ANALYZING MEASURES OF SPREAD: COVID-19 IN CONNECTICUT

Jessica Chipera

Northcentral University: School of Technology

Course Number: TIM-8501

Dr. Nicholas Harkiolakis

September 4, 2022

Introduction

This paper uses exploratory data analysis to evaluate the measure of spread, given three years of COVID-19 data from Connecticut Health and Human Services Department. As always, I first imported some standard Python libraries I thought I might need, and the CSV file containing my data (Mukhira & Ahmed, 2020).

Variables and Types

After viewing the data, I identified that I could compare "Town" and "Total Cases" as well as "Total Cases" and "Total Deaths" and possibly "Town" and "Positive Tests" and other combinations of the same variables in order to analyze this as a scientific study of viral spread. Medical case studies analyzing spread also factor time as a variable into the analysis, so it could also be useful for me to do likewise (CDC Museum COVID-19 Timeline).

Except for "town" which is categorical data, and "rate tested per 100k" which is a ratio, all data in the set $n \geq 0,\ n \in \mathbb{Z}$ is discrete (365 Careers, 2018; Gut, 2013). In short, any data that is counted is discrete; data that is calculated is usually continuous. Our data contains integers (there are no fractions of people, cases, deaths, or positive Covid tests), which means it is counted and therefore discrete.

Data Shape

The dataset is 102,245 rows by 16 columns. Disease spread data is usually shown in relation to time passing (CDC Museum COVID-19 Timeline). Therefore, I had to add another column that could show the passage of time (Mukhira & Ahmed, 2020). With the provided dates, I was able to add "days ago," which means the frequency distribution graph below (Figure 1)

shows the total cases over time, with the most recent data first. In other words, Figure 1 is a mirror image, with 2020 on the right:

Figure 1

```
In [15]: sns.displot(df['days_ago'], kde=True, height=5, aspect=2)
         plt.title('Total Cases Over Time', size=16, )
         plt.ylabel('total_cases');
```

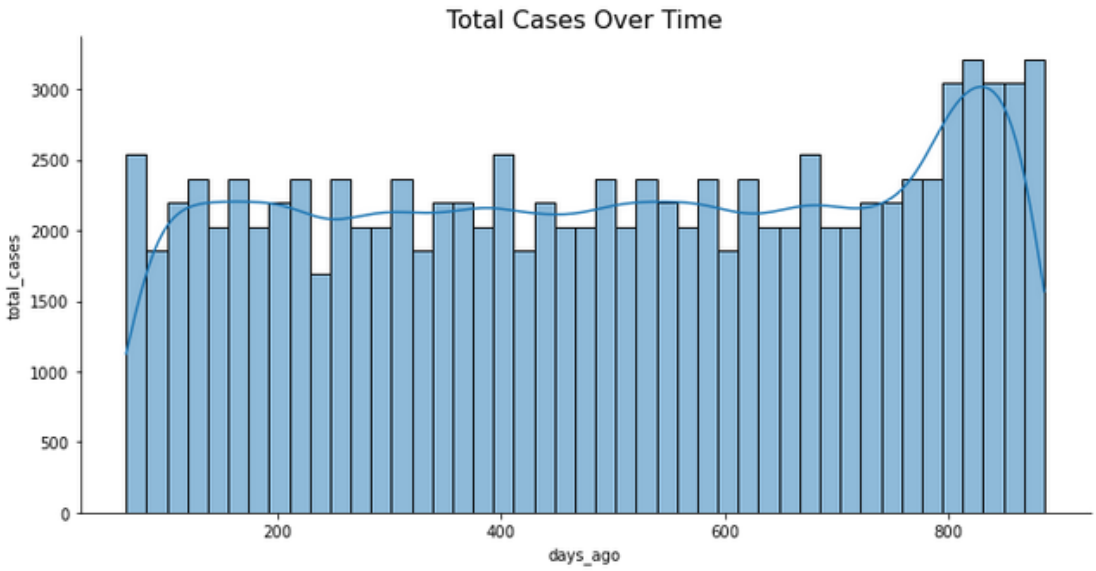


Figure 1 is mostly uniform frequency distribution, but it does have a slight left skew. It does not fit uniform or Poisson distribution, but it has aspects of both. Poisson distribution gives the probability that a number of events will occur given a fixed amount of time, and uniform implies that an event (Covid-19) is ongoing or endemic into the population (365 Careers, 2018; Gut, 2013).

