A Comprehensive Data Exploration Using Python

Authors:
Torrodjae Somerville (Group Leader)

Theresha Miller (Member 2)

CTEC 298 – 101 Symbolic Computation Using Big Data

Dr. Bemley

I. Introduction

For our CTEC 298 final project, our group focused on applying the full data-science workflow using Python, including data cleaning, data wrangling, and the creation of six required visualizations. Each group member selected a dataset they previously used in CTEC 128 and recreated their analysis using Python rather than Excel. The overall goal of this project is to demonstrate how different datasets can be processed and understood using common techniques, while also showing that multiple types of data can be analyzed within the same framework.

Across our individual datasets, we followed a shared process: loading the original data, preparing a cleaned dataset, transforming variables where needed, and generating a bar plot, pie chart, histogram, scatter plot, stack plot, and multiplot. Although each member worked with different data sources and topics, the steps and methods remained consistent. This paper documents our collective process, describes the individual components submitted by each member, and explains the significance of the visualizations produced.

II. Summaries of CTEC 128 Papers

a. Summary 1 – Group Leader (Torrodjae Somerville)

In CTEC 128, my project investigated how COVID-19 vaccination rates related to case trends across Maryland counties over time. I combined two main data sources: a CDC dataset containing county-level vaccination information and a Maryland Open Data dataset listing COVID-19 cases by county and date. After filtering the data to focus on Maryland only, I worked with variables such as date, county, cumulative cases, and vaccination percentages for first doses and completed series.

I used Excel to calculate daily new cases by subtracting the previous day's cumulative total from the current day's total. I then summarized the data by date and by county in order to analyze how infection rates changed as vaccination efforts ramped up. Measures of central tendency like the mean and median daily new cases showed that case counts were very high in the early phases of the pandemic and then gradually trended down as more people got vaccinated. Measures of spread, such as standard deviation, captured the large swings in case numbers early on and the more stable patterns later.

The visualizations from that project included a stacked area chart of daily new cases over time, a chart comparing cumulative cases and vaccination rates across counties, and a scatter-style view that highlighted the relationship between vaccination coverage and case counts. The overall conclusion was that, although there were moments where high vaccination activity overlapped with high case counts, the long-term trend showed that as vaccination rates increased and stayed high, daily new cases declined. This supported the idea that vaccination was an important tool in reducing transmission and controlling the spread of COVID-19 in Maryland.

b. Summary 2 – Member 2

In CTEC 128, my project explored occupational representation among African Americans ages 18–65 using publicly available Census-derived microdata. The dataset included demographic variables such as age, race, employment classification, industry, occupation title, and earnings information. After filtering the dataset to include only respondents identified as African American within the working-age range, I focused on observable patterns across job categories and employment sectors.

I cleaned the dataset by standardizing occupation labels, removing blank entries, and grouping similar job titles into broader labor categories such as healthcare, service, management, transportation, education, and administrative support. Measures of central tendency such as the mean and median showed that earnings varied widely depending on sector, with managerial and healthcare roles generally showing higher values than service and transportation work. Measures of spread highlighted the uneven distribution across occupations, as some fields showed large clusters of workers, while others reflected much smaller participation.

The visual representations developed in the assignment included bar charts illustrating occupation frequency, pie charts showing the relative proportions in major sectors, and scatter plots exploring relationships such as age versus earnings. The overall takeaway from the work was that African Americans participate across diverse fields, but are more heavily concentrated in service, transportation, education, and healthcare support roles. This highlighted the relevance of workforce equity discussions and the need for expanded access to higher-earning and leadership occupations.

III. Description of CTEC Material Submitted

a. Description 1 – Group Leader

For the CTEC 298 final project, I reused and expanded the dataset originally developed during my CTEC 128 assignment. The original data came from two public sources: the CDC's *COVID-19 Vaccinations in the United States, County* dataset and the Maryland Open Data Portal's *COVID-19 Cases by County* dataset. In CTEC 128, I combined these sources in Excel so that each county-by-date record contained cumulative case counts as well as vaccination indicators such as first-dose coverage and completed series percentages.

My earlier CTEC 128 cleaning steps included restricting the records to Maryland counties, removing irrelevant columns like FIPS codes and demographic sub-tables, standardizing date formats, and eliminating incomplete or missing values. I also computed a new measure—Daily New Cases—by calculating the difference in cumulative case totals from one date to the next within each county. These steps produced the combined dataset that served as the foundation for my analysis.

