

# EPI Info CDC

Nick Lauerman

3/16/2020

## Contents

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

## Libraries Used

```
library(tidyverse)
```

```
## -- Attaching packages -----  
## v ggplot2 3.3.0      v purrr  0.3.3  
## v tibble  2.1.3      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 object is masked from 'package:base':  
##  
##     date
```

## 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]
## Parsed with column specification:
## cols(
##   Date = col_character(),
##   cases = col_double(),
##   X3 = col_logical(),
##   `16 mar data` = col_double(),
##   `17 mar data` = col_double(),
##   `18 mar data` = col_double(),
##   `19 mar data` = col_double(),
##   `20 mar data` = col_double(),
##   `23 mar data` = col_double(),
##   `24 mar data` = col_double(),
##   `25 mar data` = col_double(),
##   `25 mar data_1` = col_double()
## )
```

### 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")
```

## Data

```
cdc %>%
  data.frame

##      Date Number.of.new.cases X3 X16.mar.data X17.mar.data X18.mar.data
## 1 2020-01-12           0 NA           0           0           0
## 2 2020-01-13           0 NA           0           0           0
## 3 2020-01-14           2 NA           2           2           2
## 4 2020-01-15           0 NA           0           0           0
## 5 2020-01-16           1 NA           1           1           1
## 6 2020-01-17           0 NA           0           0           0
## 7 2020-01-18           0 NA           0           0           0
## 8 2020-01-19           0 NA           0           0           0
## 9 2020-01-20           1 NA           1           1           1
## 10 2020-01-21          1 NA           1           1           1
## 11 2020-01-22           2 NA           1           1           1
## 12 2020-01-23           0 NA           0           0           0
## 13 2020-01-24          1 NA           1           1           1
```

## 14	2020-01-25	3 NA	3	3	3
## 15	2020-01-26	0 NA	0	0	0
## 16	2020-01-27	0 NA	0	0	0
## 17	2020-01-28	3 NA	2	2	2
## 18	2020-01-29	1 NA	1	1	1
## 19	2020-01-30	2 NA	1	1	1
## 20	2020-01-31	0 NA	0	1	1
## 21	2020-02-01	1 NA	1	1	1
## 22	2020-02-02	4 NA	1	1	1
## 23	2020-02-03	0 NA	0	0	0
## 24	2020-02-04	0 NA	0	0	0
## 25	2020-02-05	0 NA	0	0	0
## 26	2020-02-06	2 NA	1	1	1
## 27	2020-02-07	1 NA	0	0	0
## 28	2020-02-08	1 NA	1	1	1
## 29	2020-02-09	0 NA	0	0	0
## 30	2020-02-10	4 NA	3	4	4
## 31	2020-02-11	6 NA	1	3	3
## 32	2020-02-12	0 NA	1	1	1
## 33	2020-02-13	4 NA	3	3	3
## 34	2020-02-14	3 NA	2	2	2
## 35	2020-02-15	7 NA	0	1	1
## 36	2020-02-16	3 NA	0	1	1
## 37	2020-02-17	13 NA	5	5	5
## 38	2020-02-18	14 NA	10	9	9
## 39	2020-02-19	7 NA	6	6	6
## 40	2020-02-20	12 NA	7	8	8
## 41	2020-02-21	21 NA	11	14	14
## 42	2020-02-22	20 NA	14	17	17
## 43	2020-02-23	23 NA	13	14	14
## 44	2020-02-24	48 NA	34	38	38
## 45	2020-02-25	40 NA	16	23	23
## 46	2020-02-26	74 NA	30	40	40
## 47	2020-02-27	71 NA	19	33	33
## 48	2020-02-28	102 NA	40	50	50
## 49	2020-02-29	97 NA	40	47	47
## 50	2020-03-01	173 NA	53	66	66
## 51	2020-03-02	188 NA	51	68	68
## 52	2020-03-03	216 NA	48	71	71
## 53	2020-03-04	247 NA	58	71	71
## 54	2020-03-05	274 NA	74	57	57
## 55	2020-03-06	365 NA	43	57	57
## 56	2020-03-07	385 NA	79	54	54
## 57	2020-03-08	548 NA	46	51	51
## 58	2020-03-09	782 NA	50	70	70
## 59	2020-03-10	767 NA	15	36	36
## 60	2020-03-11	993 NA	2	25	25
## 61	2020-03-12	1341 NA	0	4	4
## 62	2020-03-13	1641 NA	NA	2	2
## 63	2020-03-14	1634 NA	NA	0	0
## 64	2020-03-15	1905 NA	NA	0	0
## 65	2020-03-16	1880 NA	NA	NA	NA
## 66	2020-03-17	1246 NA	NA	NA	NA
## 67	2020-03-18	977 NA	NA	NA	NA