Recall that Poisson frequency distribution obeys some properties, of which the one of most importance to this paper is that the number of trials “n” is indefinitely large. Yale researchers funded by the National Science Foundation have declared that Covid-19 is not yet endemic to the population (Croft, 2022; Locklear, 2022), but the frequency distribution of our data might be implying that it is endemic. Uniform frequency distribution implies that an event is

equally likely to occur randomly, which would only be the case if Covid-19 were endemic. Similarly, Poisson distribution shows that there is an infinitely large number of points in our probability space $(\Omega, \mathcal{H}, \mathbb{P})$ (Gut, 2013).

$$with\ \mathbb{P} \neq 0$$

$$\mathcal{H}\ is\ a\ sigma\ algebra$$

The large occurrence (x = 800) on Figure 1 is March, 2020 when the virus was at its peak in Connecticut, and when the United States was locked down (CDC Museum COVID-19 Timeline). The below equation defines Poisson distribution (Gut, 2013):

$$f(x) = \mathbb{P}(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}, x = 0, 1, 2, \dots, and\ \lambda > 0$$

$$Where\ X = the\ number\ of\ cases$$

$$and\ \lambda = \ the\ rate\ of\ cases\ in\ the\ long\ run.$$

$$Therefore, \lim_{n\to\infty} X \sim Poisson(\lambda)$$

We can see that this satisfies our stated set $n \geq 0,\ n \in \mathbb{Z}$ with the subset showing exponential growth represented by the section near (x = 800).

In order to more clearly understand the frequency distribution of the data in relation to the cities of Connecticut, a strip plot was created to more closely examine the spread of the virus in each city and spot potential outliers as they relate to the whole dataset (Mukhira & Ahmed, 2020). This strip plot is Figure 2, on the next page:

[98]:
plt.figure(figsize=(36,15))
sns.stripplot(x='Town', y='total_cases', data=df)
plt.title('Cases per Town', size=26)
locs, labels = plt.xticks()
plt.setp(labels, rotation=90)
plt.show()
Cases per Town
total_cases
0
10000
20000
30000
40000
Town
Sharon
Cheshire
Plymouth
Pomfret
Portland
Preston
Prospect
Putnam
Redding
Bridgeport
Thompson
Lyme
Plainville
Marlborough
Rocky Hill
New Milford
Avon
Manchester
Burlington
Hartford
Bolton
West Hartford
East Hartford
Harwinton
Litchfield
Kent
Warren
Sterling
Farmington
Plainfield
Andover
Canterbury
Watertown
Bethlehem
Sherman
Franklin
Sprague
Southington
Berlin
Colchester
Waterbury
Woodbury
Norwich
Roxbury
Middletown
Bozrah
Meriden
Middlebury
Bridgewater
East Haddam
Haddam
Naugatuck
Durham
Oxford
Lisbon
Voluntown
East Hampton
Griswold
Wolcott
Cromwell
Willington
Windsor
Middlefield
Salem
North Stonington
Montville
Ledyard
Newtown
Bethany
Beacon Falls
Danbury
Hamden
East Lyme
Killingworth
Madison
Stonington
North Haven
Chester
Guilford
Bethel
Deep River
Woodbridge
Monroe
New London
Essex
Old Lyme
East Haven
Ansonia
Ridgefield
Shelton
Brookfield
Wallingford
Southbury
Derby
New Haven
Westbrook
Old Saybrook
Clinton
Easton
Branford
Orange
Trumbull
Milford
Weston
Stratford
Wilton
Fairfield
New Canaan
Westport
Stamford
Norwalk
Greenwich
Darien
Waterford
North Branford
Seymour
Groton
West Haven
New Fairfield
Salisbury
North Canaan
Norfolk
Colebrook
Hartland
Woodstock
Canaan
East Granby
Barkhamsted
Winchester
Eastford
Ashford
Ellington
Windsor Locks
East Windsor
Windham
Scotland
Glastonbury
Columbia
Hebron
Wethersfield
Newington
Bristol
Morris
New Britain
Lebanon
Washington
Tolland
Canton
Simsbury
Cornwall
Goshen
Granby
Suffield
Enfield
Somers
Stafford
Union
Killingly
Bloomfield
New Hartford
Torrington
Vernon
South Windsor
Coventry
Mansfield
Chaplin
Hampton
Brooklyn
Thomaston

Figure 2

Anomalous Data & Relationships

In Figure 2, the x-axis is the town name, and the y-axis is the total cases. As the data shows, there are 9 cities that had the largest number of cases, and these cities will heavily influence the mean (365 Careers, 2018; Gut, 2013). They are: Bridgeport, Hartford, Waterbury, Meriden, Danbury, New Haven, Stamford, Norwalk, and New Britain. We will henceforth call these "outbreak cities." As is expected, the outbreak cities are those with higher populations.