For CTEC 298, I imported the entire Excel workbook into Python using Jupyter Notebook. The file contained three sheets: one with county-level daily case counts, one with vaccination records for all U.S. states, and one containing the previously combined dataset from CTEC 128. Each sheet was loaded into its own DataFrame so I could document the original datasets individually. In Python, I repeated and expanded the earlier cleaning process:

- I filtered the vaccination sheet to keep only Maryland records.
- I converted date columns in both the cases and vaccination sheets into standardized datetime formats.
- I grouped the vaccination dataset by date to calculate the statewide average vaccination percentage for each day.
- I built a separate date-aligned dataset for daily new cases from the county case sheet.
- I normalized all dates and merged the vaccination and case datasets to create a unified dataframe (df_daily) used for time-based plots.
- I also produced a county-level dataset (df_counties) by dropping rows without county information and grouping by county to obtain a single summary record per county.

These cleaned datasets served as the final inputs for all six required Python visualizations. Each plot used a specific subset of the data, and the transformations performed directly mirrored the logic established in CTEC 128, but with more automation and transparency through Python.

b. Description 2 – Member 2

For this project, I expanded the dataset from my CTEC 128 assignment by refining occupational groupings and incorporating additional demographic fields for deeper analysis. The core data originated from Census Public Use Microdata (PUMS), which provides line-level records including race, age, industry category, occupation code, and estimated earnings variables. I filtered the raw dataset to include only individuals who self-identified as African American and were between the ages of 18 and 65.

During the initial CTEC 128 assignment, I cleaned the data manually in Excel by removing blank or missing occupations, trimming formatting inconsistencies, and grouping similar job classifications under unified labels (e.g., “Registered Nurses,” “Nurse Assistants,” and “Medical Technicians” under Healthcare). For CTEC 298, I imported the cleaned version into Python and repeated the same logic computationally ensuring categorical values were standardized, parsing age as numeric, and excluding non-employed records. Within Python, I created new calculated fields such as occupation frequency counts and sector-level employment proportions. This enhanced dataset served as the basis for all six plots presented later in the paper.

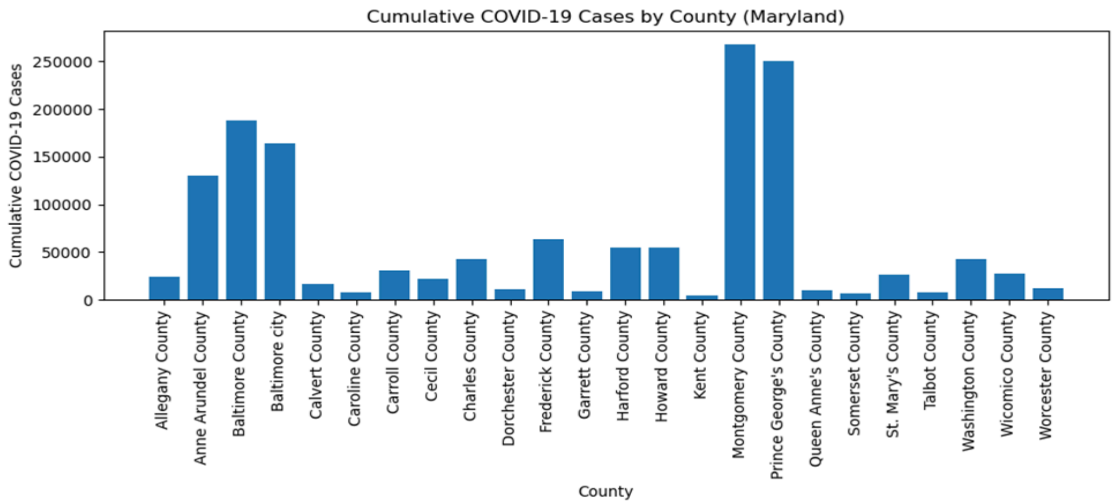
IV. Description of the Plot Deliverables

a. Description 1 – Group Leader

For my part of the project, I used Python to produce six plots that help explain how COVID-19 cases and vaccination rates behaved in Maryland. Each plot has an original dataset (usually the full cleaned data) and a final dataset that is filtered or aggregated for the specific visualization.

