1.) Choose any of the 2 covid-19-us-counties datasets (2020 and 2021), perform the following:

```
library(readr)
library(ggplot2)
us_counties_2020 <- read_csv("~/Data_Analytics/Assignment03/covid-19-data-master/us-counties-2020.csv")
```

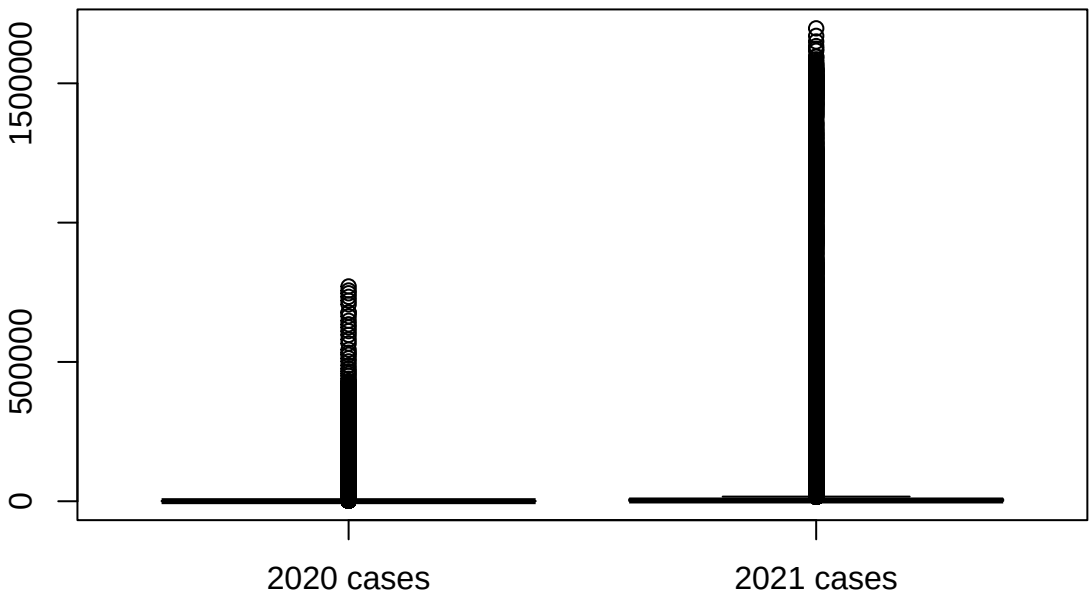
```
## Rows: 884737 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): county, state, fips
## dbl  (2): cases, deaths
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
us_counties_2021 <- read_csv("~/Data_Analytics/Assignment03/covid-19-data-master/us-counties-2021.csv")
```

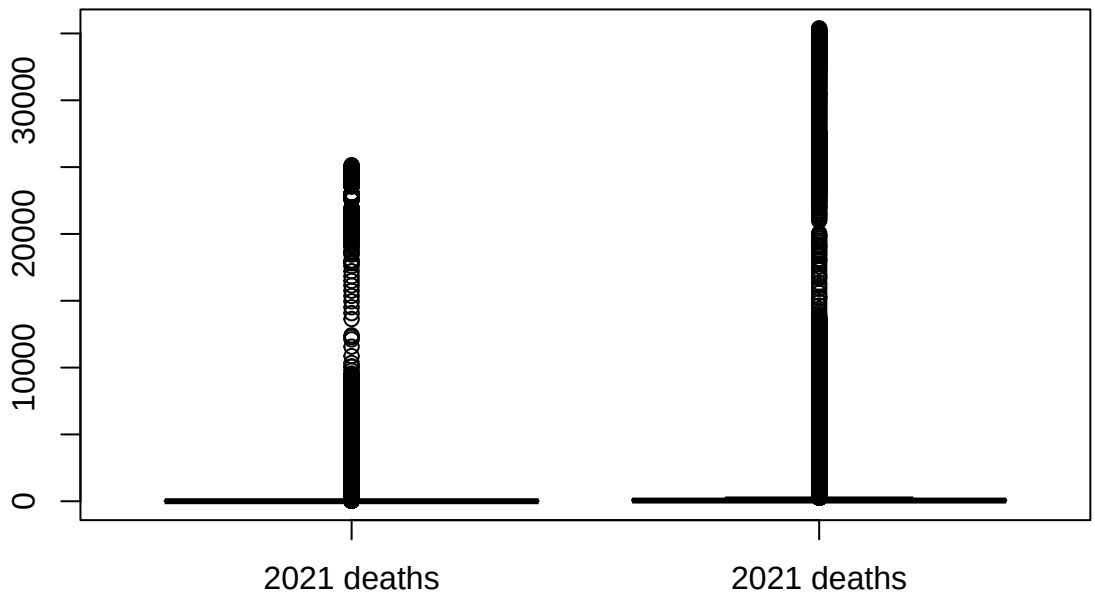
```
## Rows: 1185373 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): county, state, fips
## dbl  (2): cases, deaths
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

1a.) Create boxplots for the "Cases and"Deaths" variables comparing the varibales between the 2 datasets, i.e. two figures (one for each variable) with 2 boxplots (for the 2 different datasets).

```
boxplot(us_counties_2020$cases, us_counties_2021$cases, names=c("2020 cases", "2021 cases"))
```




```
boxplot(us_counties_2020$deaths, us_counties_2021$deaths, names=c("2021 deaths", "2021 deaths"))
```



1a.) Describe and run summary statistics on the two chosen variables and explain them in your own words. min. 2-3 sentences.

```
summary(us_counties_2020$cases)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0      36     228    1952     993  770915
```

```
summary(us_counties_2021$cases)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1136    2778   11160    7340 1697286
```

```
summary(us_counties_2020$deaths)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0     0.0     4.0    53.6    21.0 25144.0   18761
```

```
summary(us_counties_2021$deaths)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0    20.0    52.0   193.6   125.0 35382.0   28470
```

There is a large increase in the mean and median number of COVID cases and deaths between 2020 and 2021. Additionally the median deaths recorded in one day is 4.0 in 2020, which is likely because the outbreak in the US didn't start for several months into the year. Finally it is interesting to note that at least 1 county, even in 2021, didn't report any cases for a day, nor did they report lived.

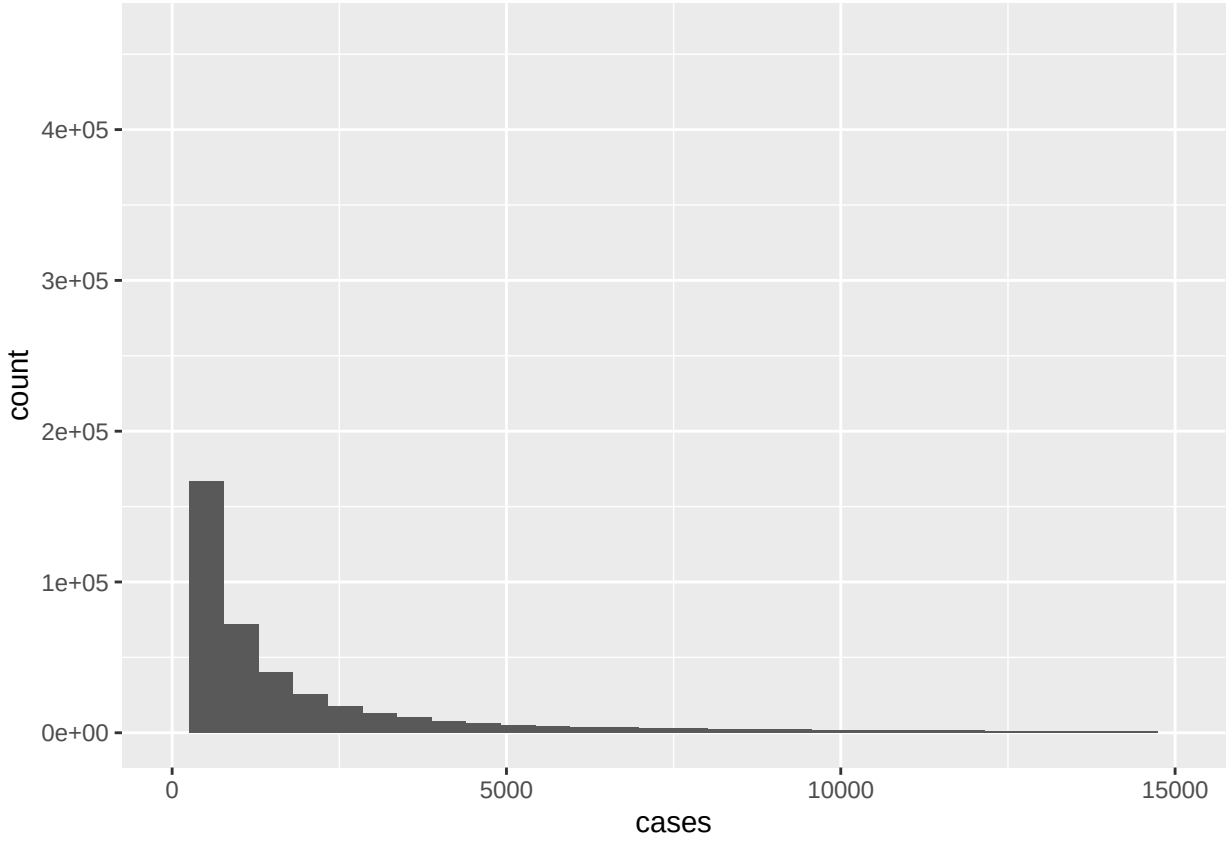
1b.) create histograms for those two varibales in the 2 datasets. Describe the distributions in terms of known parametric distributions and similarities/differences among them.

