

EPI Info CDC

Nick Lauerman

3/16/2020

Contents

Libraries Used	1
Load CDC data	2
Read CSV File	2
Clean data and calculate cumulative number of cases	2
Visualize all data	2
Filter to remove incomplete reporting	7
Visualize	7

Libraries 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 -----  
## 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
```

Load CDC data

Read CSV File

```
cdc <- read_csv(file = "../data/CDC_data.csv")

## Warning: Missing column names filled in: 'X3' [3]
## Warning: Duplicated column names deduplicated: '25 mar data' => '25 mar
## data_1' [12]
```

Clean data and calculate cumulative number of cases

```
names(cdc)[1:2] <- c("Date",
                    "Number of new cases")
cdc$cum <- cumsum(cdc$`Number of new cases`)
cdc$Date <- as.Date(cdc$Date,
                  format = "%d-%b-%y")
cdc$`Week Number` <- week(cdc$Date)
```

Visualize all data

```
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,
                 #y = `cum`))+
  geom_smooth(aes(x = Date,
                 y = `cum`,
                 color = "red",
                 fill = "blue") +
  labs(y = "Cumulative number of cases",
```

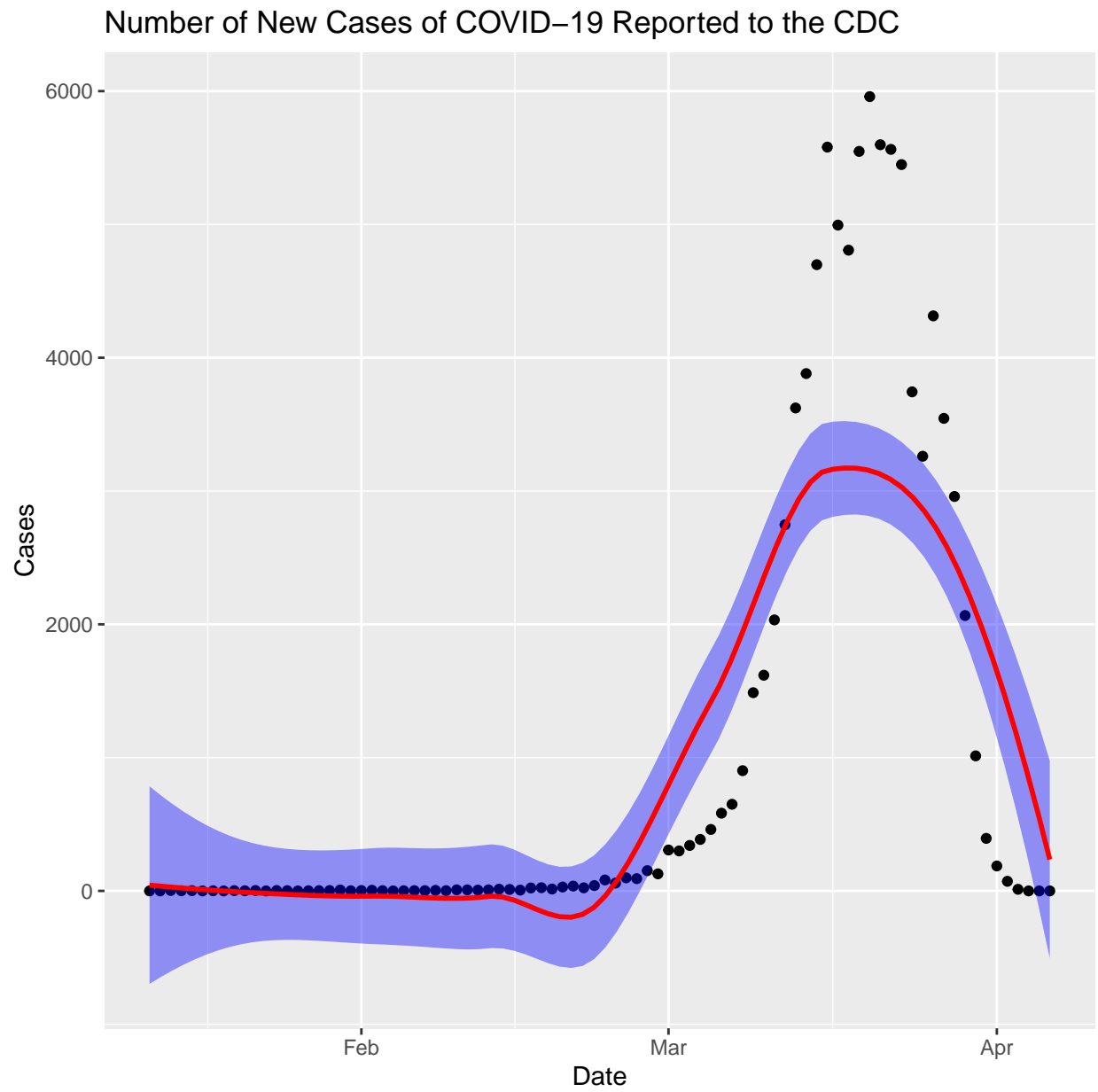


Figure 1: Epi curve 1

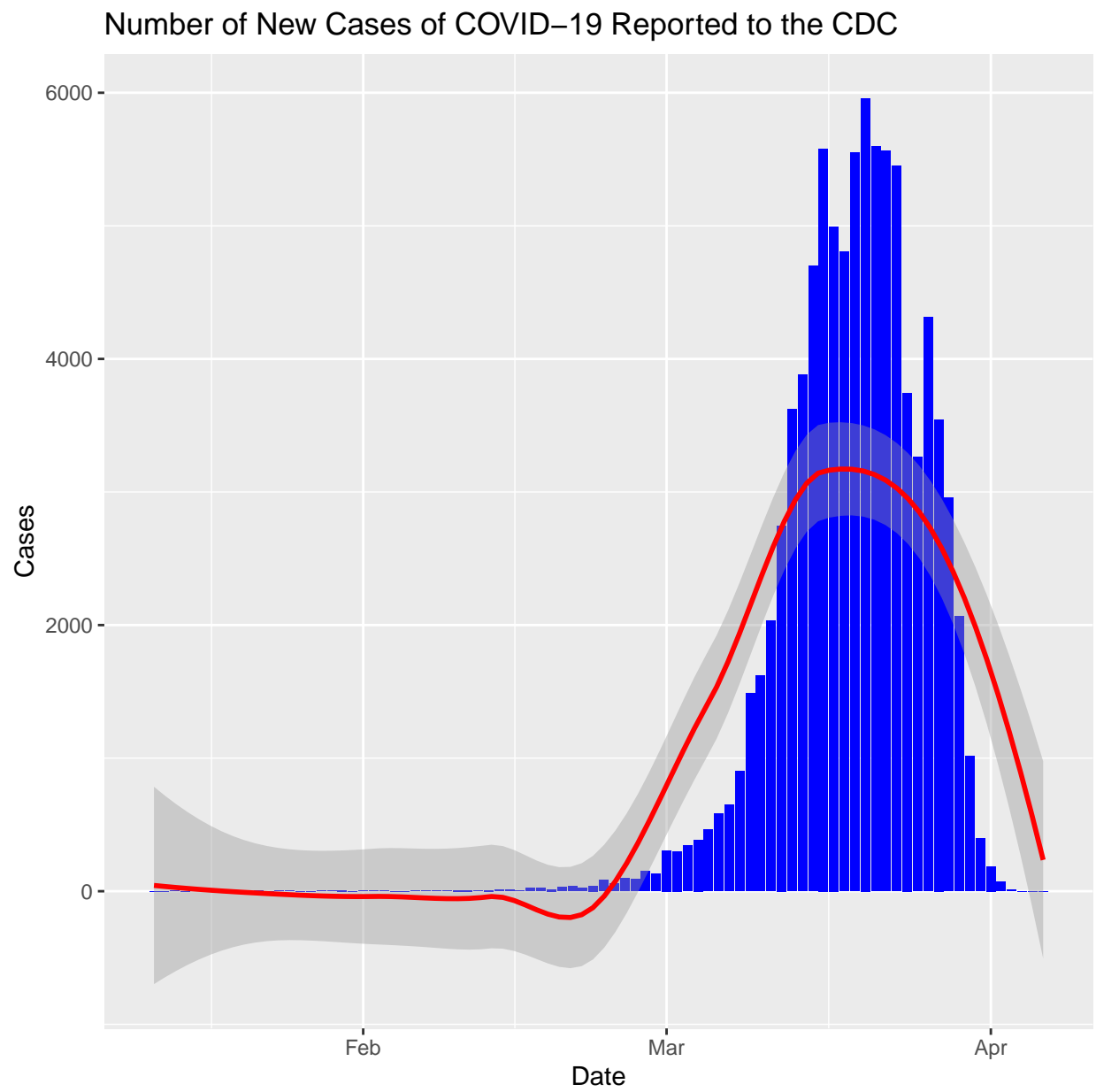


Figure 2: Epi curve 2, traditional