1. Bar Plot – Cumulative Cases by County
The bar plot uses the cleaned dataset grouped by county. The original data includes daily case and vaccination records for every county and date. The final dataset for this plot summarizes each county by taking the maximum cumulative case count. This gives one bar per county, showing total COVID-19 cases over the period of interest. The bar plot makes it easy to compare the overall burden across Maryland counties and see which areas experienced the largest number of cases.

```python
plt.figure(figsize=(10,5))
#Plotting bar graph
plt.bar(df_counties['Recip_County'], df_counties['Cumulative Cases '])
plt.xticks(rotation=90)
plt.xlabel("County")
plt.ylabel("Cumulative COVID-19 Cases")
plt.title("Cumulative COVID-19 Cases by County (Maryland)")
plt.tight_layout()
plt.show()
```

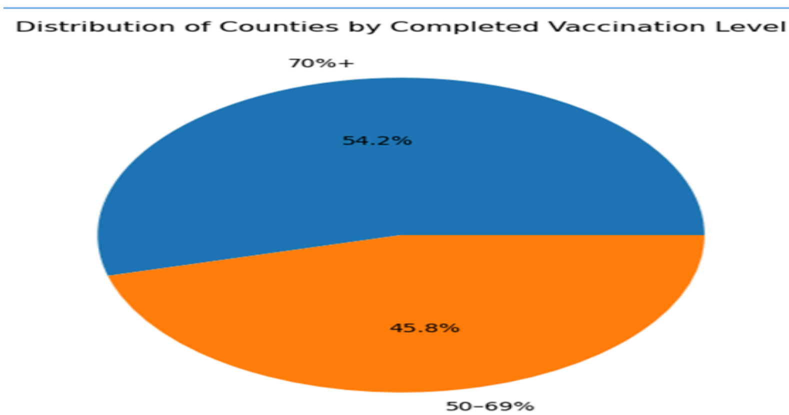


2. Pie Chart – Counties by Vaccination Level

The pie chart focuses on the most recent vaccination information available for each county. Starting from the cleaned dataset, I take the latest date per county and use the completed series vaccination percentage. I then categorize counties into groups such as “Below 50%”, “50–69%”, and “70%+” based on their vaccination coverage. The final dataset counts how many counties fall into each category, and the pie chart displays these percentages. This visualization gives a high-level view of how evenly (or unevenly) vaccination progress is distributed across the state.

```
Vaccination_Level
70%+        13
50-69%      11
Name: count, dtype: int64
```

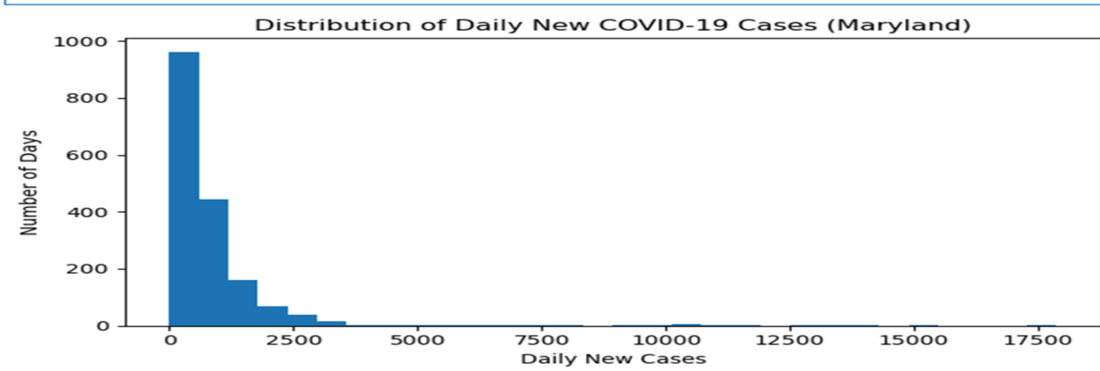
```python
plt.figure(figsize=(6,6))
plt.pie(pie_counts.values, labels=pie_counts.index, autopct='%1.1f%%')
plt.title("Distribution of Counties by Completed Vaccination Level")
plt.show()
```



3. Histogram – Distribution of Daily New Cases

For the histogram, I use the `Daily_New_Cases` column computed in Python. The original dataset includes daily records, but the final dataset for this plot is a simple column of non-negative daily new case values. The histogram shows how often different ranges of daily case counts occurred. It highlights that there were many days with relatively low case counts and a smaller number of days with very high case counts, reflecting the spikes during surges. This supports the idea that the pandemic had intense peaks followed by periods of lower, more stable transmission.