```
ggplot(us_counties_2020) +
  geom_histogram(aes(x=cases)) +
  xlim(0, 15002)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 21708 rows containing non-finite outside the scale range
## (`stat_bin()`).
```

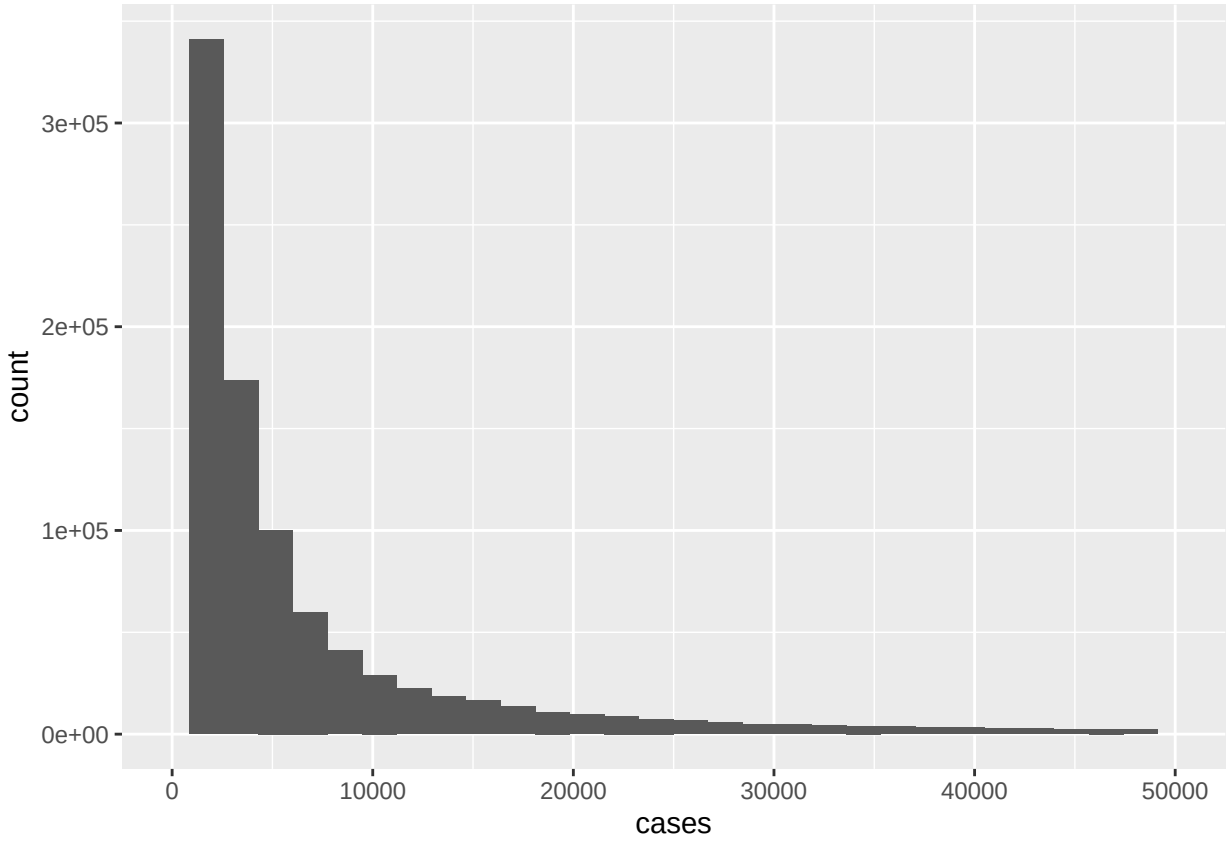
```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```




```
ggplot(us_counties_2021) +
  geom_histogram(aes(x=cases)) +
  xlim(0, 50002)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

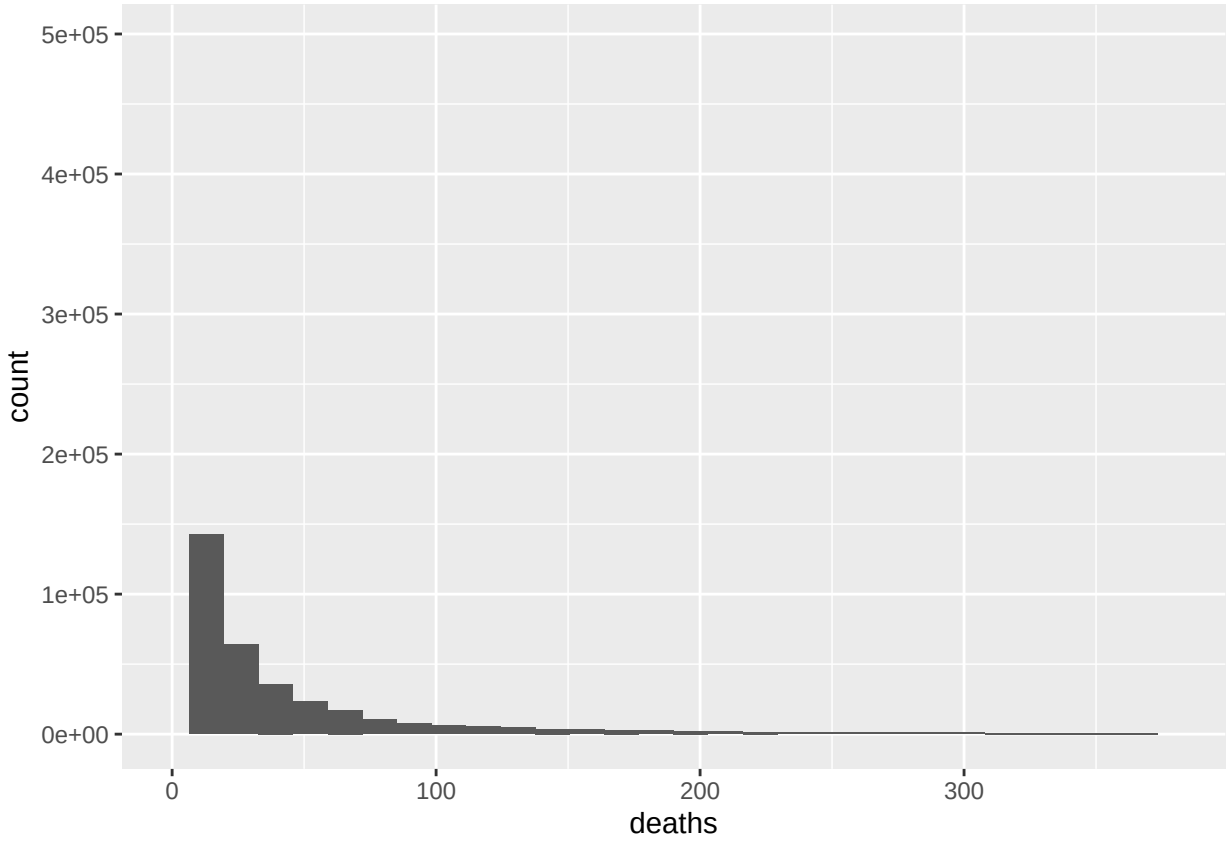
```
## Warning: Removed 51910 rows containing non-finite outside the scale range
## (`stat_bin()`).
## Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```




```