```

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'
```

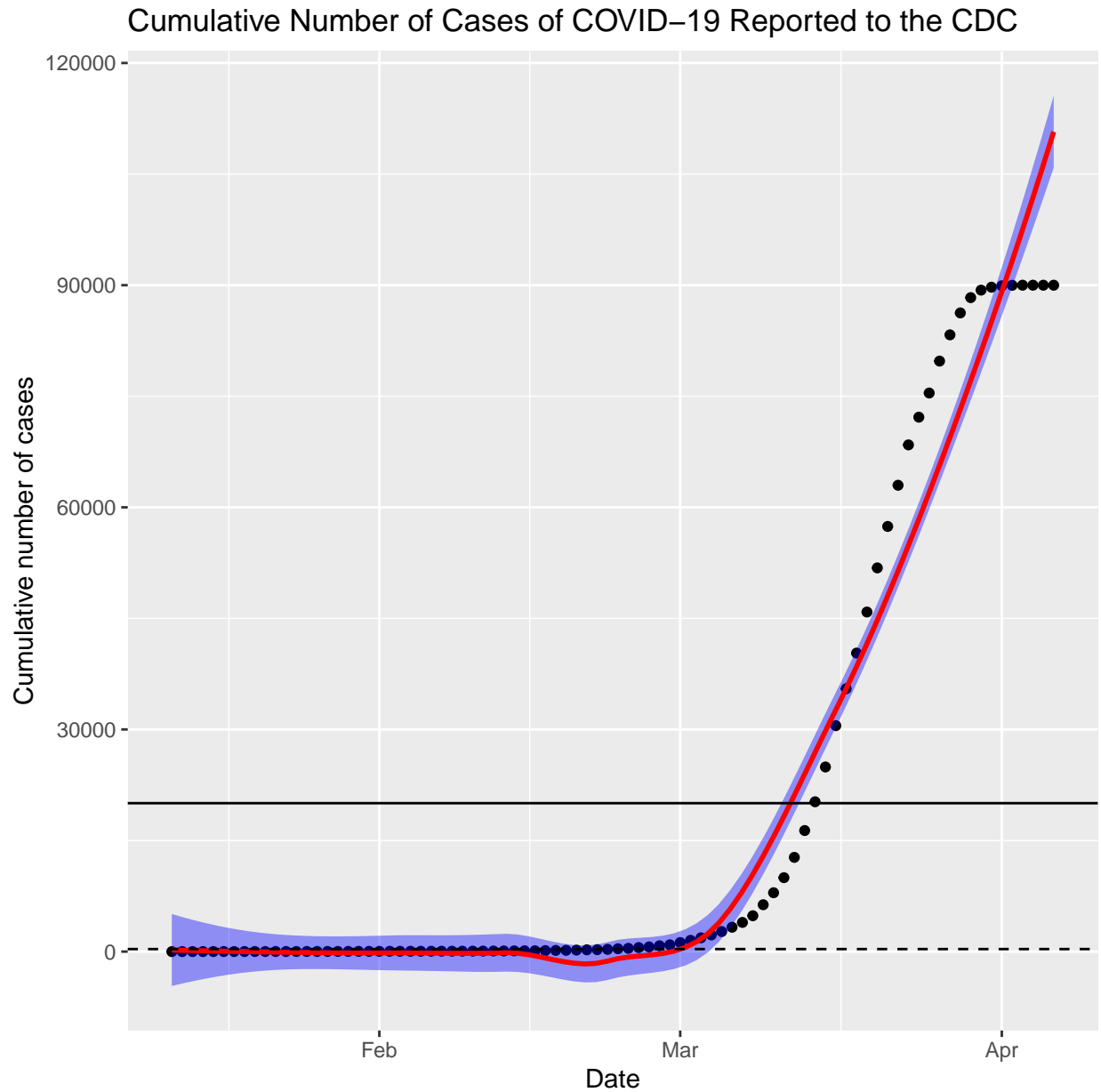


Figure 3: Cumulative cases

```

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'