It's important to analyze the relationship between cases and deaths from the virus. Since Covid 19 did not kill every person whom it infected, any data including ours should reflect deaths < cases. In fact, we can see from the mean that the variables satisfy the requisite inequality.

Figure 3

```
In [43]: df.total_cases.describe()

Out[43]: count    90753.000000
         mean      2095.054356
         std       4082.136179
         min          0.000000
         25%        191.000000
         50%        700.000000
         75%       2111.000000
         max      42752.000000
         Name: total_cases, dtype: float64

In [44]: df.total_deaths.describe()

Out[44]: count    90753.000000
         mean        45.237325
         std         68.422667
         min          0.000000
         25%          3.000000
         50%         17.000000
         75%         56.000000
         max        503.000000
         Name: total_deaths, dtype: float64
```

Approximately 11% of the rows are missing data in one or more columns. Thankfully, the majority of missing data are in the same rows, and not in columns that are used in this analysis, so the missing data can either be kept (Mukhira & Ahmed, 2020).

To compute standard deviation and variance for discrete variables, we would use summation notation instead of a Reimann integral (Gut, 2013). The code is shown in Figure 3 on the previous page (Mukhira & Ahmed, 2020). We can use the mean and standard deviation for analysis. Since the frequency distribution of our data is skewed, the mean will also have a skew (365 Careers, 2018; Gut, 2013).

Findings & Insights

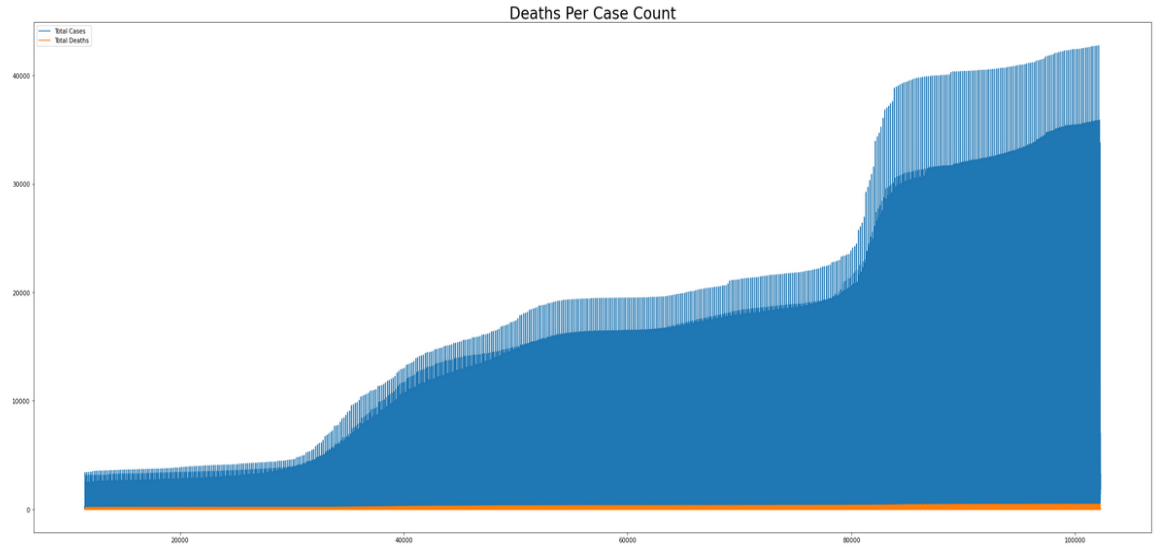
Over the timespan of the collected data sample, we can calculate a ratio of deaths per cases, given the mean of each variable. That ratio is 2.15%. This is surprising because it is close to the televised ratio given the original Covid-19 variant in 2020 (CDC Museum COVID-19 Timeline), and approximately 81% of Connecticut residents have gotten one of the available vaccine options, according to (Mayo Clinic, 2022).

Further analysis would be needed to definitively say whether or not the vaccine was successful, and that is beyond the scope of this paper. However, we can in Figure 4 that deaths grew linearly while cases grew nonlinearly. In other words, the two functions have different gradients and limits. This is what would be expected if the virus became less deadly as time passed or the population gained immunity to said virus as time passed, or both. It also implies that, since deaths are remaining low, it might not matter whether Covid-19 is endemic or not (Mayo Clinic, 2022).