ggplot(us_counties_2020) +
  geom_histogram(aes(x=deaths)) +
  xlim(0, 380)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

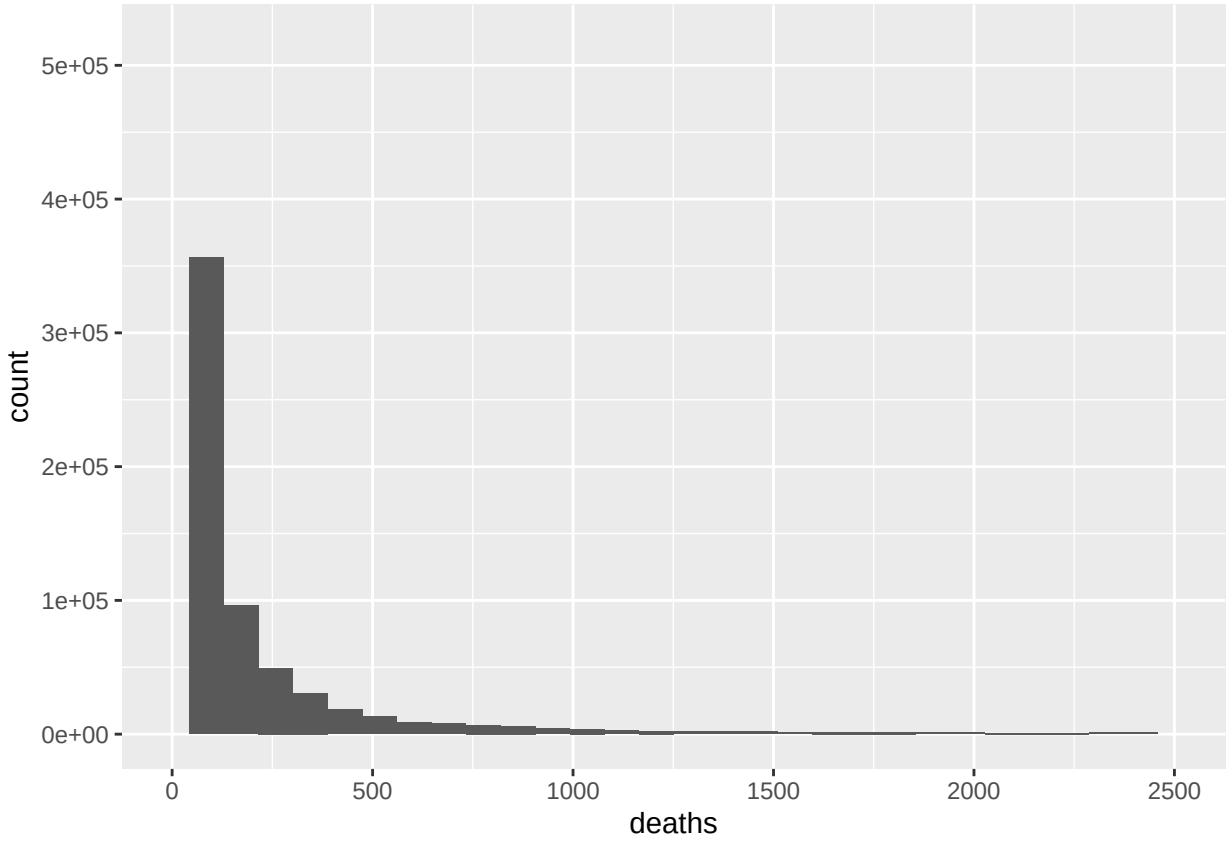
```
## Warning: Removed 40579 rows containing non-finite outside the scale range
## (`stat_bin()`).
## Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```




```
ggplot(us_counties_2021) +
  geom_histogram(aes(x=deaths)) +
  xlim(0, 2502)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

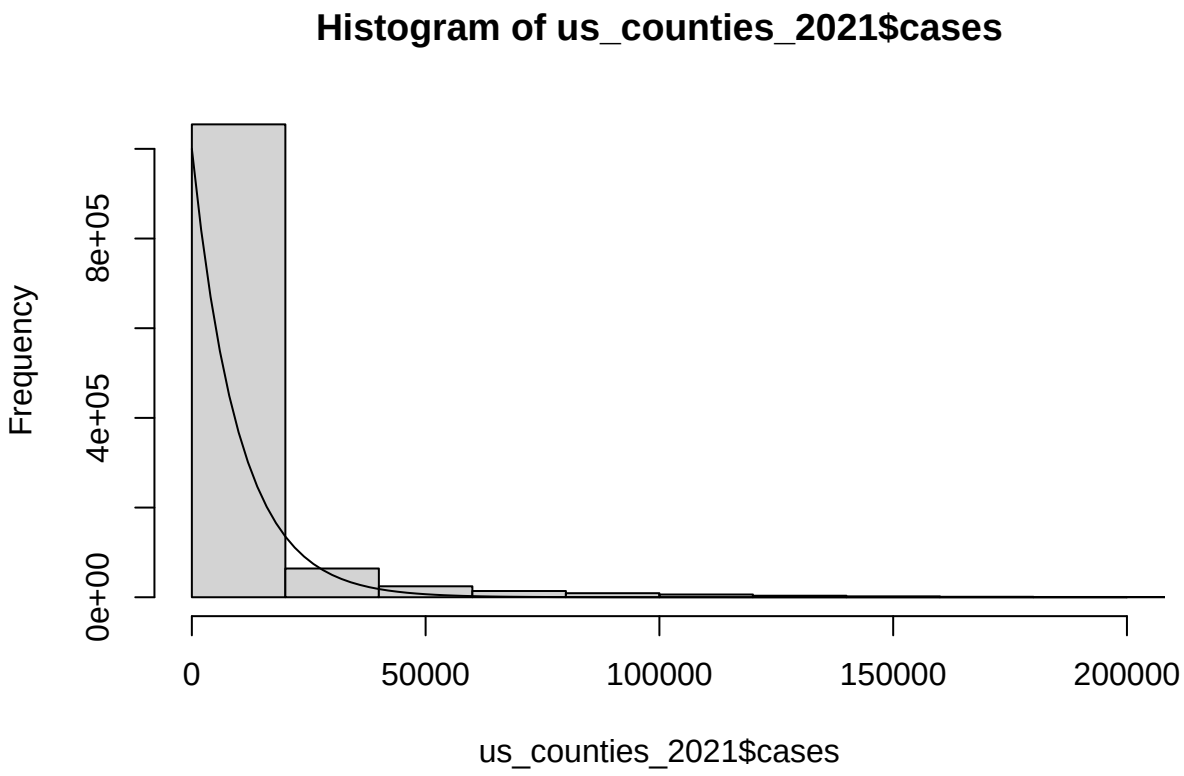
```
## Warning: Removed 38452 rows containing non-finite outside the scale range
## (`stat_bin()`).
## Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```