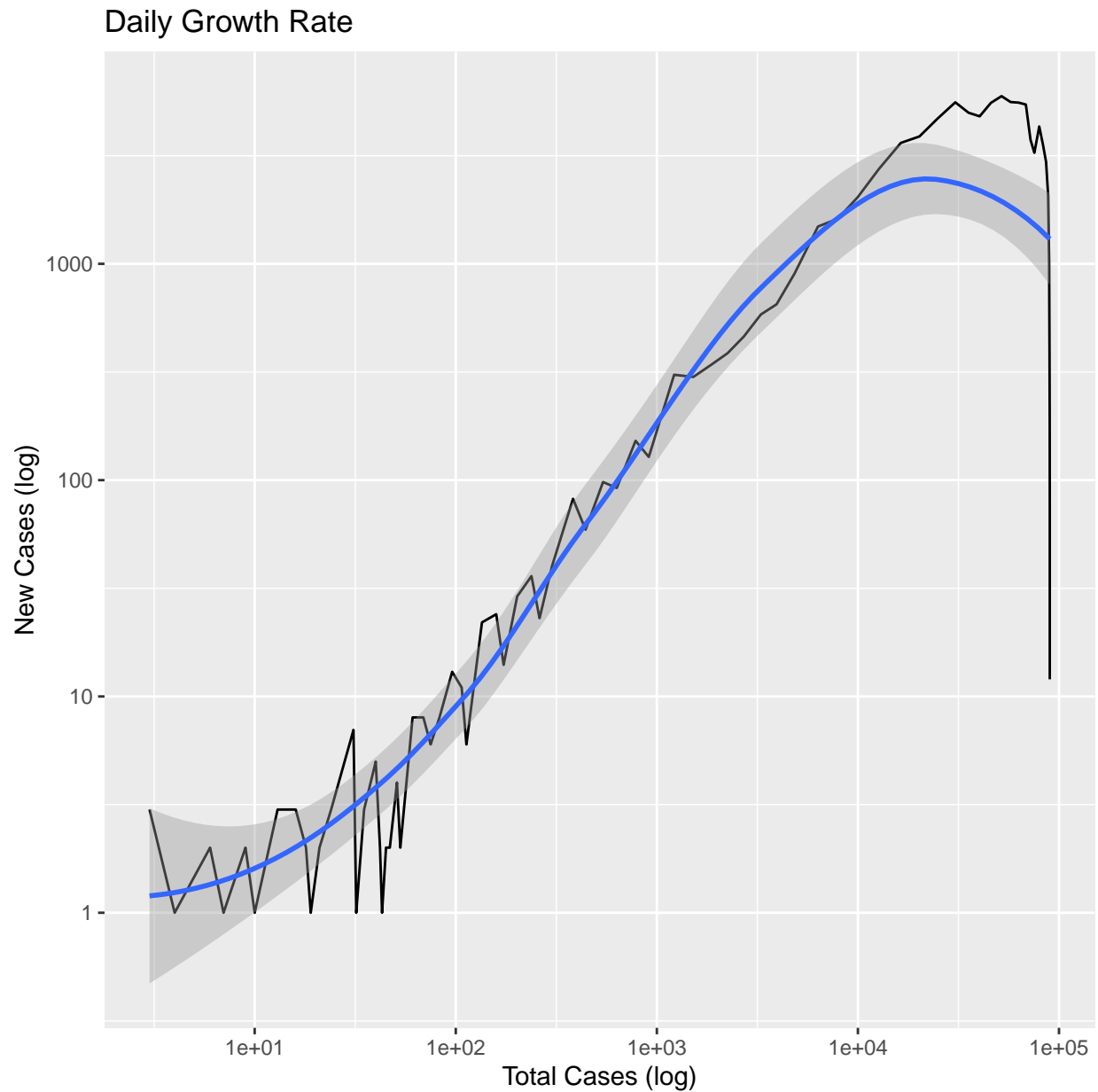


Figure 4: Growth Rate

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

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'

Filter to remove incomplete reporting

remove dates on or after 25 March as this data may not be completely reported

```
cdc <- cdc %>%
  filter(Date < as.Date("2020-03-25"))
```