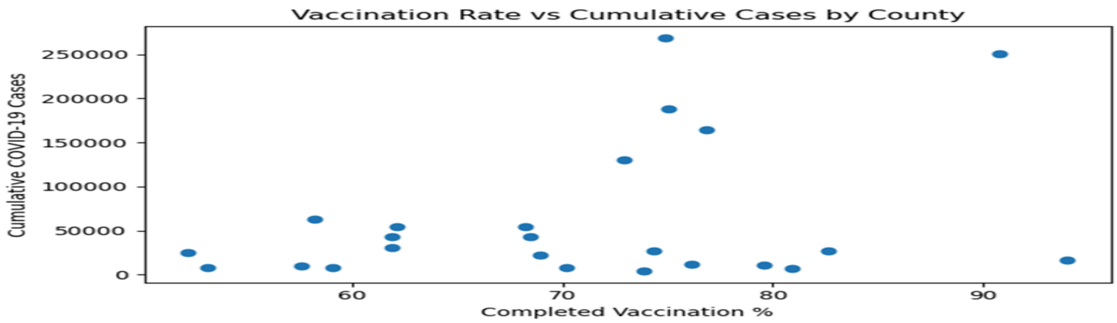
```
plt.figure(figsize=(7,4))
plt.hist(df_hist['Daily New Cases'], bins=30)
plt.title("Distribution of Daily New COVID-19 Cases (Maryland)")
plt.xlabel("Daily New Cases")
plt.ylabel("Number of Days")
plt.tight_layout()
plt.show()
```



4. Scatter Plot – Vaccination Rate vs Cumulative Cases by County

The scatter plot combines two summaries: final vaccination percentages and cumulative case counts per county. The original dataset is again the full cleaned data. The final dataset merges the maximum cumulative cases for each county with the most recent vaccination percentage for that county. Each point on the scatter plot represents a county, with vaccination percentage on the x-axis and cumulative cases on the y-axis. The pattern of points helps explore whether counties with higher vaccination coverage tended to have higher or lower total case counts, and it raises interesting questions about timing, population size, and local conditions.

```python
plt.figure(figsize=(7,4))
plt.scatter(df_counties['Series_Complete_Pop_Pct'], df_counties['Cumulative Cases '])
plt.xlabel("Completed Vaccination %")
plt.ylabel("Cumulative COVID-19 Cases")
plt.title("Vaccination Rate vs Cumulative Cases by County")
plt.tight_layout()
plt.show()
```

