Covid 19 Analysis

Megan Arnold

2022-09-05

Load Libraries

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.1
## v readr   2.1.2     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(lubridate)
```

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

```
knitr::opts_chunk$set(echo = TRUE)
```

Import Data

The data that is going to be used is the Johns Hopkins' time series data. This data source has references and links to the origins of the data and it has the critical information required for this analysis. I will import the data directly from the raw data source on GitHub. The data includes both global data and US data for cumulative cases and deaths.

Question of Interest

I'm going to use this data to get a better understanding of the cases around my location in North Carolina, United States. I will compare cumulative deaths to the cases and build a model attempting to predict the

information. Within the analysis, I will also attempt to identify any potential causes for what we are seeing in the data and make recommendations for future research.

```
rootUrl = "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_cov

fileNames = c("time_series_covid19_confirmed_global.csv","time_series_covid19_deaths_global.csv","time_

urls = str_c(rootUrl,fileNames)

globalCases = read_csv(urls[1], show_col_types = FALSE)
globalDeaths = read_csv(urls[2], show_col_types = FALSE)
usCases = read_csv(urls[3], show_col_types = FALSE)
usDeaths = read_csv(urls[4], show_col_types = FALSE)
problems()
```

Clean Data

Data description

The data source provides information for the cumulative number of cases and deaths for countries as a whole, sometimes broken down by providence/state, in the Global Data. Then there is another subset of data that specifically is for the United States, which further breaks it down into counties.

After importing the data, I need to convert the 4 DataFrame to long format for easier analysis. This will create a single column for the cases and/or deaths, with data being a separate column.

Global Data Cleaning

I will put the data in a long format, this is the ideal format for many R packages.

```
globalCasesLong = globalCases %>%
  pivot_longer(cols = -c('Province/State','Country/Region',Lat,Long),
               names_to = 'date',
               values_to = 'cases') %>%
  select(-c(Lat,Long))
```

```
globalDeathsLong = globalDeaths %>%
  pivot_longer(cols = -c('Province/State','Country/Region',Lat,Long),
               names_to = 'date',
               values_to = 'deaths') %>%
  select(-c(Lat,Long))
```

US Data Cleaning

```
usCasesLong = usCases %>%
  pivot_longer(cols = -(UID:Combined_Key),
               names_to = 'date',
               values_to = 'cases') %>%
  select(Admin2:cases) %>%
```

```
  mutate(date = mdy(date)) %>%
  select(-c(Lat,Long_))
```

```
usDeathsLong = usDeaths %>%
  pivot_longer(cols = -(UID:Population),
               names_to = 'date',
               values_to = 'deaths') %>%
  select(Admin2:deaths) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,Long_))
```

Join Data

For the global data, I will join the DataFrames based on Province/State, Country/Region, and Date.

For the US data, I will join the DataFrames based on Admin2 (county), Province/State, Country/Regoin, Combined_Key, and Date.

```
usData = usCasesLong %>%
  full_join(usDeathsLong) %>%
  filter(cases > 0)
```

```
## Joining, by = c("Admin2", "Province_State", "Country_Region", "Combined_Key",
## "date")
```

Global Data Combined

```
head(globalData)
```

```
## # A tibble: 6 x 5
##   Province_State Country_Region date       cases deaths
##   <chr>          <chr>          <date>     <dbl>  <dbl>
## 1 <NA>           Afghanistan    2020-02-24     5      0
## 2 <NA>           Afghanistan    2020-02-25     5      0
## 3 <NA>           Afghanistan    2020-02-26     5      0
## 4 <NA>           Afghanistan    2020-02-27     5      0
## 5 <NA>           Afghanistan    2020-02-28     5      0
## 6 <NA>           Afghanistan    2020-02-29     5      0
```

As we can see in the above table, We have the Province_State and Country_Region as identifiers for the number of cases and deaths on a specific day. This will allow us to group based on the country and run a time-series analysis if desired

US Data Combined