update with g

Visualize

```
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))+
```

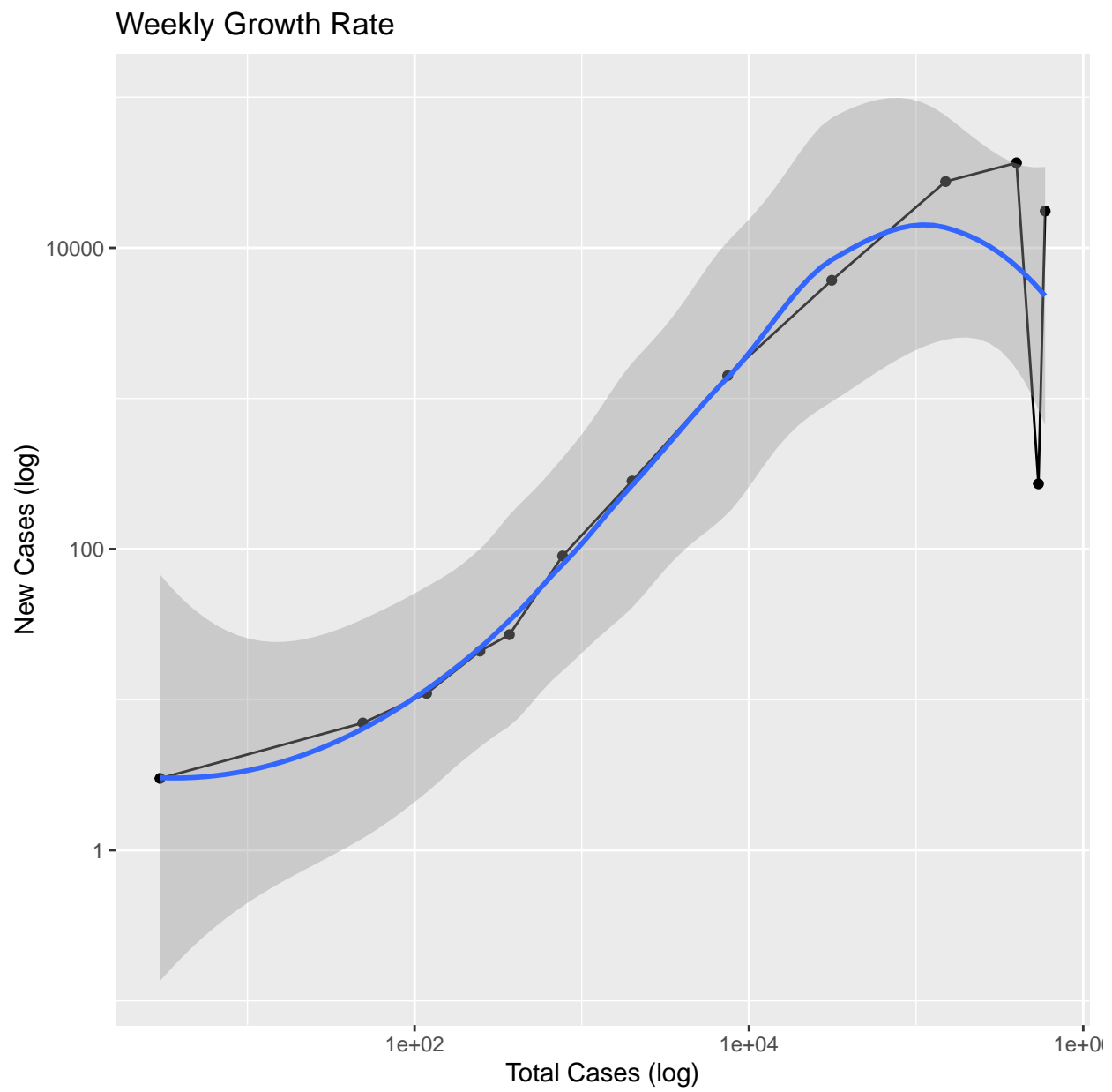


Figure 5: weekly growth rate

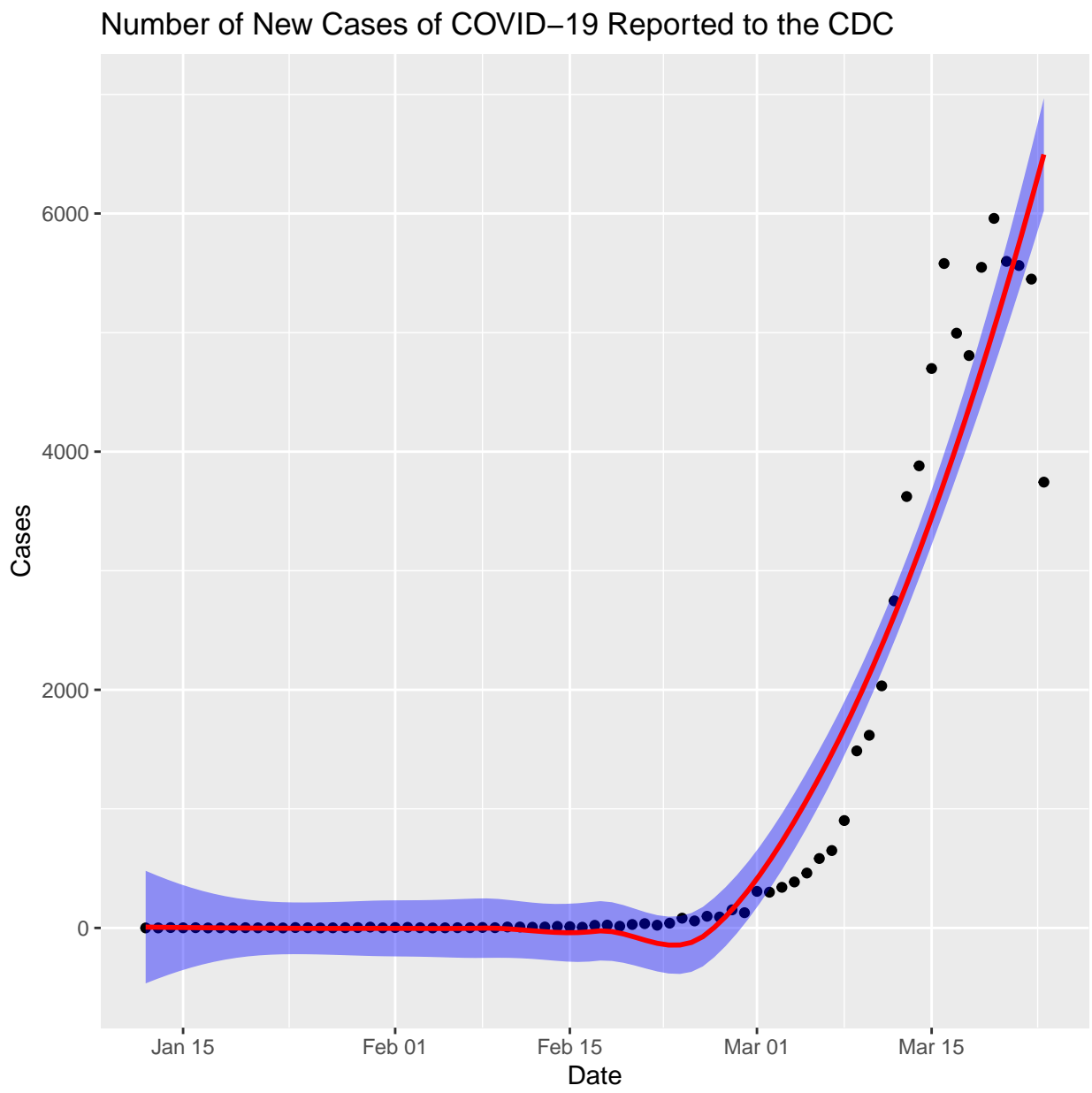