All of the distributions follow either a gamma or a chi-squared distribution with low degree of freedom. This is because of all of the distributions follow a similar trajectory as a 1/x curve, and of the various distributions, the gamma and chi-squared distributions are able to get to a 1/x curve. As these distributions do not

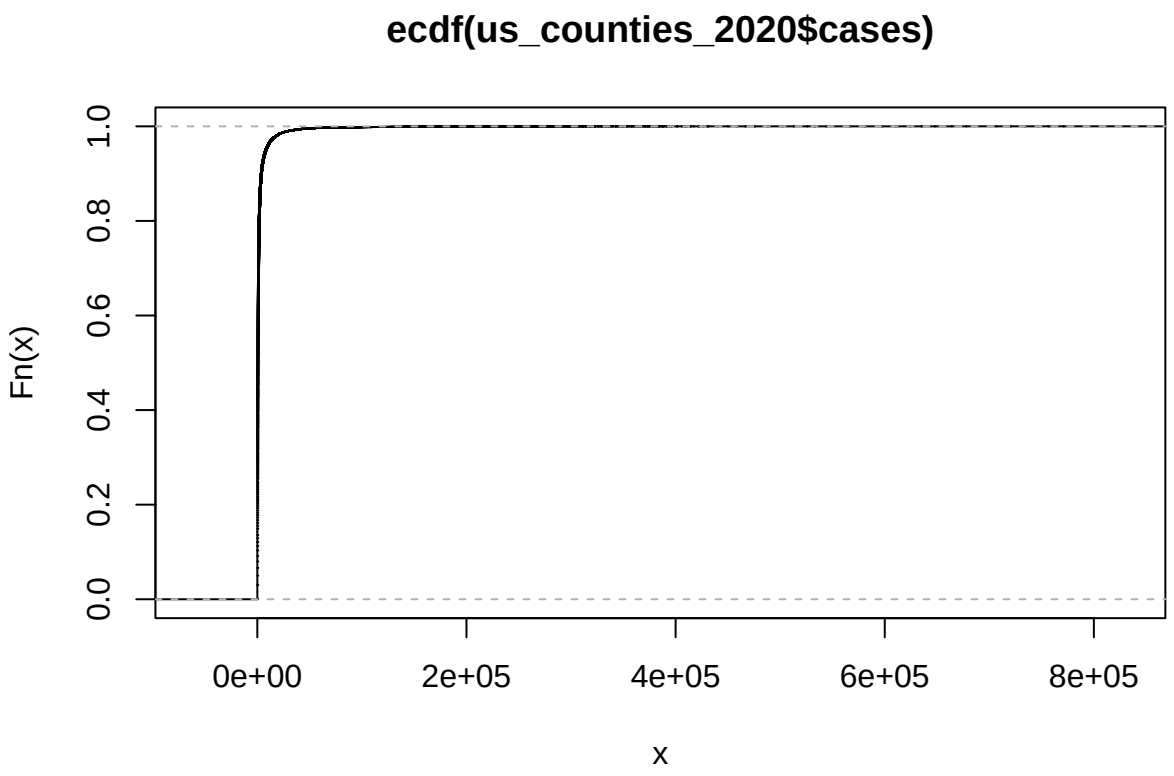
1b.) Plot the distribution you think matches the histogram (gamma) overlayed on the histogram

```
hist(us_counties_2021$cases, breaks=80, xlim=c(0,200000))
curve(100000 * dgamma(x/100000, shape = 1, rate = 10), add=TRUE)
```

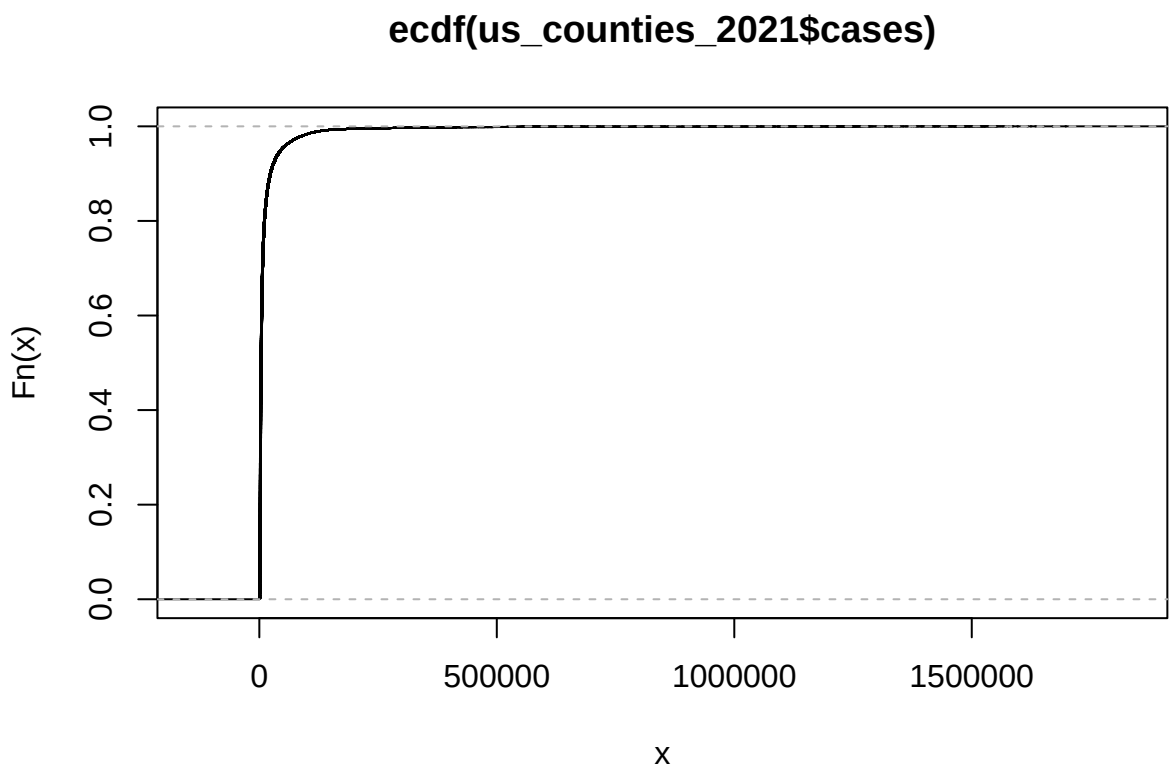


1c.) Plot the ECDFs (Empirical Cumulative Distribution Function for the two variables in both datasets

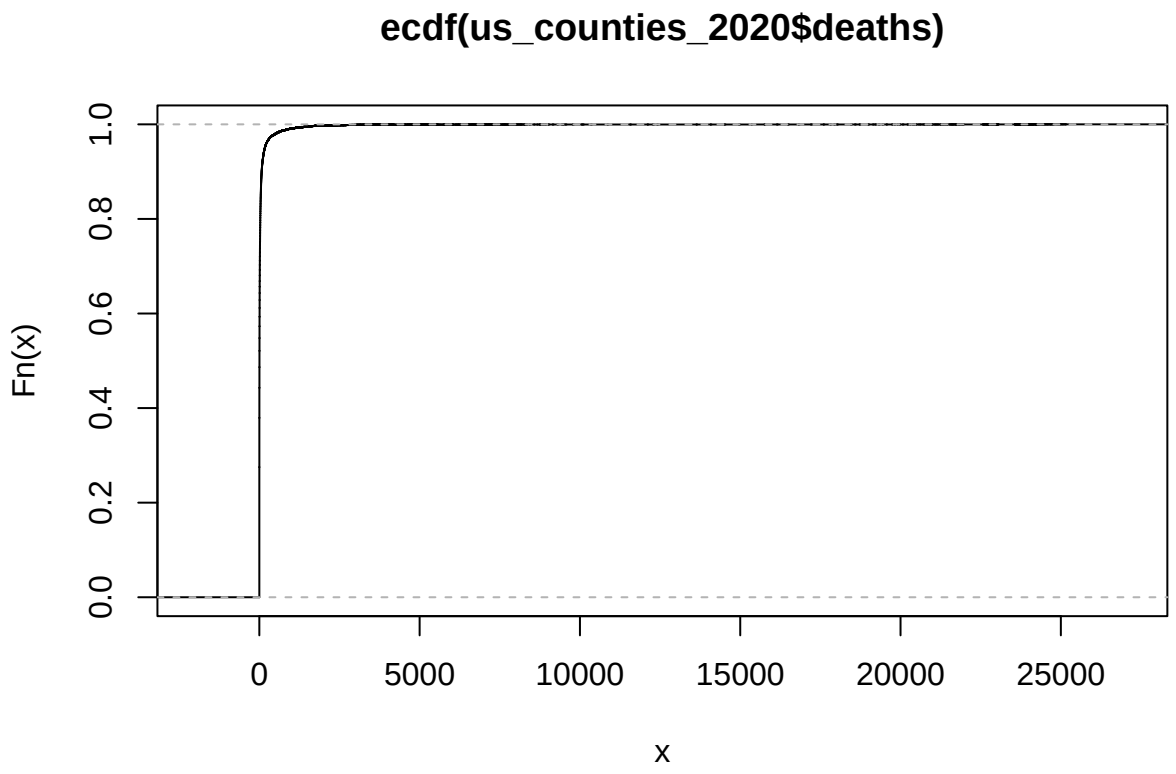
```
plot(ecdf(us_counties_2020$cases), do.points=FALSE, verticals=TRUE)
```