```
head(usData)
```

```
## # A tibble: 6 x 8
##   Admin2  Province_State Country_Region Combin~1 date       cases Popul~2 deaths
##   <chr>   <chr>          <chr>          <chr>    <date>     <dbl>   <dbl>  <dbl>
## 1 Autauga Alabama        US             Autauga~ 2020-03-24     1   55869      0
## 2 Autauga Alabama        US             Autauga~ 2020-03-25     5   55869      0
## 3 Autauga Alabama        US             Autauga~ 2020-03-26     6   55869      0
## 4 Autauga Alabama        US             Autauga~ 2020-03-27     6   55869      0
## 5 Autauga Alabama        US             Autauga~ 2020-03-28     6   55869      0
## 6 Autauga Alabama        US             Autauga~ 2020-03-29     6   55869      0
## # ... with abbreviated variable names 1: Combined_Key, 2: Population
```

In the above Table, we can see that we have the County (Admin2), Province_State, Country_Region, and a Combined_Key. This can be grouped upon to do a time-series analysis on the cases, and deaths. This table also includes the population information for the county (verified through an independent resource).

Finding Population Data

The US DataFrame has a column for the population data; however, the global data does not. Fortunately, the original data source has a lookup table for that information. I will download it and join it with the global data.

```
uidURL = 'https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/UID_ISO_F

uidData = read_csv(uidURL, show_col_types = FALSE) %>%
  select(-c(Lat,Long_,Combined_Key,code3,iso2,iso3,Admin2))
uidData
```

```
## # A tibble: 4,317 x 5
##       UID FIPS  Province_State Country_Region      Population
##     <dbl> <chr> <chr>          <chr>                   <dbl>
##  1     4 <NA>  <NA>           Afghanistan           38928341
##  2     8 <NA>  <NA>           Albania                2877800
##  3    10 <NA>  <NA>           Antarctica                  NA
##  4    12 <NA>  <NA>           Algeria                43851043
##  5    20 <NA>  <NA>           Andorra                   77265
##  6    24 <NA>  <NA>           Angola                 32866268
##  7    28 <NA>  <NA>           Antigua and Barbuda       97928
##  8    32 <NA>  <NA>           Argentina              45195777
##  9    51 <NA>  <NA>           Armenia                 2963234
## 10    40 <NA>  <NA>           Austria                 9006400
## # ... with 4,307 more rows
```

Visualization of Historical Cases and Deaths in several counties in North Carolina

I currently live in North Carolina, in the United States. I decided to look at the major counties in the state and the counties close to my location. The two main cities in North Carolina, Charlotte and Raleigh, are in Mecklenburg and Wake county.