Figure 6: Epi curve 1

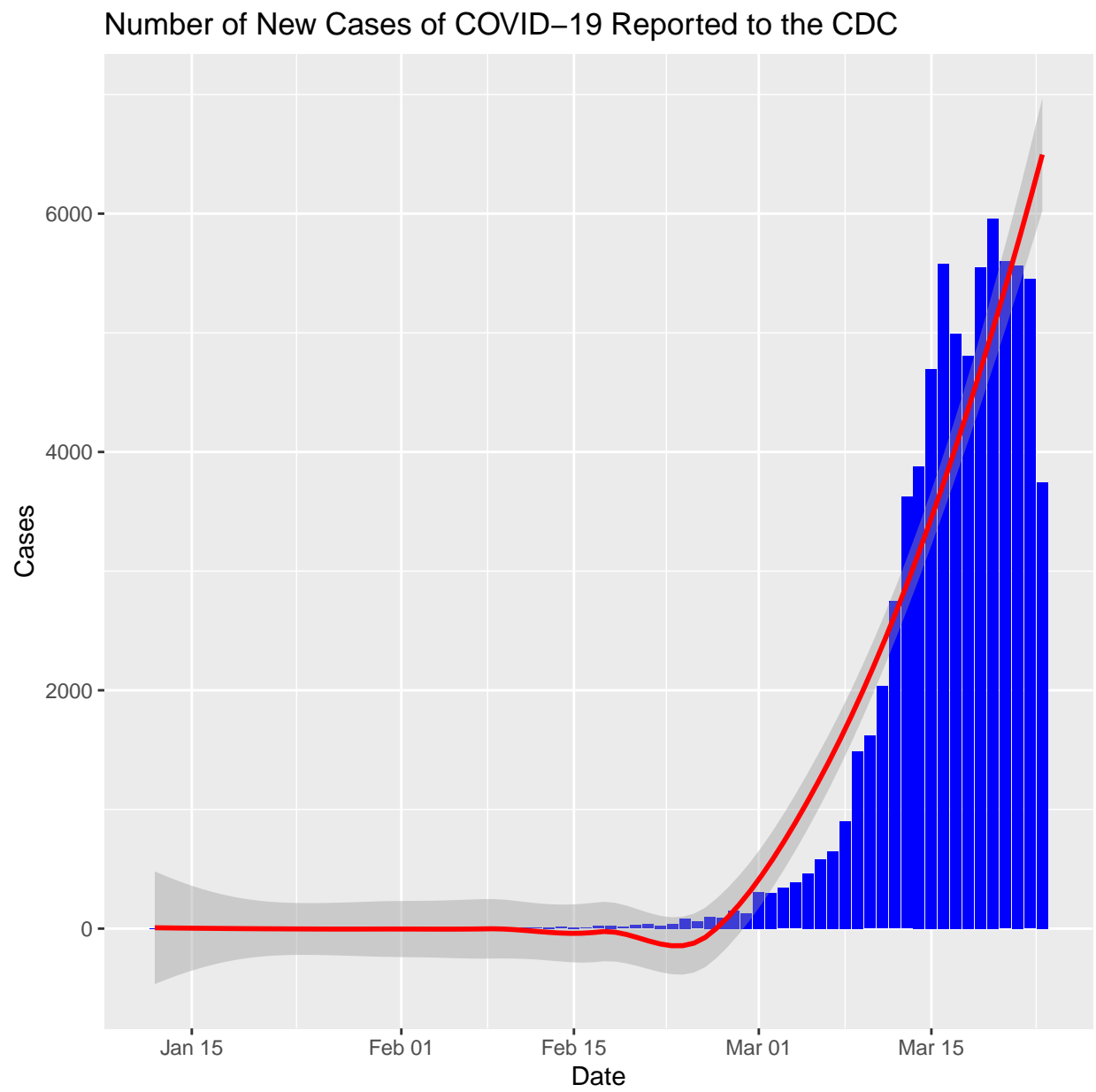


Figure 7: Epi curve 2, traditional

```
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'
```

