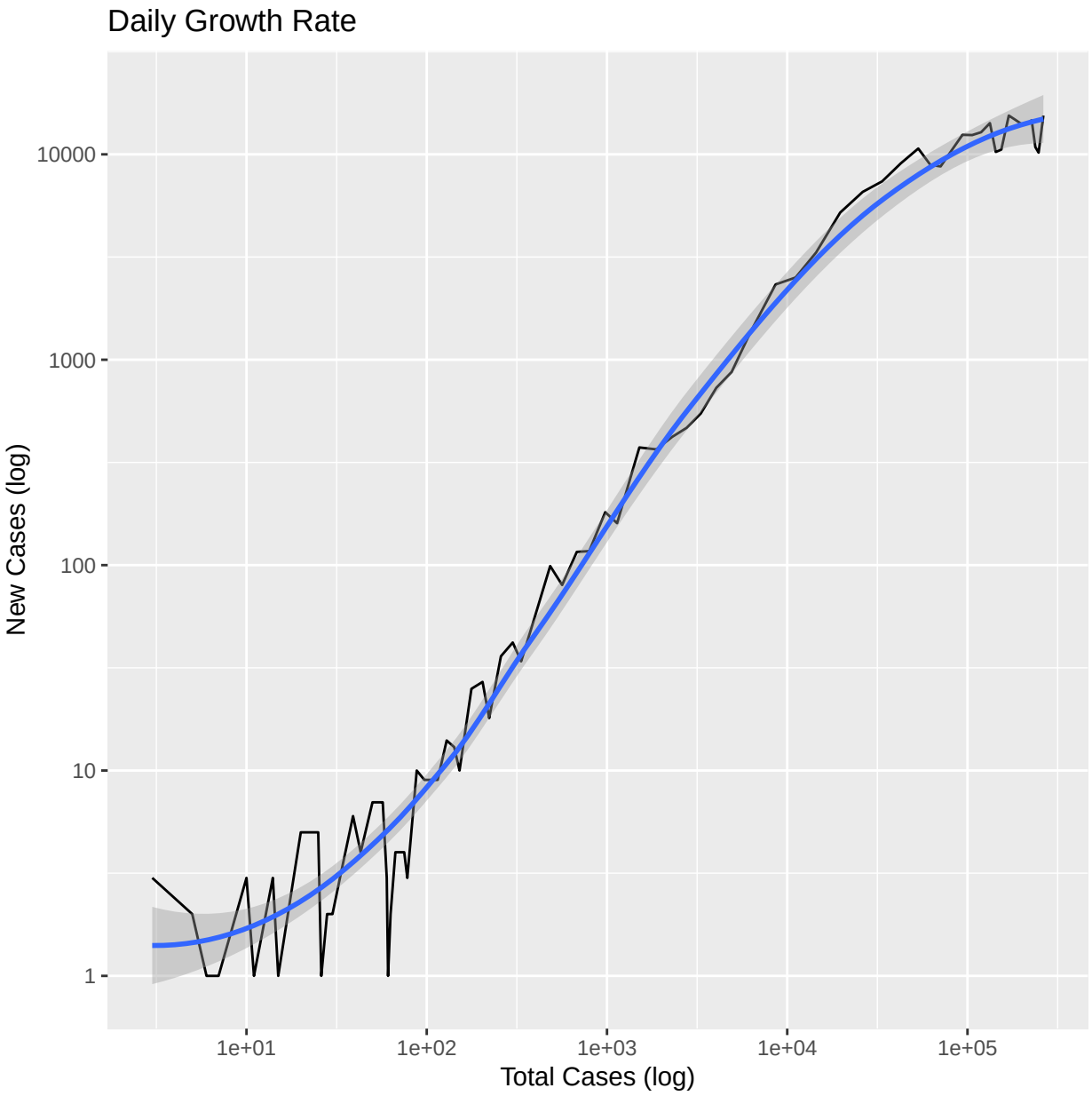

Figure 19: stable Growth Rate