```
coeff = 0.01 #Used for the secondary y axis
usData %>% filter(Province_State == "North Carolina") %>%
  filter(Admin2 == "Forsyth" | Admin2 == "Surry" | Admin2 == "Mecklenburg" | Admin2 =="Wake") %>%
  ggplot(aes(x = date,y1 = cases, y2 = deaths)) +
  geom_line(aes(date,cases, color = Admin2)) +
  geom_line(aes(date,deaths/coeff)) +
  scale_y_continuous(name = "Number of Cases",
    sec.axis = sec_axis(~.*coeff, name = "Number of Deaths")
  ) +
  facet_wrap( ~Admin2) +
  #geom_point(aes(color = Admin2),) + #show.legend = FALSE) +
  theme_minimal() +
  scale_color_brewer(palette = "Dark2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(x = "Date", title = "Cumulative Cases and Scaled Deaths(100x) by Date", color = "County")
```

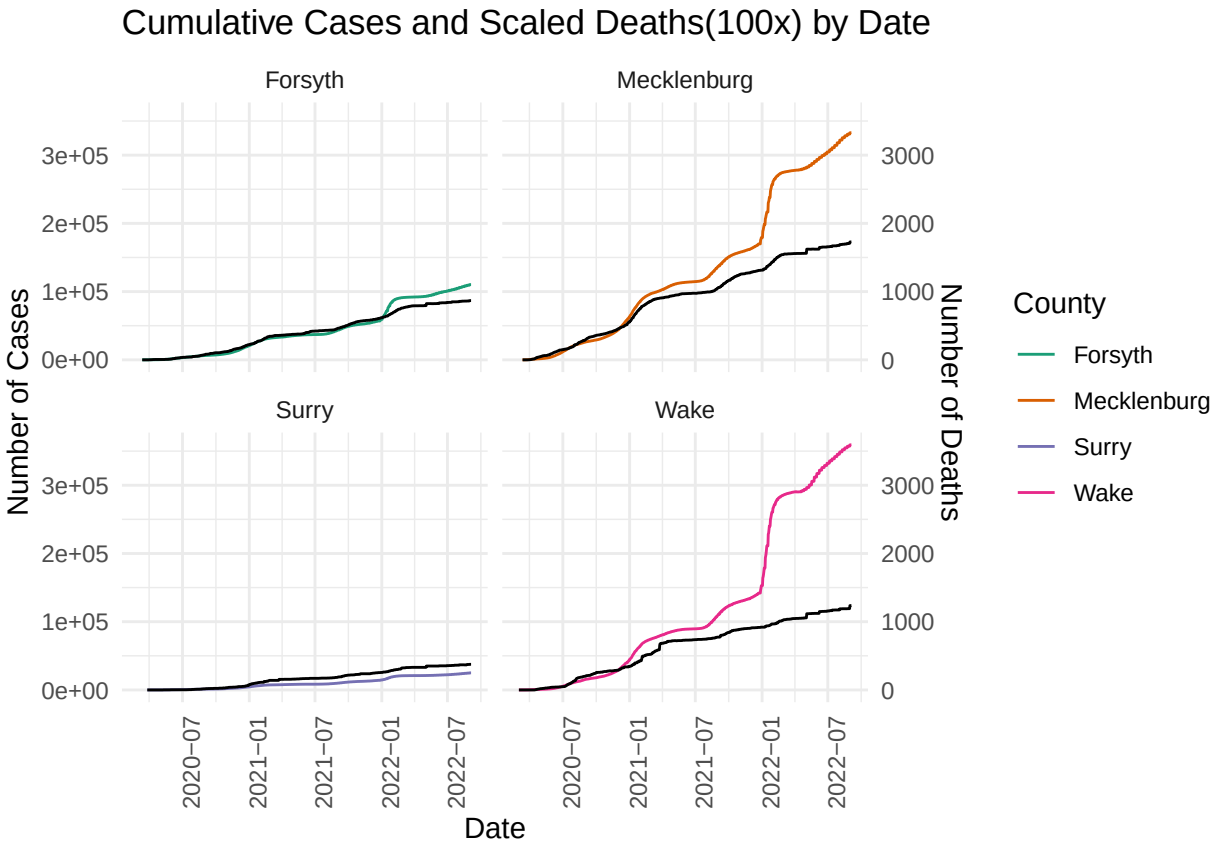

Figure 1: Cases and Deaths for select counties

As we can see in the above Figure "Cases and Deaths for select counties", I created a faceted grid with 4 plots, cases (use the legend for color reference) use the left y-axis, and deaths (black line) use the right y-axis. Through an iterative modification of the scaled coefficient, I scaled the deaths data by 100 times to get the data to align with the original cases information. It appears that initially, deaths were reliably about 1% of the total cases; however, it appears that the death rate decreased as the pandemic went on.

This could be for several reasons:

1. This could be effected by the testing being more widespread and complete. This would increases the total number of cases without having a significant impact on the number of deaths
    i) Positivity rate of the tests data could be superimposed on the above graphic to determine if this is a potential cause for the difference.
2. If our protocols for treatment improved throughout the pandemic, then the rate of death could have decreased towards the end of the time frame.
3. There were many variants of the virus, with their own unique mutations. The dominate variants could have been less fatal.
    i) Having a dataset where it had what variant the person tested positive for would be interesting because we could group that data to see if the fatality rate is different as the pandemic progressess (even taking the vaccine rate into consideration)
4. As the vaccine rolled out in early 2021, and became more wide spread towards the end of the year, could have had a lower probability of dying.
    i) It would be interesting to see vaccine rates superimposed on the above graphic. This could help draw a connection if one exists

From this data, I would like to see if I can take Forsyth county data and create a model between the number of cases and the expected deaths. I spend most of my time in this county and that information would be interesting to know what the relationship is.

Visualization of Cases vs Deaths in Forsyth County, North Carolina, United States

I would like to get a better understanding of the correlation between cases and the predicted deaths.