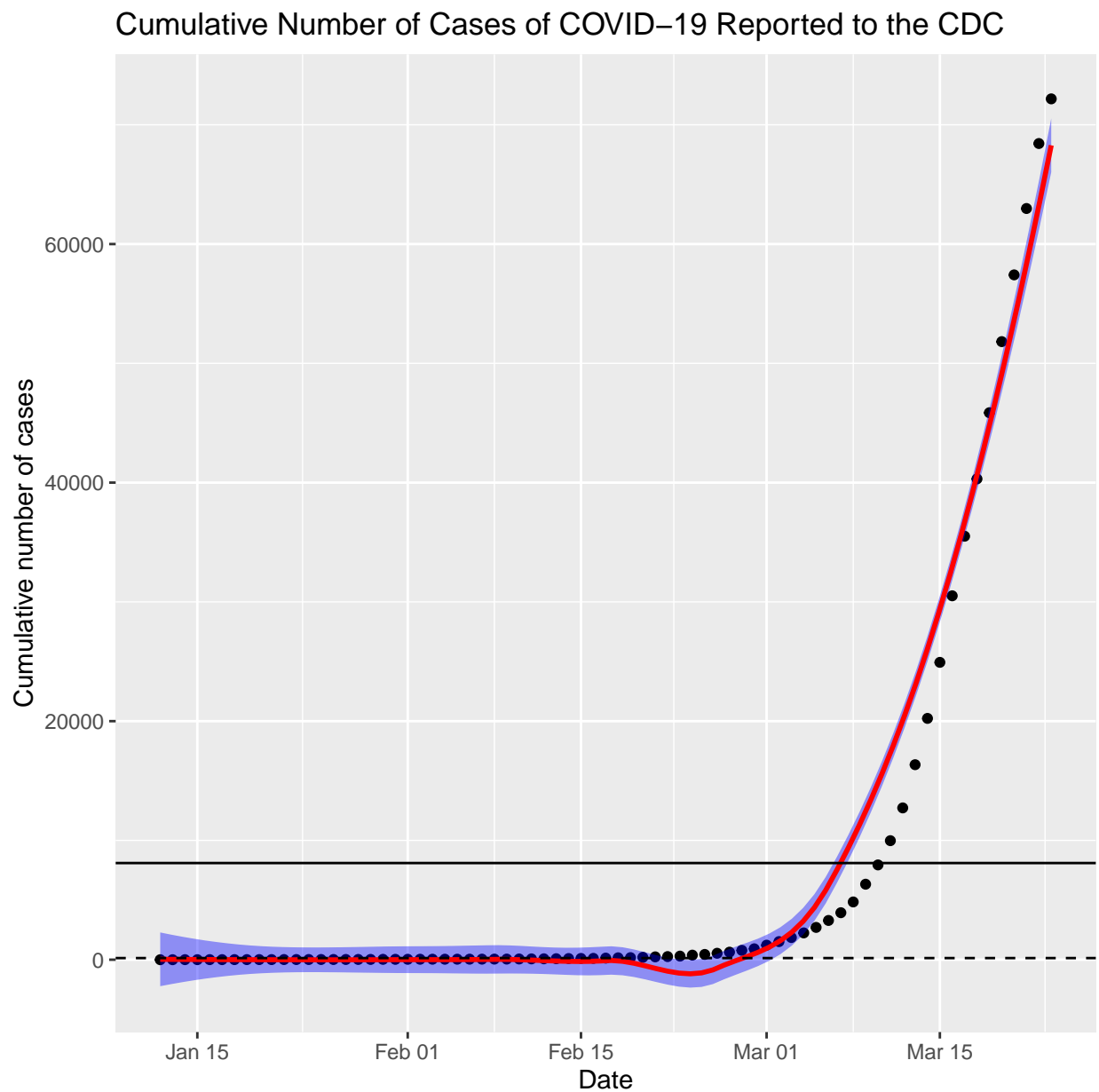


Figure 8: Cumulative cases

```

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))

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'