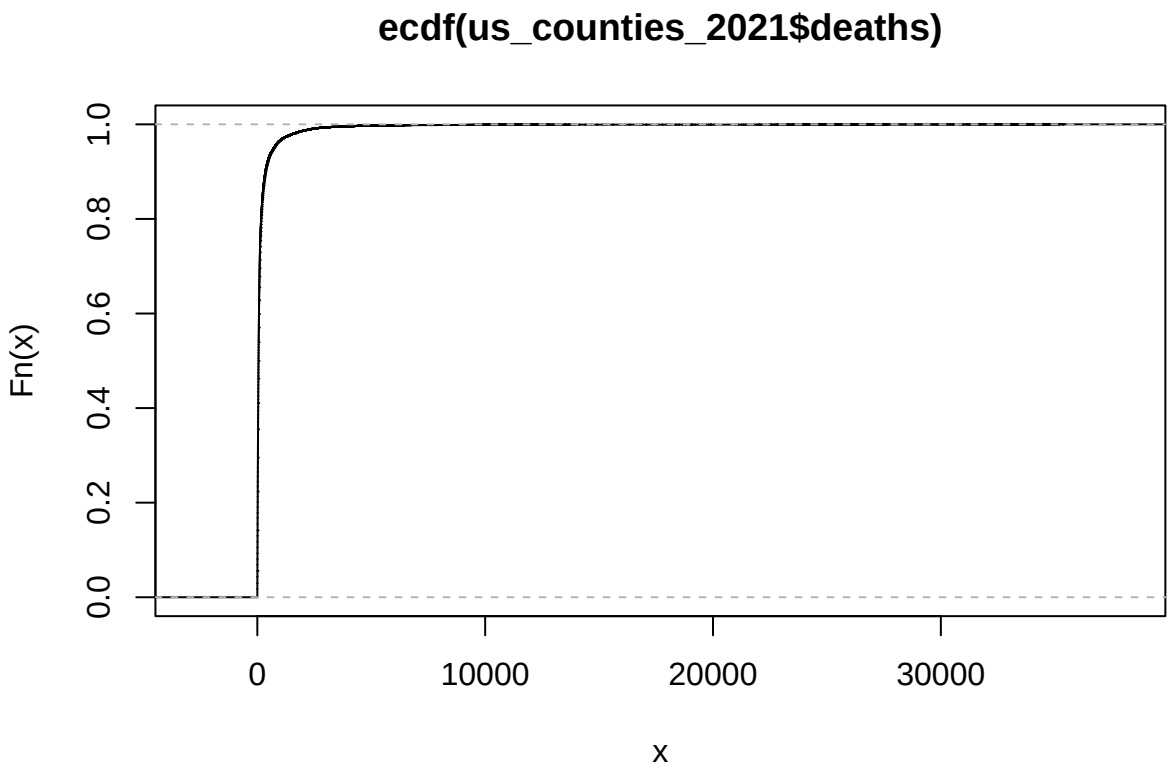

```
plot(ecdf(us_counties_2021$cases), do.points=FALSE, verticals=TRUE)
```




```
plot(ecdf(us_counties_2020$deaths), do.points=FALSE, verticals=TRUE)
```

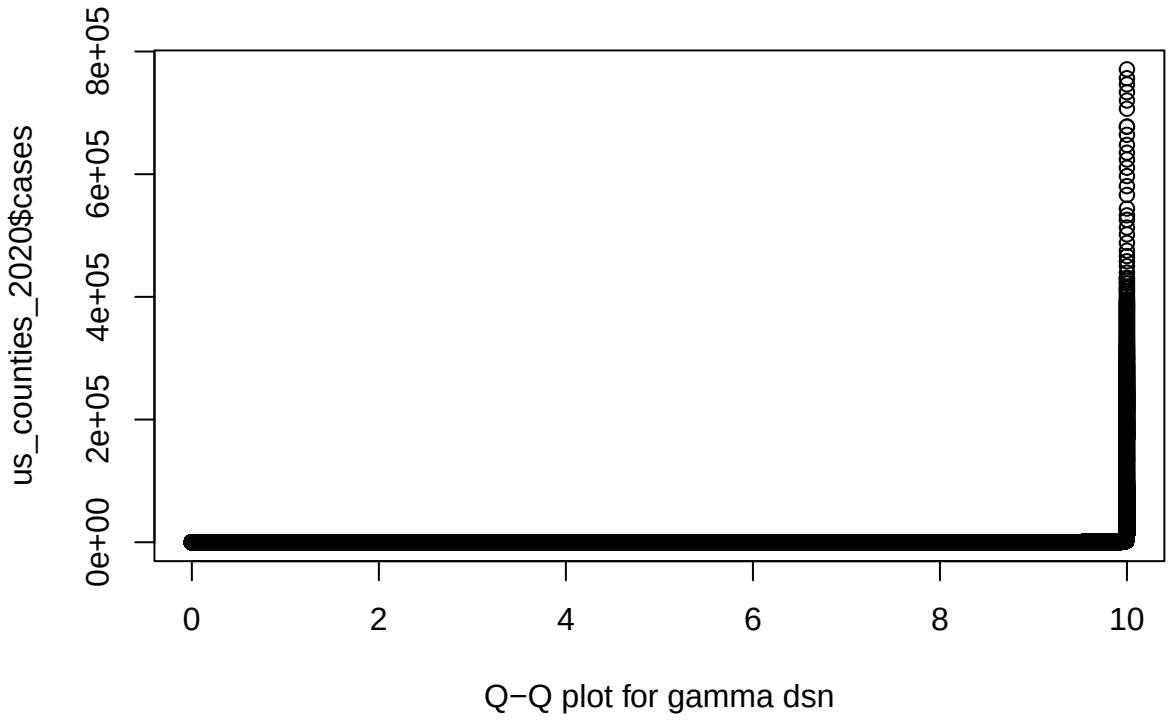



```
plot(ecdf(us_counties_2021$deaths), do.points=FALSE, verticals=TRUE)
```