```
degrees = c(1,2)
color = c("red","green")

usData %>% filter(Province_State == "North Carolina") %>%
  filter(Admin2 == "Forsyth") %>%
  ggplot(aes(x = cases, y = deaths)) +
  geom_point() +
  map(1:length(degrees),~geom_smooth(method="lm",formula=y~poly(x,degrees[.x]),se=F,aes(color=factor(deg
  scale_color_manual(name="Degrees of Curve", breaks=degrees,values=set_names(color,degrees))+
  labs(title = "Deaths per Cases in Forsyth County",x="Cases",y="Deaths")+
  theme_bw()
```

As we can see from the above figure “Deaths vs Cases in Forsyth, North Carolina”, I plotted two fit curves against the data. The red fit curve is a first degree polynomial and the green fit curve is a second degree polynomial. It looks like a second degree curve fits the data a little better than a first degree curve. This could be due to the fact that the death rate appears to decrease as the pandemic continues (This apparent decrease is discussed in the previous section). I’m going to use this information to fit two models in the next section and compare their performance.

Modeling the Data

To model the data, I am going to use a linear model. I will force the second model to be a 2 degree polynomial. I will evaluate the model on the R squared value to determine how much variation in deaths is predicted by the cases.

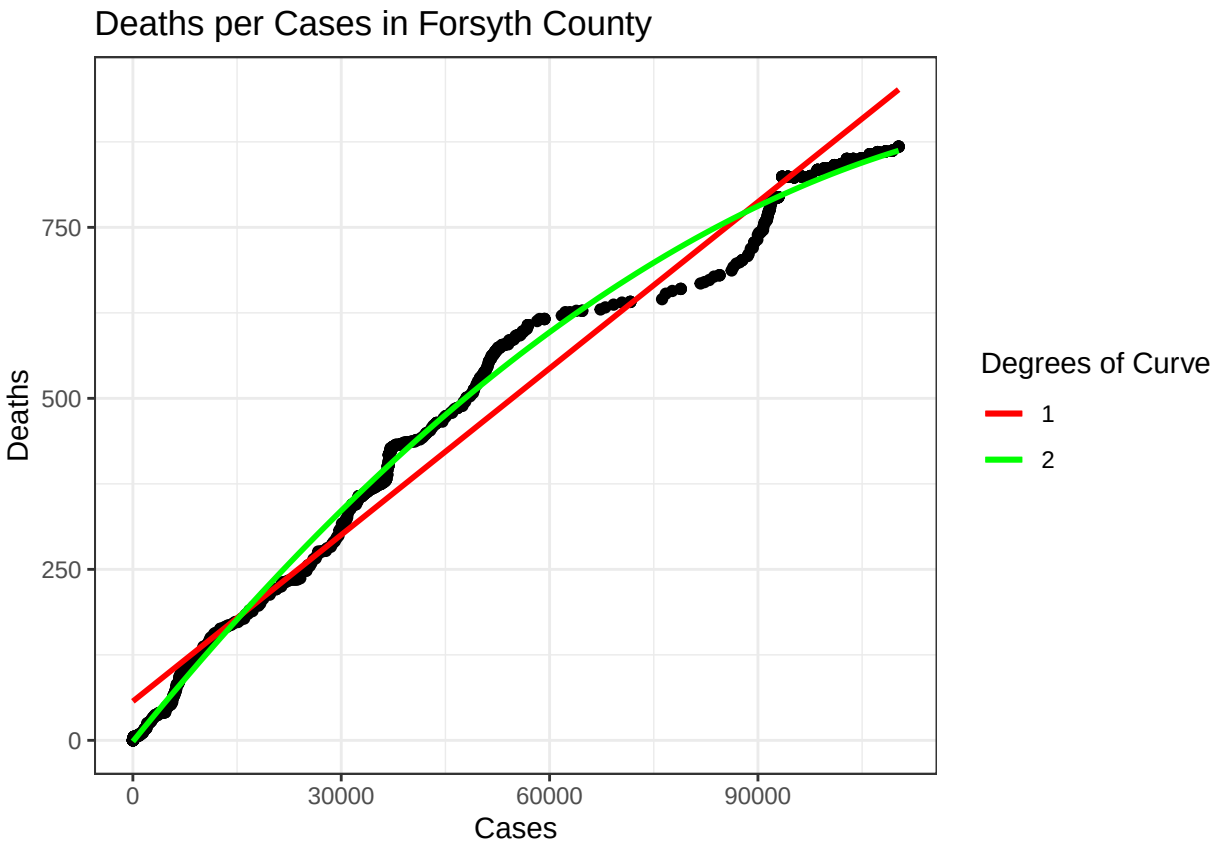

Figure 2: Deaths vs Cases in Forsyth, North Carolina