## 68	2020-03-19		896	NA	NA	NA	NA
## 69	2020-03-20		457	NA	NA	NA	NA
## 70	2020-03-21		179	NA	NA	NA	NA
## 71	2020-03-22		90	NA	NA	NA	NA
## 72	2020-03-23		59	NA	NA	NA	NA
## 73	2020-03-24		7	NA	NA	NA	NA
## 74	2020-03-25		1	NA	NA	NA	NA
## 75	2020-03-26		1	NA	NA	NA	NA
##	X19.mar.data	X20.mar.data	X23.mar.data	X24.mar.data	X25.mar.data		
## 1	0	0	0	0	0	0	
## 2	0	0	0	0	0	0	
## 3	2	2	2	2	2	2	
## 4	0	0	0	0	0	0	
## 5	1	1	1	1	1	1	
## 6	0	0	0	0	0	0	
## 7	0	0	0	0	0	0	
## 8	0	0	0	0	0	0	
## 9	1	1	1	1	1	1	
## 10	1	1	1	1	1	1	
## 11	1	1	1	1	1	2	
## 12	0	0	0	0	0	0	
## 13	1	1	1	1	1	1	
## 14	3	3	3	3	3	3	
## 15	0	0	0	0	0	0	
## 16	0	0	0	0	0	0	
## 17	3	3	3	3	3	3	
## 18	1	1	1	1	1	1	
## 19	1	1	1	1	1	1	
## 20	0	0	0	0	0	0	
## 21	1	1	1	1	1	1	
## 22	1	1	2	3	4		
## 23	0	0	0	0	0		
## 24	0	0	0	0	0		
## 25	0	0	0	0	0		
## 26	1	2	2	2	2		
## 27	0	0	0	0	1		
## 28	1	1	1	1	1		
## 29	0	0	0	0	0		
## 30	4	4	3	3	3		
## 31	5	5	6	6	6		
## 32	0	0	0	0	0		
## 33	4	4	4	4	4		
## 34	1	3	3	3	3		
## 35	5	5	6	6	7		
## 36	3	3	3	3	3		
## 37	9	9	12	12	13		
## 38	10	10	14	14	14		
## 39	6	6	7	7	8		
## 40	10	10	12	12	12		
## 41	18	20	23	23	23		
## 42	22	22	22	22	23		
## 43	18	18	20	21	23		
## 44	42	43	50	50	50		
## 45	33	34	35	37	39		

## 46	52	52	61	64	73
## 47	43	47	55	58	68
## 48	66	66	82	87	98
## 49	55	57	64	70	91
## 50	97	103	113	125	165
## 51	91	95	120	147	180
## 52	108	121	146	179	213
## 53	117	125	159	183	235
## 54	111	119	159	204	257
## 55	124	139	193	249	331
## 56	104	120	176	280	349
## 57	121	140	214	379	496
## 58	163	194	307	559	692
## 59	140	172	302	523	659
## 60	123	174	331	628	791
## 61	75	122	275	741	1048
## 62	54	95	269	802	1105
## 63	23	55	238	643	1051
## 64	13	35	203	455	991
## 65	2	12	173	369	916
## 66	0	8	90	181	455
## 67	0	0	44	82	281
## 68	NA	0	21	41	180
## 69	NA	NA	2	12	52
## 70	NA	NA	NA	2	13
## 71	NA	NA	NA	1	5
## 72	NA	NA	NA	0	4
## 73	NA	NA	NA	NA	0
## 74	NA	NA	NA	NA	NA
## 75	NA	NA	NA	NA	NA
##	X25.mar.data_1	cum			
## 1	0	0			
## 2	0	0			
## 3	2	2			
## 4	0	2			
## 5	1	3			
## 6	0	3			
## 7	0	3			
## 8	0	3			
## 9	1	4			
## 10	1	5			
## 11	2	7			
## 12	0	7			
## 13	1	8			
## 14	3	11			
## 15	0	11			
## 16	0	11			
## 17	3	14			
## 18	1	15			
## 19	1	17			
## 20	0	17			
## 21	1	18			
## 22	4	22			
## 23	0	22			