```

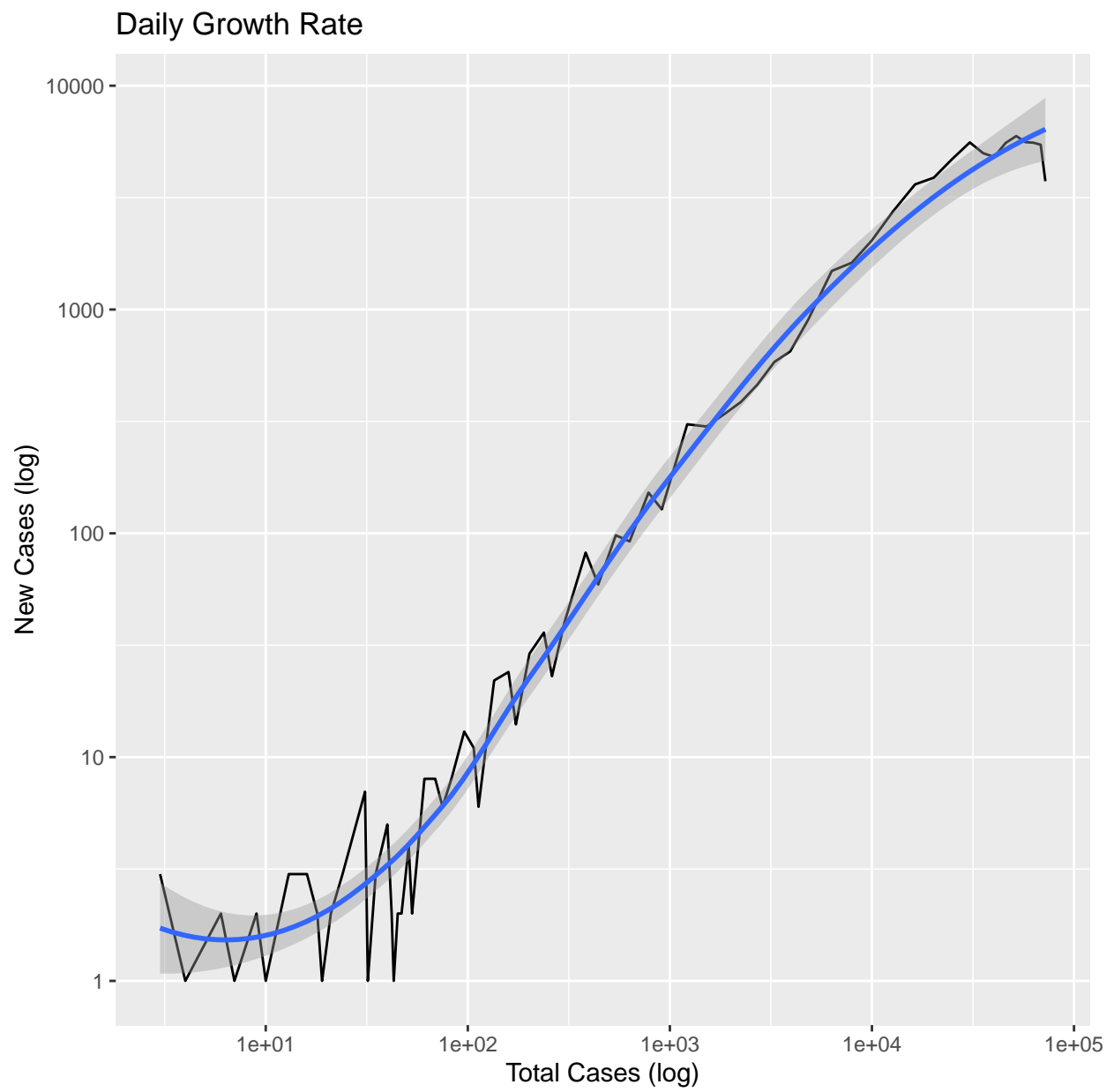


Figure 9: Growth Rate

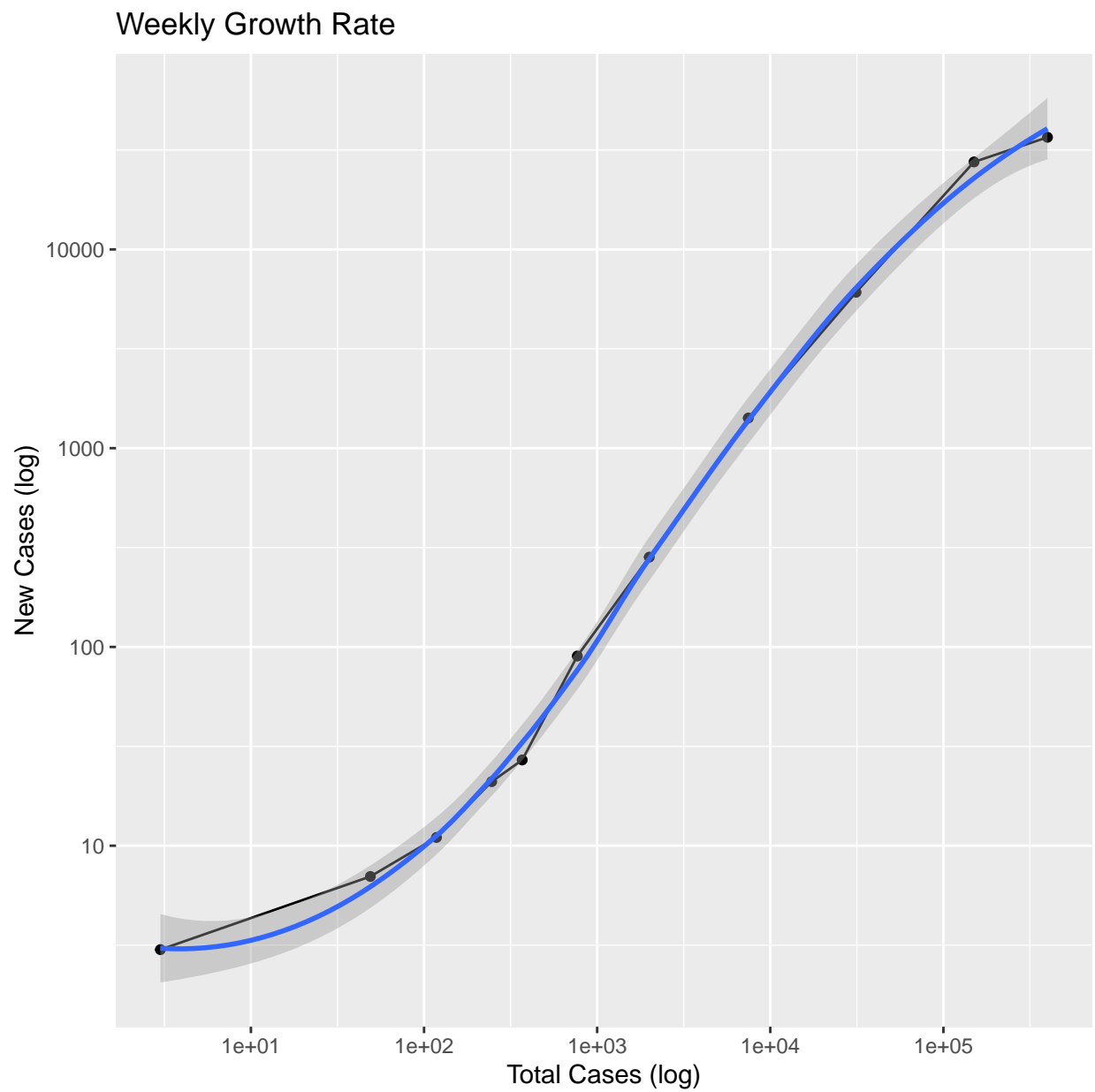


Figure 10: weekly growth rate