```
forsythData = usData %>% filter(Province_State == "North Carolina") %>%
  filter(Admin2 == "Forsyth")
model = lm(deaths~cases, forsythData)
summary(model)
```

```
##
## Call:
## lm(formula = deaths ~ cases, data = forsythData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -83.508 -46.902  -8.941  34.318  90.233
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.702e+01  2.519e+00   22.64   <2e-16 ***
## cases       8.112e-03  4.496e-05  180.43   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.7 on 905 degrees of freedom
## Multiple R-squared:  0.973,  Adjusted R-squared:  0.9729
## F-statistic: 3.255e+04 on 1 and 905 DF,  p-value: < 2.2e-16
```

```
forsythData = usData %>% filter(Province_State == "North Carolina") %>%
  filter(Admin2 == "Forsyth")
model2 = lm(deaths~poly(cases,2), forsythData)
summary(model2)
```

```
##
## Call:
## lm(formula = deaths ~ poly(cases, 2), data = forsythData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.986  -9.103   0.821  12.972  34.524
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       410.4421     0.6467  634.66   <2e-16 ***
## poly(cases, 2)1  8606.9030    19.4765  441.91   <2e-16 ***
## poly(cases, 2)2 -1310.1411    19.4765  -67.27   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.48 on 904 degrees of freedom
## Multiple R-squared:  0.9955, Adjusted R-squared:  0.9955
## F-statistic: 9.991e+04 on 2 and 904 DF,  p-value: < 2.2e-16
```

As we can see in the above data, the first degree fit has a R squared value of 0.973 and the second degree fit has an r squared value of 0.9955. This states that the residual error is less with the 2 degree polynomial fit. I will plot the residuals

```
model1DF = data.frame(fitted_Model = fitted(model),residual_Model = residuals(model))
model1DF['Model'] = "1 Degree Poly"


model2DF = data.frame(fitted_Model = fitted(model2),residual_Model = residuals(model2))
model2DF['Model'] = "2 Degree Poly"

modelDF = rbind(model1DF, model2DF)
#head(modelDF)