## 24	0	22
## 25	0	22
## 26	2	24
## 27	1	25
## 28	1	26
## 29	0	26
## 30	3	30
## 31	6	36
## 32	0	36
## 33	4	40
## 34	3	43
## 35	7	50
## 36	3	53
## 37	13	66
## 38	14	80
## 39	8	87
## 40	12	99
## 41	23	120
## 42	23	140
## 43	23	163
## 44	50	211
## 45	39	251
## 46	73	325
## 47	69	396
## 48	99	498
## 49	91	595
## 50	166	768
## 51	180	956
## 52	214	1172
## 53	237	1419
## 54	257	1693
## 55	333	2058
## 56	352	2443
## 57	498	2991
## 58	696	3773
## 59	662	4540
## 60	792	5533
## 61	1055	6874
## 62	1113	8515
## 63	1058	10149
## 64	1001	12054
## 65	926	13934
## 66	467	15180
## 67	302	16157
## 68	187	17053
## 69	55	17510
## 70	13	17689
## 71	8	17779
## 72	5	17838
## 73	0	17845
## 74	0	17846
## 75	NA	17847

## Visualize all data

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

```
epi + geom_point(aes(x = Date,  
  y = `Number of new cases`))+  
#   geom_line(aes(x = Date,  
#               y = `Number of new cases`),  
#             linetype = 2) +  
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",  
  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'
```

## Filter to remove incomplete reporting

remove dates on or after 15 March as this data may not be completely reported

```
cdc <- cdc %>%  
  filter(Date < as.Date("2020-03-16"))
```