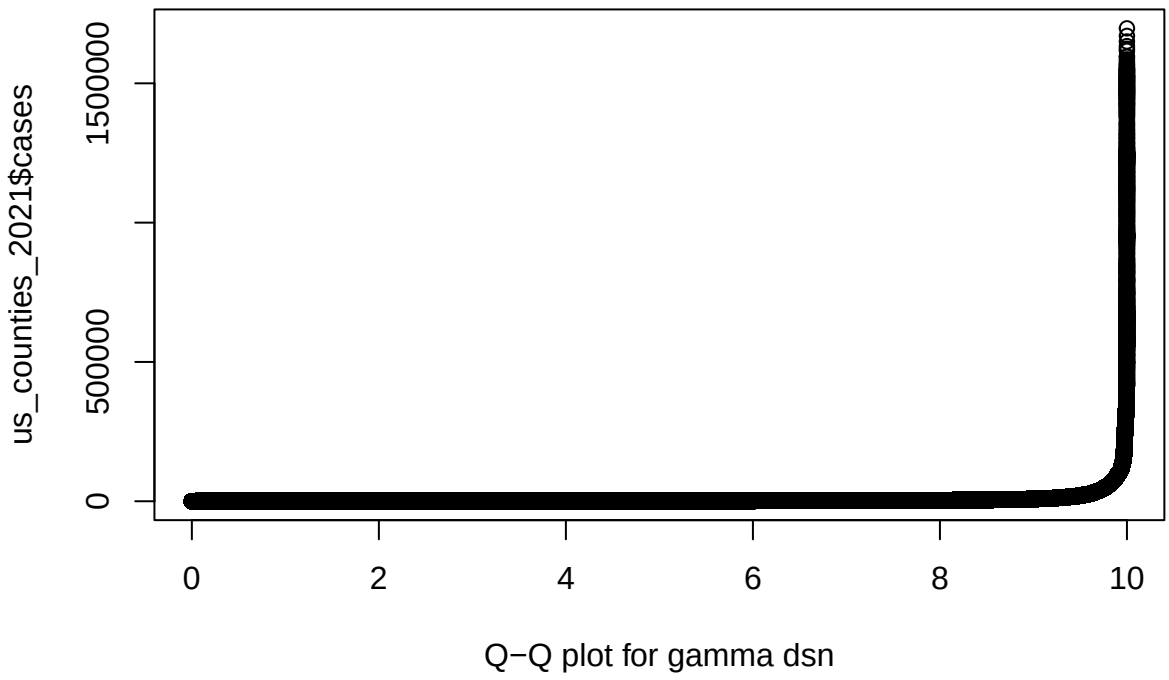


1c.) Plot the quantile-uqantile distribution using a suitable parametric distribution you chose in 1b (gamma)

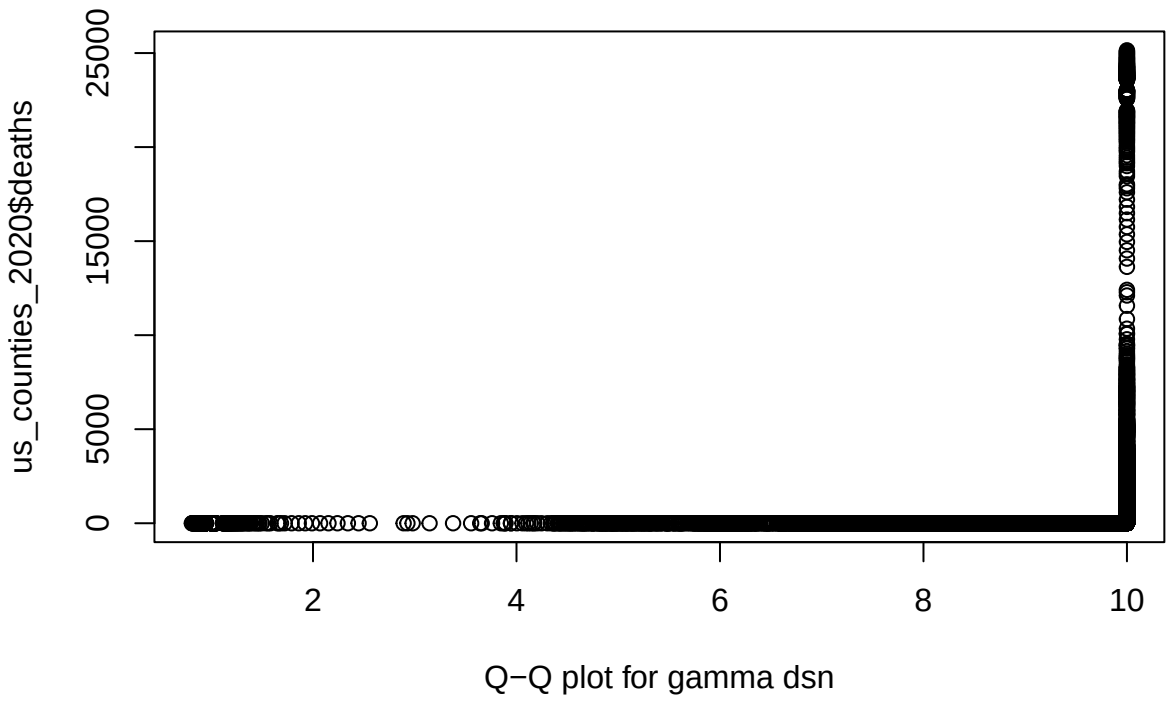
```
gamma_dist <- dgamma(us_counties_2020$cases/100000, shape = 1, rate = 10)
qqplot(x=gamma_dist, y=us_counties_2020$cases, xlab = "Q-Q plot for gamma dsn")
```



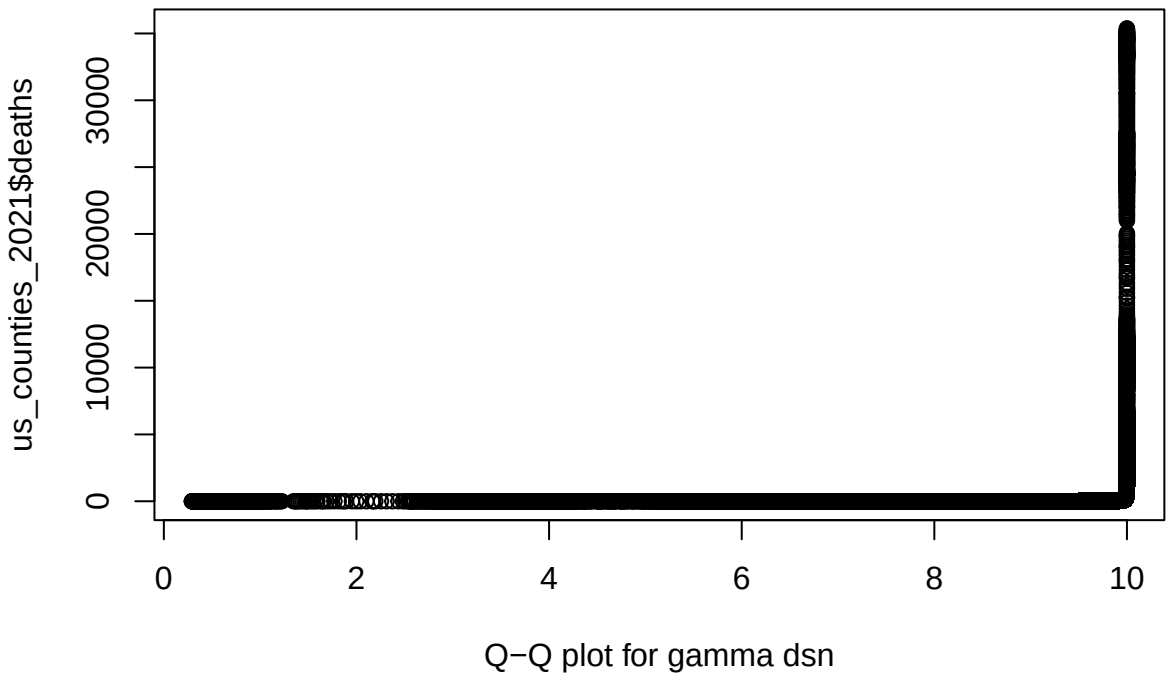

```
gamma_dist <- dgamma(us_counties_2021$cases/100000, shape = 1, rate = 10)
qqplot(x=gamma_dist, y=us_counties_2021$cases, xlab = "Q-Q plot for gamma dsn")
```




```
gamma_dist <- dgamma(us_counties_2020$deaths/100000, shape = 1, rate = 10)
qqplot(x=gamma_dist, y=us_counties_2020$deaths, xlab = "Q-Q plot for gamma dsn")
```




```
gamma_dist <- dgamma(us_counties_2021$deaths/100000, shape = 1, rate = 10)
qqplot(x=gamma_dist, y=us_counties_2021$deaths, xlab = "Q-Q plot for gamma dsn")
```



1c.) describe the features of these plots. min. 2-3 sentences.