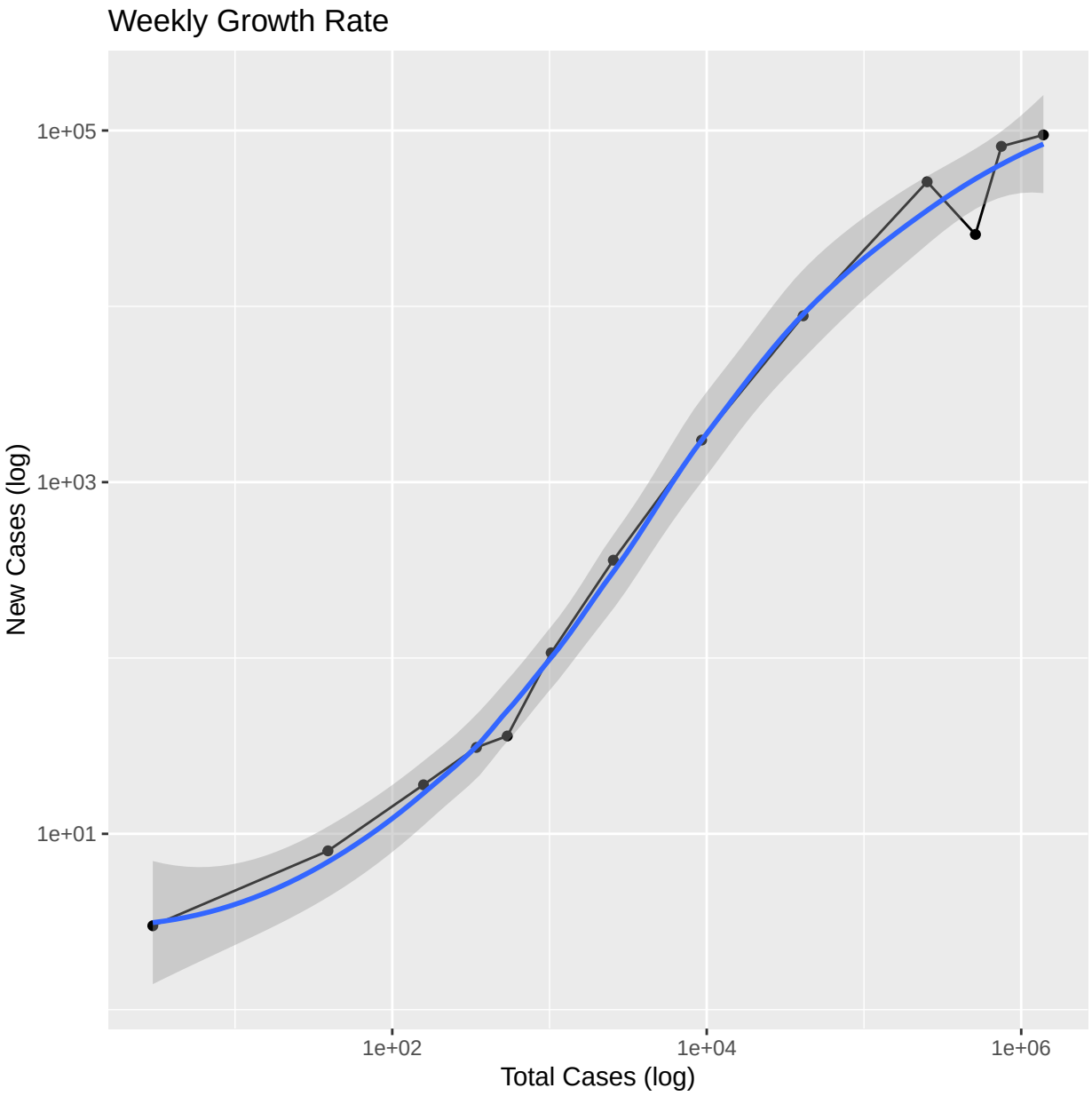

Figure 20: stable weekly growth rate