## Visualize

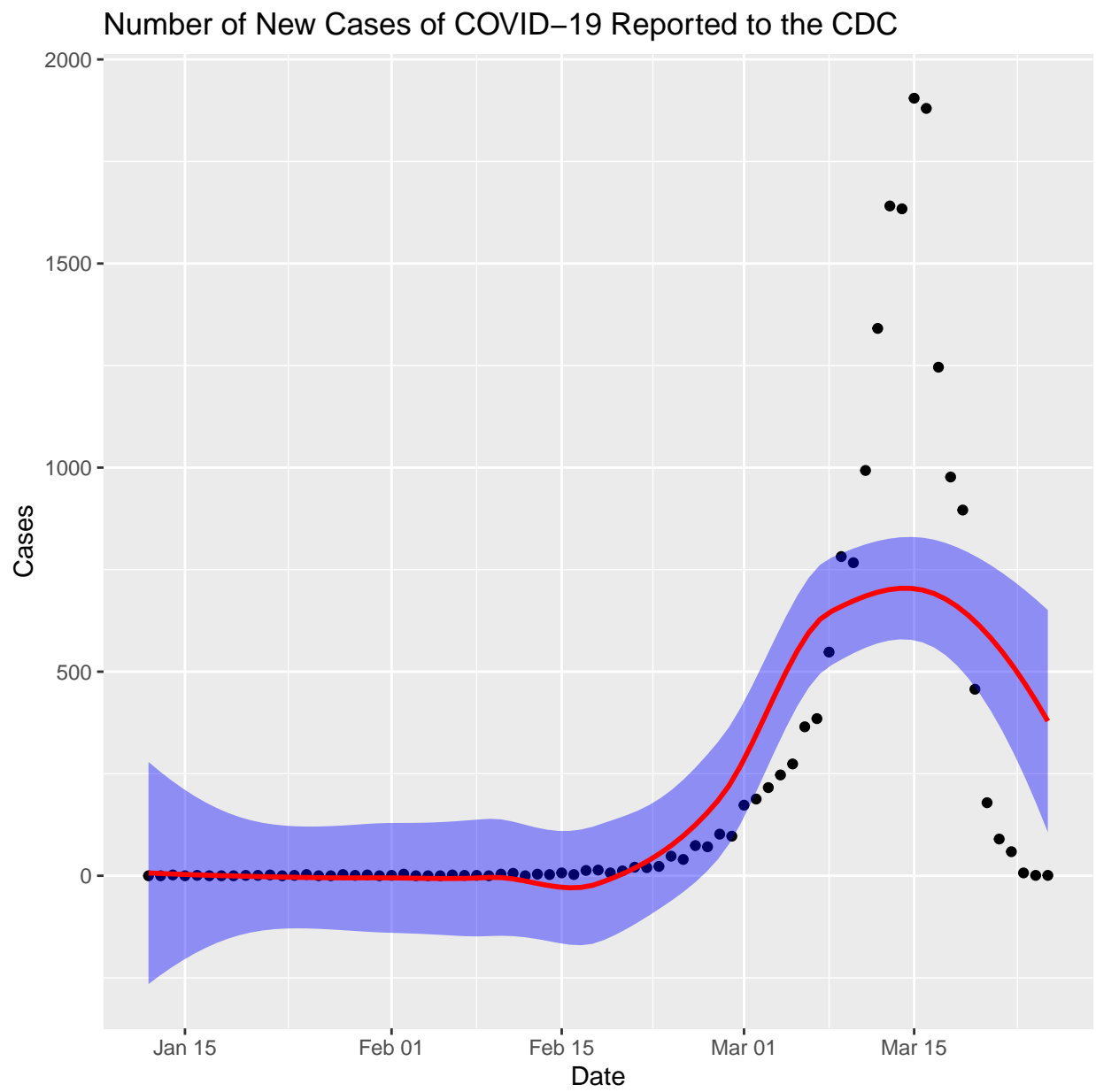


Figure 1: Epi curve 1



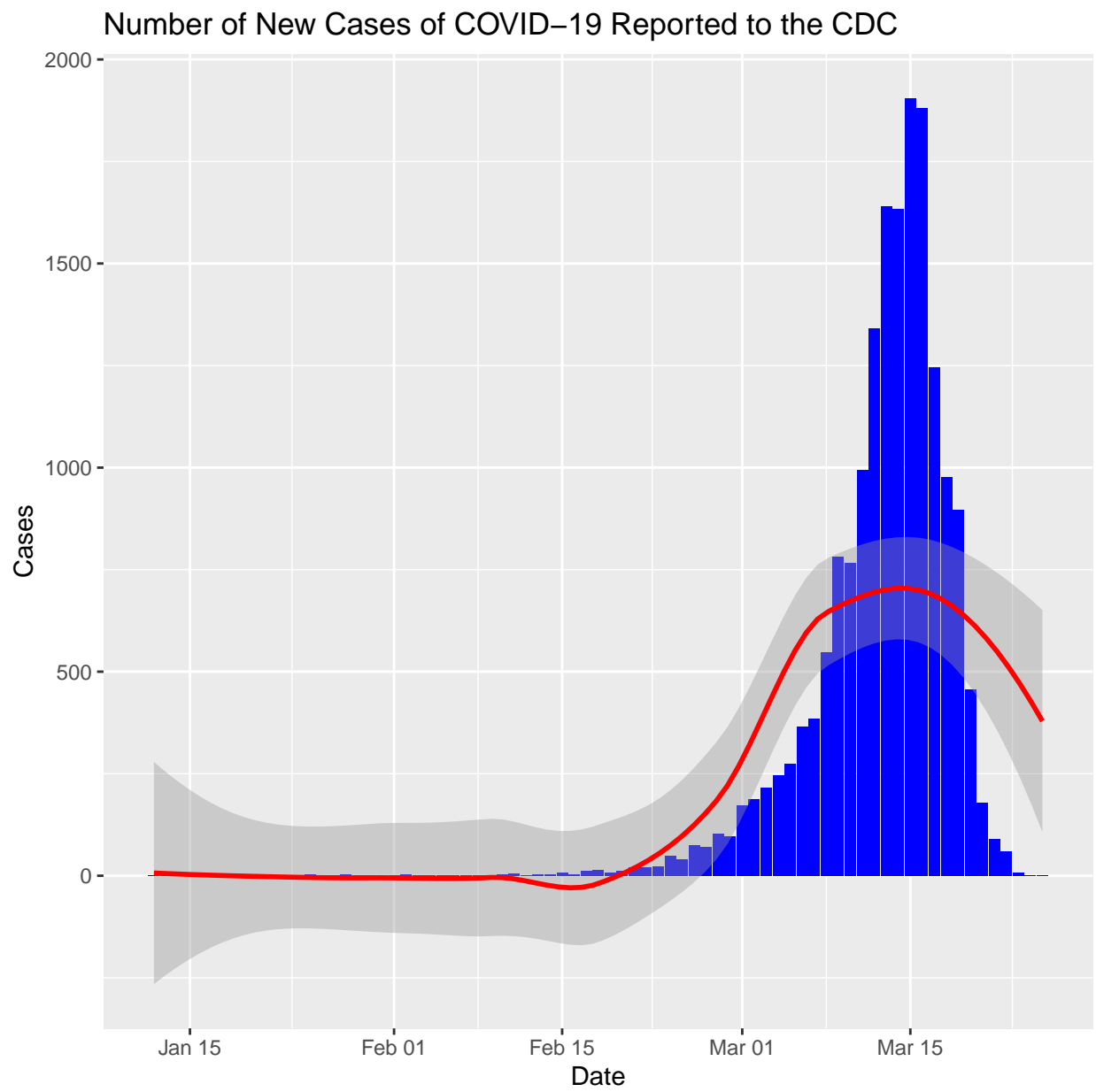


Figure 2: Epi curve 2, traditional

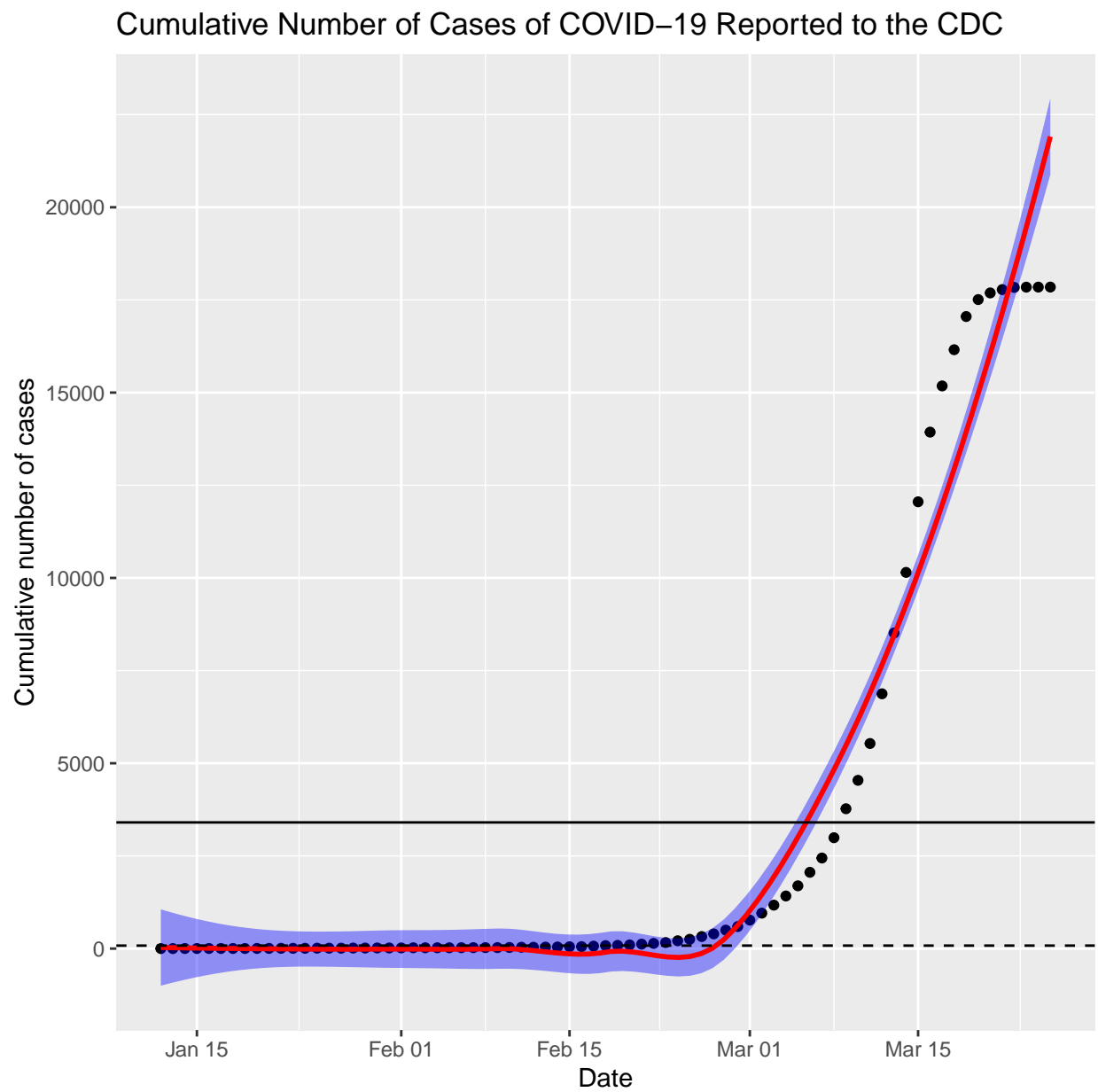


Figure 3: Cumulative cases

```

epi <- ggplot(data = cdc)

epi + geom_point(aes(x = Date,
                     y = `Number of new cases`))+
#   geom_line(aes(x = Date,
#                 y = `Number of new cases`),
#             linetype = 2) +
  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",
       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'