All of the ECDF plots follow a logarithmic curves. When I paired the Q-Q plot with the gamma distribution it created an exponential curve. This means the distributions are very off. I'm not sure how to fix this, however it's possible that another distribution would be more accurate for this graph

3.) Using the NY house dataset

3a.) fit a linear model using the formula PRICE ~ BEDS + BATH + PROPERTYSQFT and identify the variable most significantly influencing house price.

```
ny_house <- read_csv("NY-House-Dataset.csv")
```

```
## Rows: 4801 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): BROKERTITLE, TYPE, ADDRESS, STATE, MAIN_ADDRESS, ADMINISTRATIVE_AR...
## dbl  (6): PRICE, BEDS, BATH, PROPERTYSQFT, LATITUDE, LONGITUDE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
lm.ny <- lm(PRICE~BEDS+BATH+PROPERTYSQFT, ny_house)
lm.ny
```

```
##
```

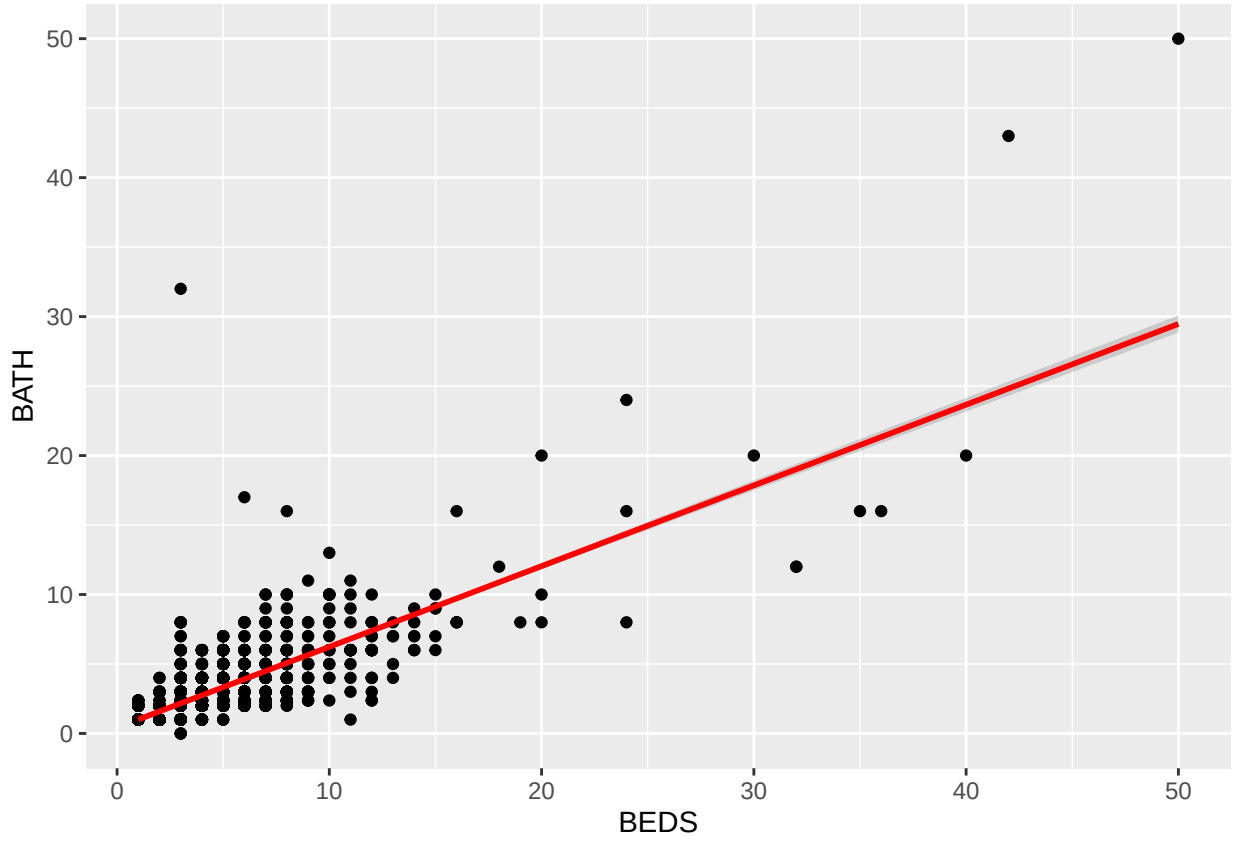
```
## Call:
## lm(formula = PRICE ~ BEDS + BATH + PROPERTYSQFT, data = ny_house)
##
## Coefficients:
##  (Intercept)          BEDS          BATH  PROPERTYSQFT
##    -1301012       -417608        958233          1275
```

The variable most significantly influencing the house price is the number of baths, as it has the highest absolute value coefficient. This means that this variable has the most influence. For example if it's coefficient was 0, that means no matter how many or how few the number of beds there would be, the estimated price would be not effected, all else equal.

3a.) Produce a scatterplot of that variable (BATH) with another and overlay the best fit line.

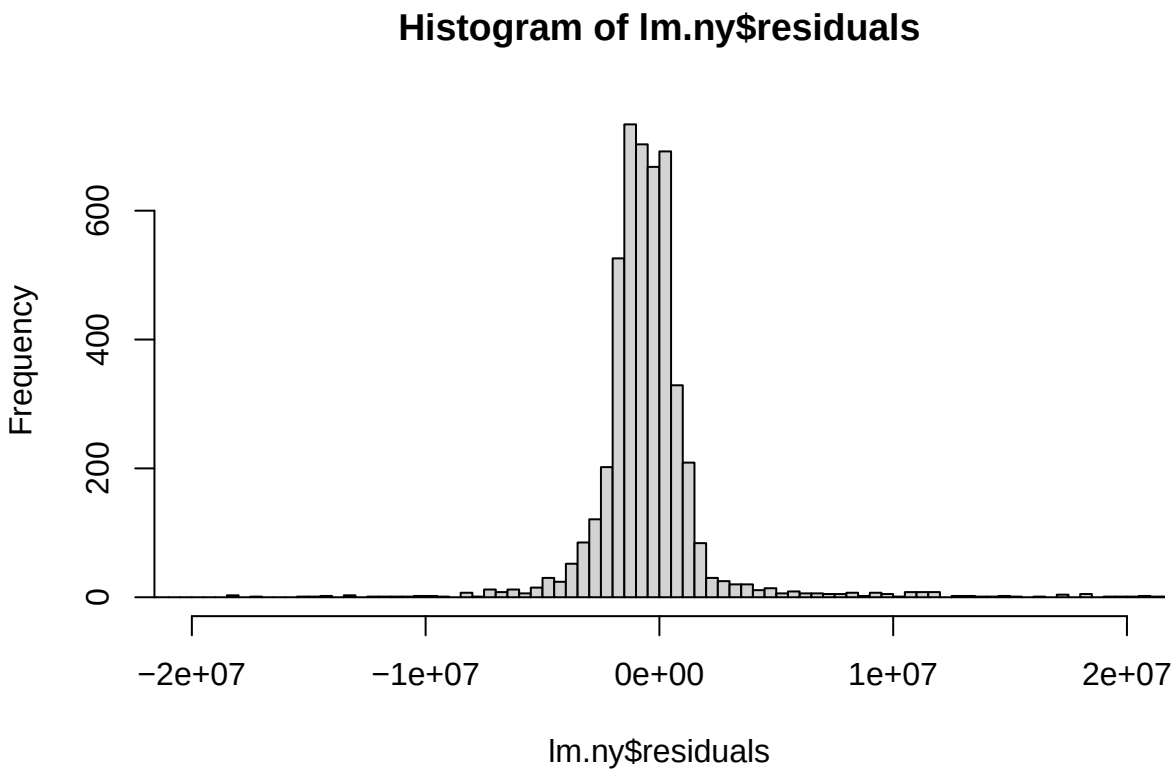
```
ggplot(ny_house, aes(x=BEDS, y=BATH)) +
  geom_point() +
  geom_smooth(method = "lm", color = "red")
```