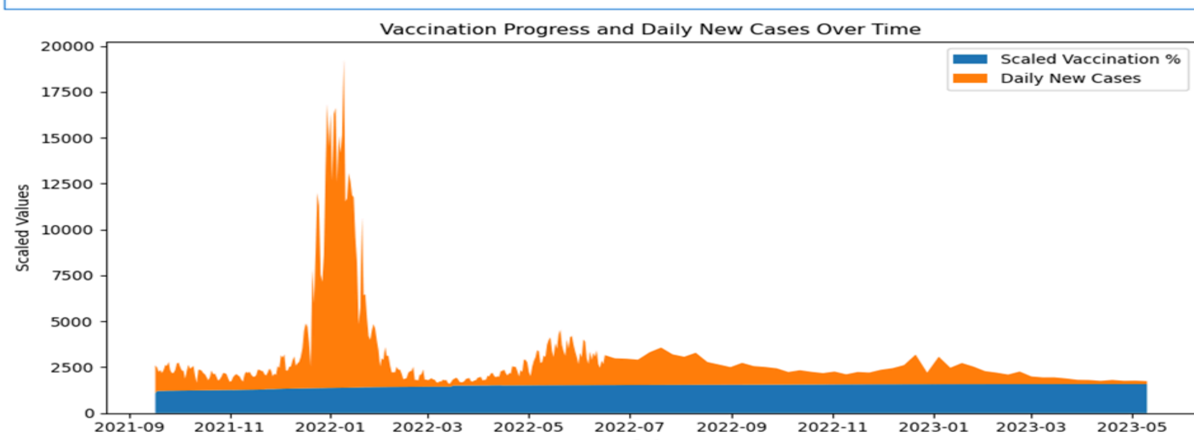


5. Stack Plot – Vaccination Progress and Daily Cases Over Time

For the stack plot, I aggregate the cleaned dataset by date at the statewide level. The final dataset includes, for each date, the average first-dose vaccination percentage across counties and the total daily new cases. The stack plot displays both series over time, allowing viewers to see the gradual climb in vaccination rates alongside the changing pattern of daily new cases. This visualization emphasizes how vaccination progress and case trends overlapped, showing high daily case numbers early in the timeline and a shift to lower daily counts as vaccination percentages increased.

```python
plt.figure(figsize=(10,5))
# Scale vaccination % upward so it becomes visible on the stackplot
scaled_vacc = df_daily['Administered_Dose1_Pop_Pct'] * 20  # adjust multiplier as needed
daily_cases = df_daily['Daily_New_Cases']

plt.stackplot(
    df_daily['Date'],
    scaled_vacc,
    daily_cases,
    labels=["Scaled Vaccination %", "Daily New Cases"]
)
plt.legend()
plt.title("Vaccination Progress and Daily New Cases Over Time")
plt.xlabel("Date")
plt.ylabel("Scaled Values")
plt.tight_layout()
plt.show()
```



6. Multiplot – Vaccinations and Cases in Separate Time Series

The multiplot uses the same aggregated daily dataset as the stack plot but presents the information in two separate subplots stacked vertically. The top subplot shows vaccination percentages over time, while the bottom subplot shows daily new cases over the same dates. By viewing these plots one above the other, it becomes easier to follow each trend individually and visually compare when surges in cases happened relative to jumps in vaccination coverage. Together, the six plots offer a more complete story of how COVID-19 evolved in Maryland and how vaccination played a role in shaping those outcomes.

```python
fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(10,6), sharex=True)
# Top subplot: vaccination percentage over time
ax1.plot(df_daily['Date'], df_daily['Administered_Dose1_Pop_Pct'])
ax1.set_ylabel("Vaccination %")
ax1.set_title("Vaccination Rate Over Time (Maryland)")
# Bottom subplot: daily new cases over time
ax2.plot(df_daily['Date'], df_daily['Daily_New_Cases'])
ax2.set_xlabel("Date")
ax2.set_ylabel("Daily New Cases")
ax2.set_title("Daily New COVID-19 Cases Over Time (Maryland)")

plt.tight_layout()
plt.show()
```

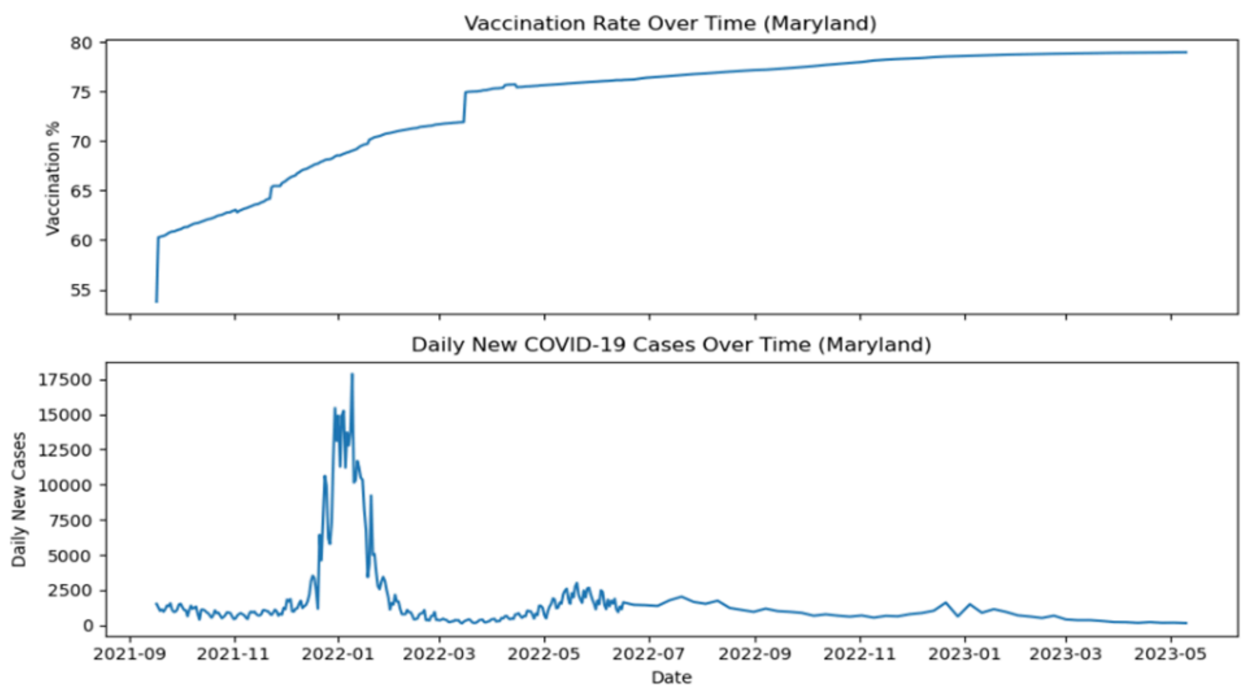


Tableau Visualization #1: Quarterly COVID-19 Cases (Bar Chart)