```

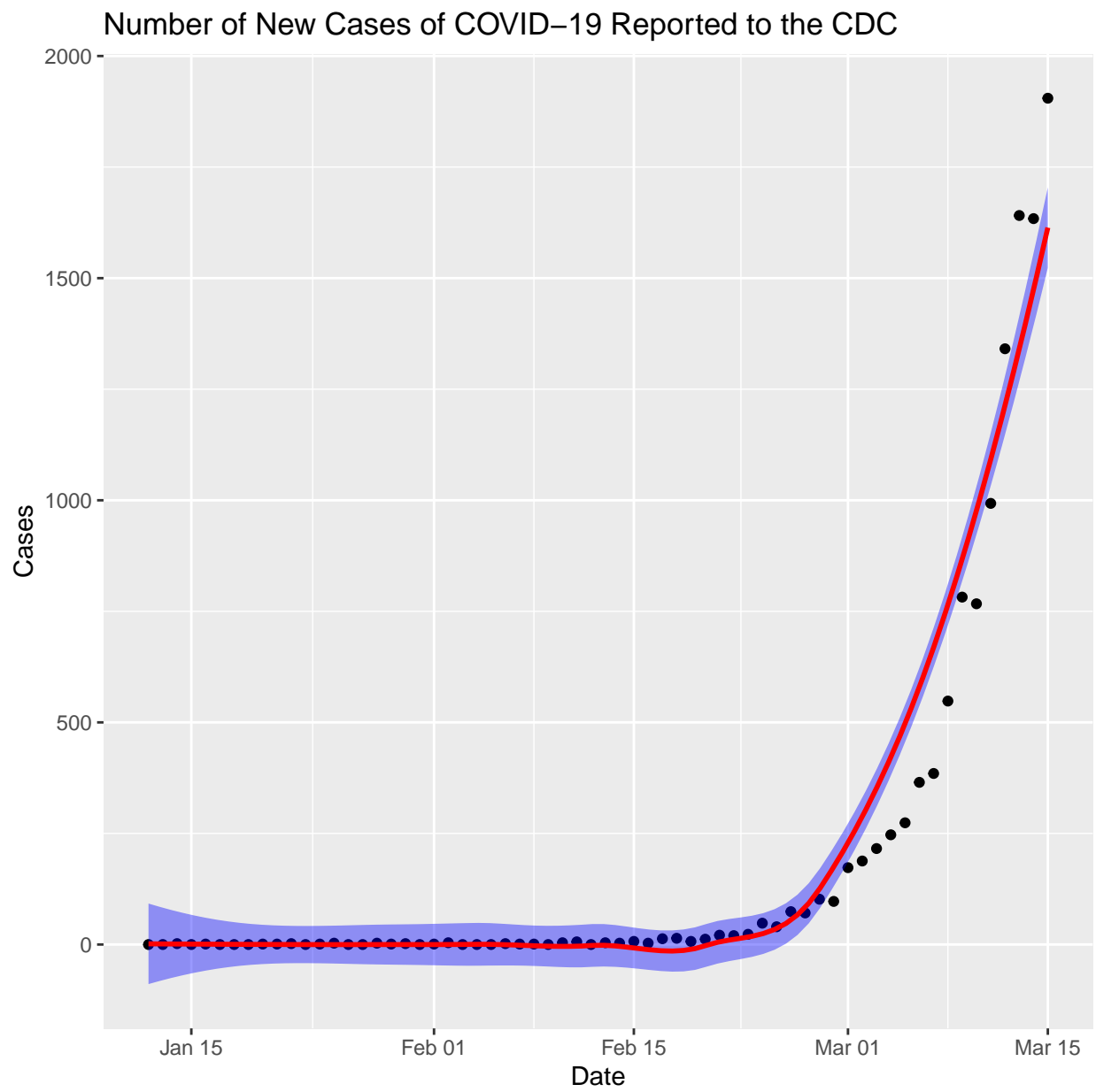


Figure 4: Epi curve 1

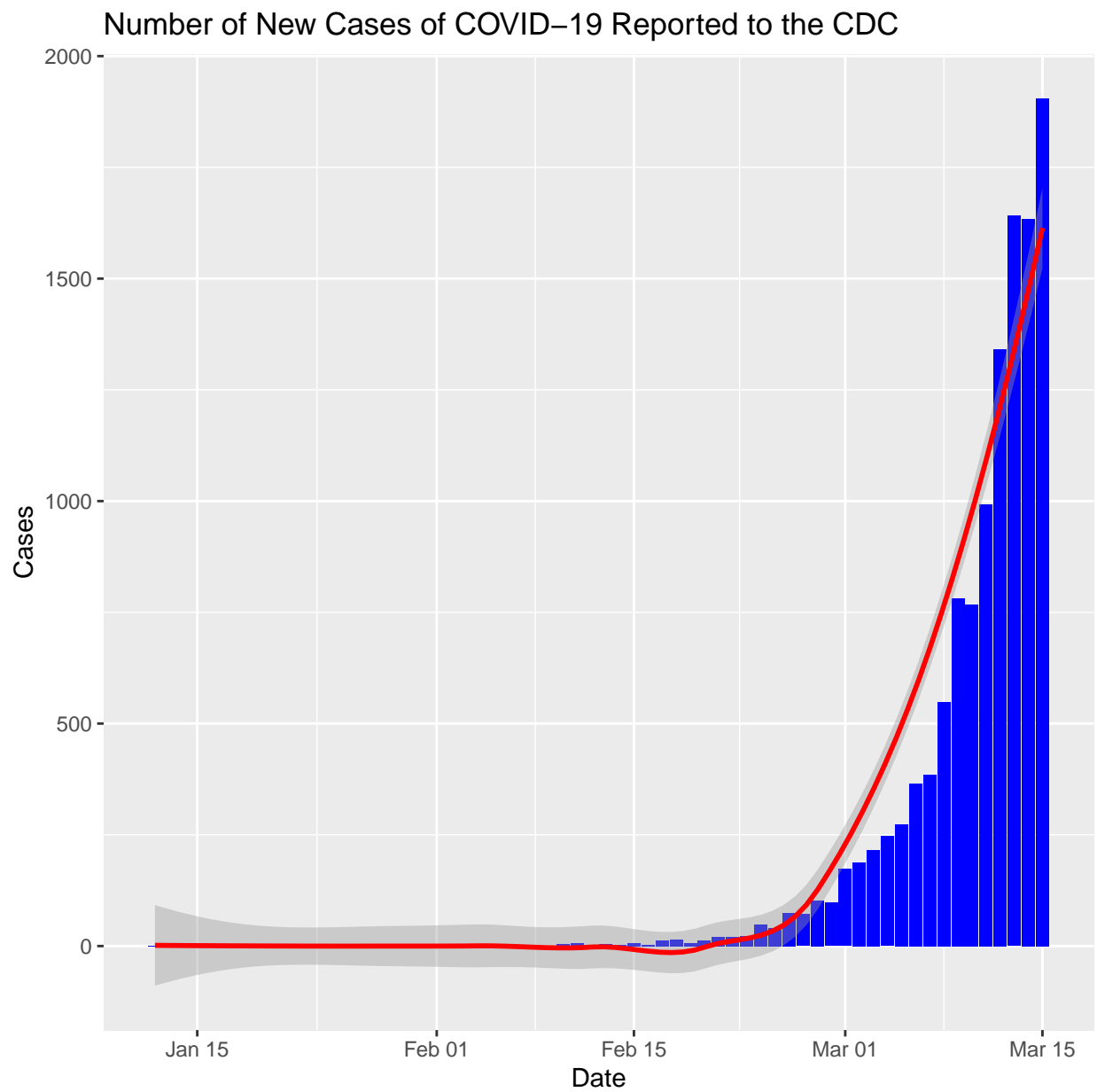