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



3a.) Plot the residuals of the linear model.

```
hist(lm.ny$residuals, breaks=7500, xlim=c(-20000000,20000000))
```



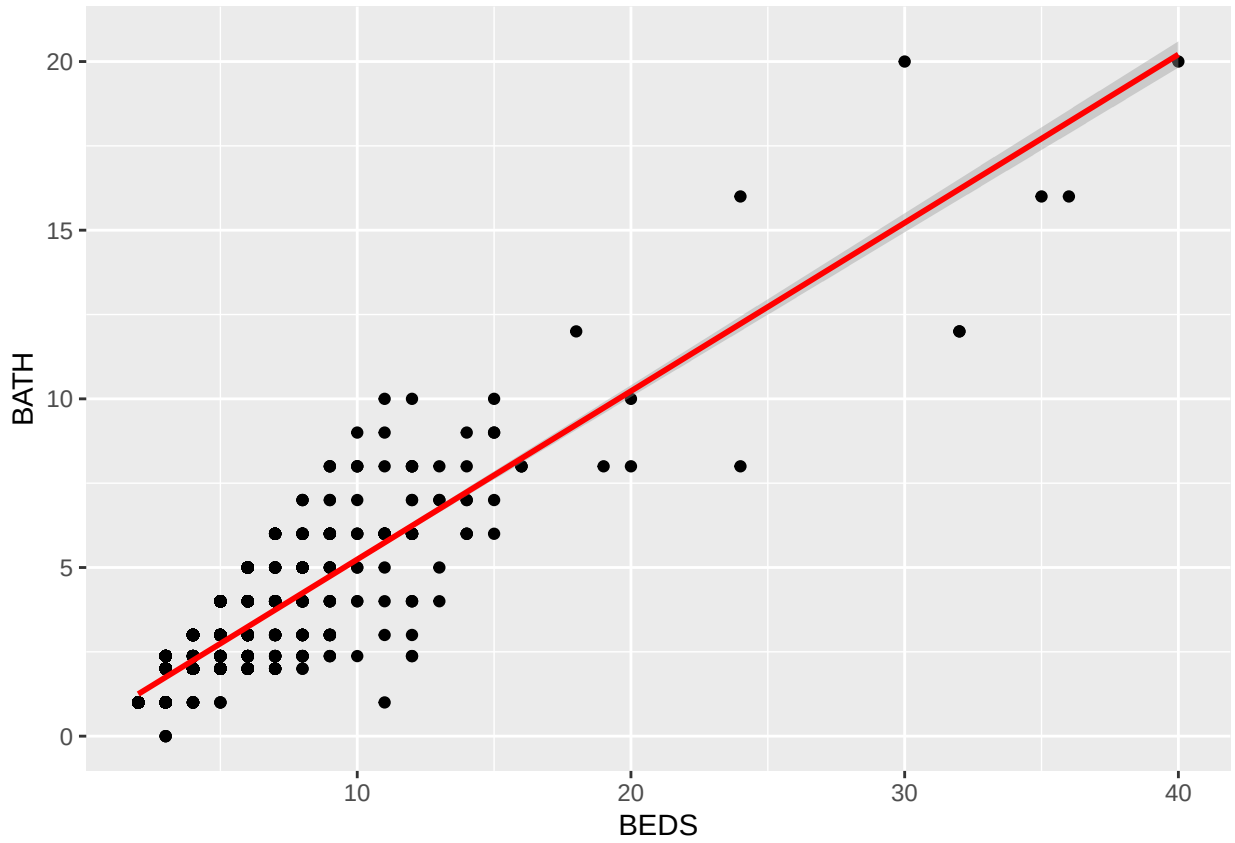
Derive a subset of the dataset according to any criteria (BED > BATH) and repeat the linear model with its plots. Explain how the significance of the input variables changes and your interpretation of the change.

```
ny_house.subset <- ny_house[ny_house$BEDS > ny_house$BATH, ]
lm.ny.s <- lm(PRICE~BEDS+BATH+PROPERTYSQFT, ny_house.subset)
lm.ny.s
```

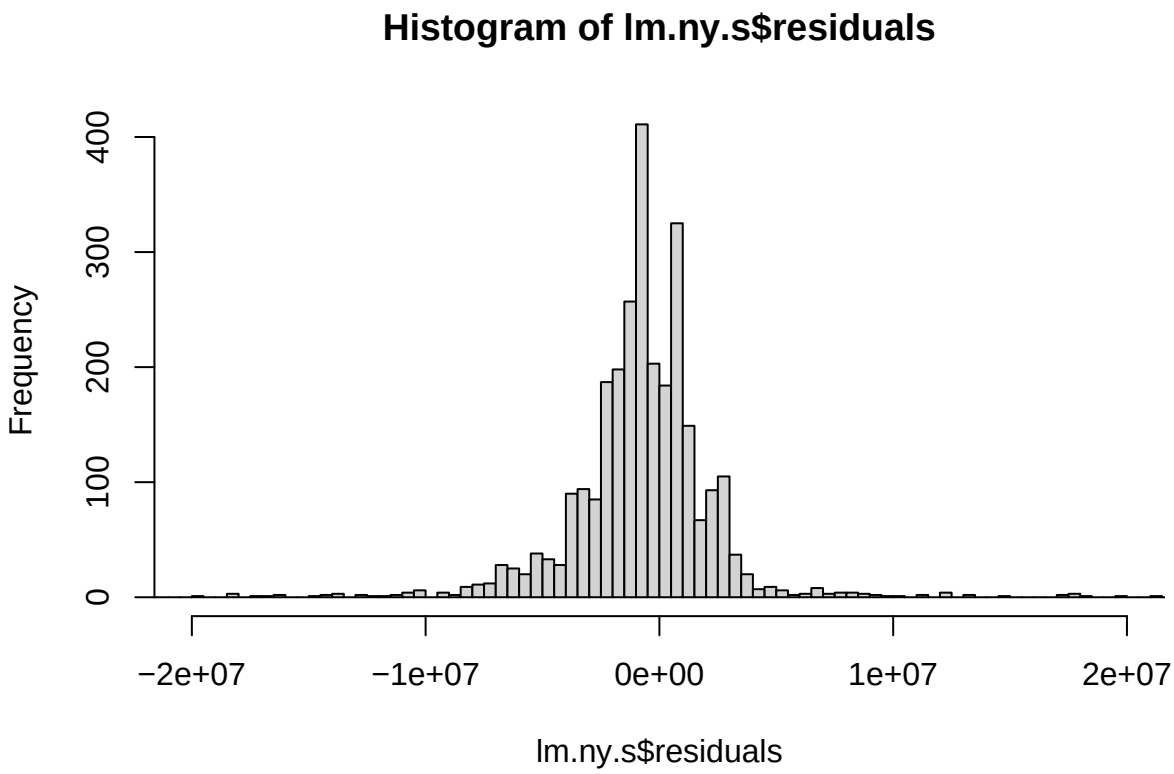
```
##
## Call:
## lm(formula = PRICE ~ BEDS + BATH + PROPERTYSQFT, data = ny_house.subset)
##
## Coefficients:
##  (Intercept)          BEDS          BATH  PROPERTYSQFT
##     -1948314      -1323686       2964277          1271
```

```
ggplot(ny_house.subset, aes(x=BEDS, y=BATH)) +
  geom_point() +
  geom_smooth(method = "lm", color = "red")
```

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




```
hist(lm.ny.s$residuals, breaks=7500, xlim=c(-20000000,20000000))
```



because oll of the methods use are results of manipulating an input, when that input changes so to do the results. This is why the linear model coefficients for BED and BATH changed so much. While this didn't change too much of the interpretation of the results, I'm sure if there were more fields it might have.