Figure 4

```
In [62]: # create data
x=df.last_update_date
y=df.count
df2 = df.total_cases
df3 = df.total_deaths

# plot lines
plt.figure(figsize=(36,15))
plt.title('Deaths Per Case Count', size=26)
plt.plot(df2, label="Total Cases")
plt.plot(df3, label="Total Deaths")
plt.legend()
plt.show()
```



Subsequent Actions

Given the implications of this data, it might be useful to recall the medical test paradox, also known as Bayes' Theorem (Gut, 2013) to make some further inferences. In short, Bayes' Theorem shows that because of probability, an accurate Covid-19 test is not necessarily a predictive test. This could be useful if one wants to attempt to use Covid-19 data to create a machine learning algorithm to predict the spread of subsequent pandemics or if this data were to be used for judging the effectiveness of the vaccine effort.

$If\ (\Omega, \mathcal{H}, \mathbb{P})\ is\ a\ probability\ space$

$Subsets\ A, B \in\ \Omega, with\ \mathbb{P}(A) \neq 0\ and\ \mathbb{P}(B) \neq 0$

$\mathcal{H}\ is\ a\ sigma\ algebra$

$Let\ \mathbb{P}(A) = Had\ Covid$

$Let\ \mathbb{P}(B) = Positive\ Test$

$Thus, \mathbb{P}(A|B)$ *defines the probability that someone who got a positive result from the Covid-19 test actually has the disease.*

Our sample contains 172,716,911 Covid-19 cases in Connecticut, 232,289,264 positive tests, and 4,265,321,308 negative tests. (Notice that $negative\ tests > positive\ tests \geq total\ cases$, as expected.) One might point out that we are missing the subset of people who had Covid and did not take a test. For now, we will omit the people who didn't take a Covid test because they are members of a subset of Ω but not $\mathbb{P}(\cdot\ |B)$: $\mathcal{H} \rightarrow [0,1]$, a conditional probability measure given B, where " $\cdot$ " is a placeholder. In other words, we are excluding the people who did not take a Covid test because the test has only two results, either positive or negative; the people who didn't take the test are irrelevant to this measure space (Gut, 2013).

To analyze test accuracy, one must know the false negative rate $\mathbb{P}(A|B^C)$, as well as the false positive rate or $\mathbb{P}(B|A^C)$ where B^C is the complement of B and A^C is the complement of A. According the University of Massachusetts Medical School, the sensitivity (false negative rate) and specificity (false positive rate) of the Roche Covid-19 Antibody Test were 65.3%, and 99.8% respectively for both the delta and omicron variants (Hafer, N. & Soni, A, 2021). In other words, $\mathbb{P}(A|B) = 0.653$ and $\mathbb{P}(B|A) = 0.998$.

Therefore, we know that $\mathbb{P}(A|B^C) = 34.7\%$, or more than 80 million tests of 232 million returned a false positive result. $\mathbb{P}(B|A^C) = 0.2\%$, so, 8.5 million tests out of 4.3 billion returned

a false negative. This means that Connecticut had more people quarantining than were sick. While this seems like the preference over the alternative, it could have cost the state funding for unemployment and other spending initiatives that it otherwise wouldn't have had to spend.

Though it is outside the scope of this paper, it would be interesting to experiment with building an algorithm to track the virus and compare its results with that of the medical test to see if the algorithm can be more predictive of spread and therefore potentially helpful in future pandemic situations.

References

365 Careers. (2018). Statistics for data science and business analysis. Packt Publishing.

CDC Museum COVID-19 Timeline. (https://www.cdc.gov/museum/timeline/covid19.html)

Croft, J. (2022). COVID-19 Could Become Endemic in 2024, Study Says. WebMD.

Deisenroth. M et all (2020). Mathematics for Machine Learning. Cambridge University Press.

Gut, A. (2013). Probability: A Graduate Course. Second Ed. Springer Science+Business Media.

Hafer, N. & Soni, A. (2022). Just how accurate are rapid antigen tests? Two testing experts explain the latest data. UMass Chan Medical School.

Locklear, M. (2022). For COVID-19, endemic stage could be two years away. Yale News.

Mayo Clinic. (2022). Coronavirus COVID-19 Map. https://www.mayoclinic.org/coronavirus-covid-19/map

Mukhiya, S. K., & Ahmed, U. (2020). Hands-on exploratory data analysis with Python. Packt Publishing.