Figure 5: Epi curve 2, traditional

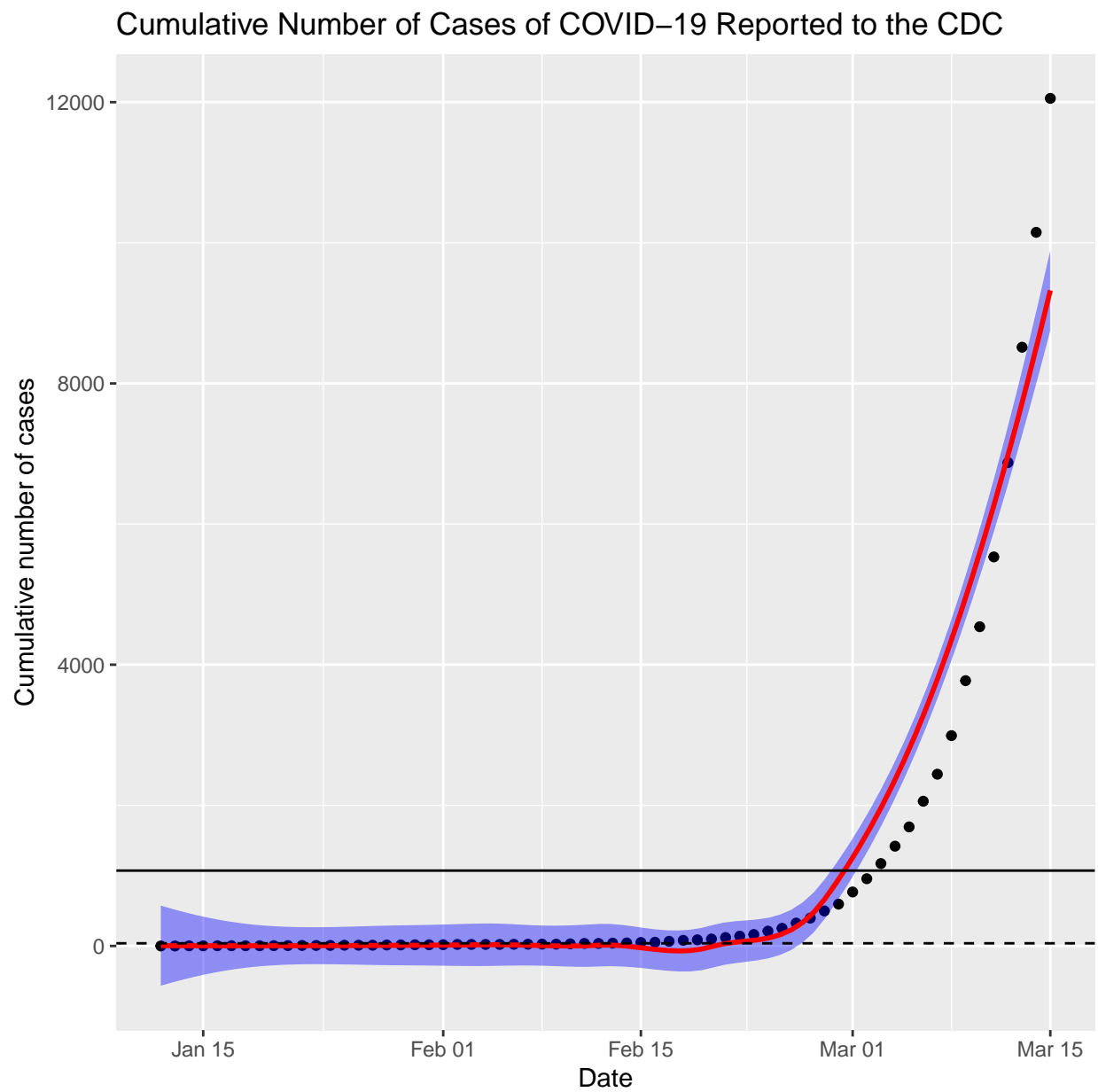


Figure 6: Cumulative cases