modelDF %>%
  ggplot(aes(x = fitted_Model, y = residual_Model)) +
  geom_point(aes(color = Model)) +
  facet_wrap(~Model) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal()+
  labs(title = "Plot of Residuals of Models", x = "Actual Deaths Value (Fitted Values)", y ="Residual o
```

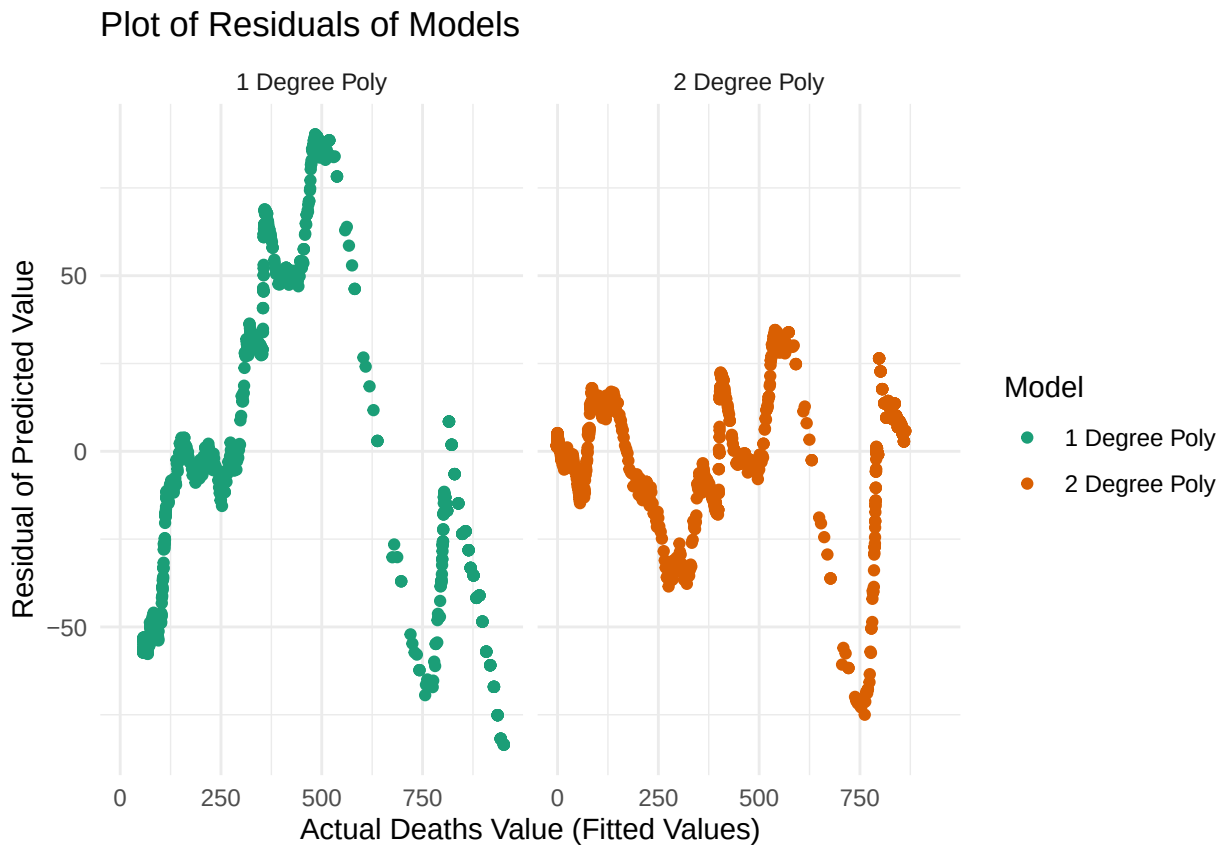

Figure 3: Model Residuals

In the figure "Model Residuals", we can see that it verifies the R squared value. The residuals of the second degree polynomial has less error, an exception being around the 750 deaths mark. If I were to make predictions on the number of deaths, based off the cumulative cases, I would choose to use the 2nd degree polynomial model since it had the least amount of error.

Potential sources of Bias

There are several potential error sources and bias in the data.

Initially, at the beginning of the pandemic, it was difficult to get tested unless you were critically ill. This would decrease the number of cases that were reported.

Also, in January of 2022, home rapid were mailed out to individuals. This would have a potential effect of increasing the number of mildly ill, or non-symptomatic people to get officially tested. It could may have had an effect in the other direction where people just took the rapid tests without reporting it. Further research would need to be done to determine how readily available rapid tests effected the reporting rate.

For future research and analysis opportunities, I would like to get the vaccine data and testing positivity rate and overlay it on the charts. I think this may help highlight some of the cases vs deaths in the data, especially where the discrepancy became significant as the pandemic went on