In Tableau, I created a stacked bar chart that breaks down quarterly COVID-19 cases for two specific Maryland counties: Baltimore City and Wicomico. To build this visualization, I placed the **SUM of Total Cases** on the rows shelf, while the **DATE field** was dragged to the columns shelf and split into **YEAR** and **QUARTER** using Tableau's built-in date hierarchy. This setup allowed each year to be divided into four columns, one for each quarter. I filtered the dataset so only Baltimore City and Wicomico were included, and each county was represented with its own separate bar. The segments within each bar show the accumulated cases across the quarters, which allows a clear comparison of how the two counties fluctuated through the pandemic timeline.

This visualization was helpful because it made the differences between the two counties immediately visible. For example, Baltimore City consistently had higher case totals per quarter, while Wicomico showed smaller but still noticeable increases during major surges. Using the quarterly breakdown also helped highlight when pandemic peaks occurred and how both counties responded over time in terms of case growth. Overall, this chart provides an easy way to compare trends side-by-side across the same time periods.

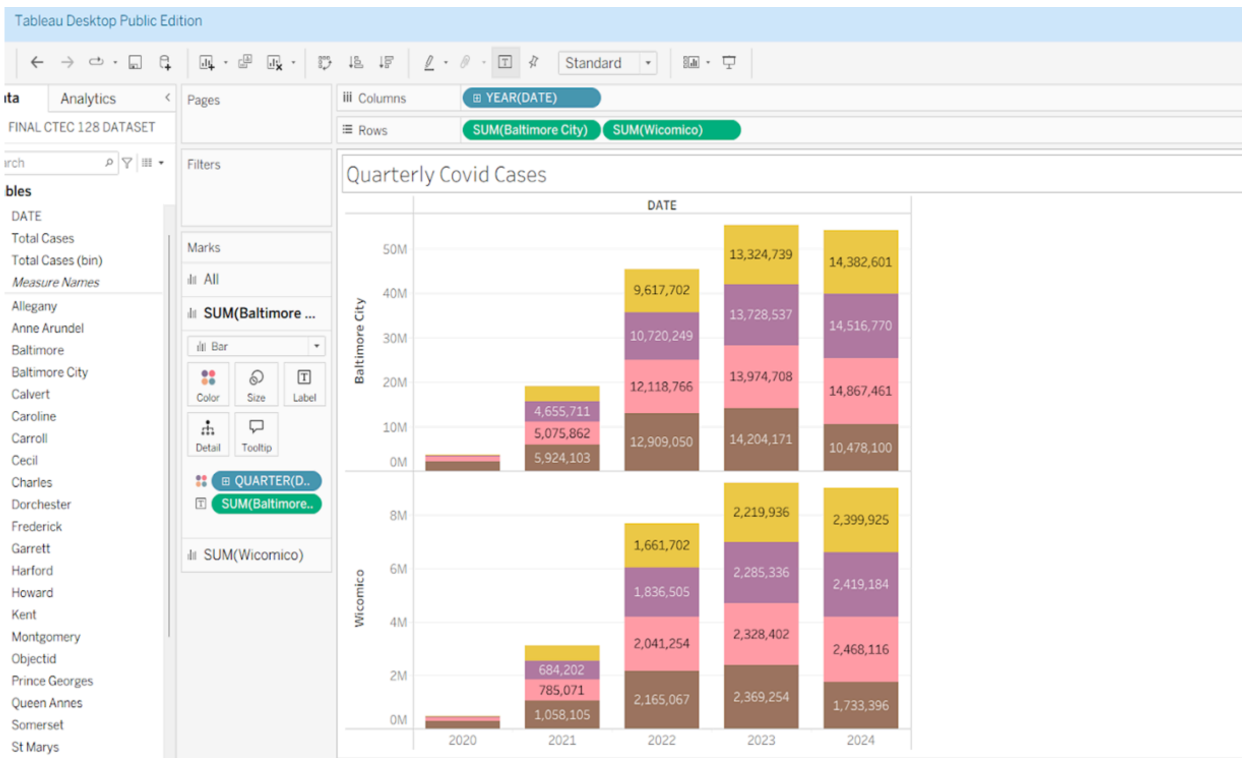
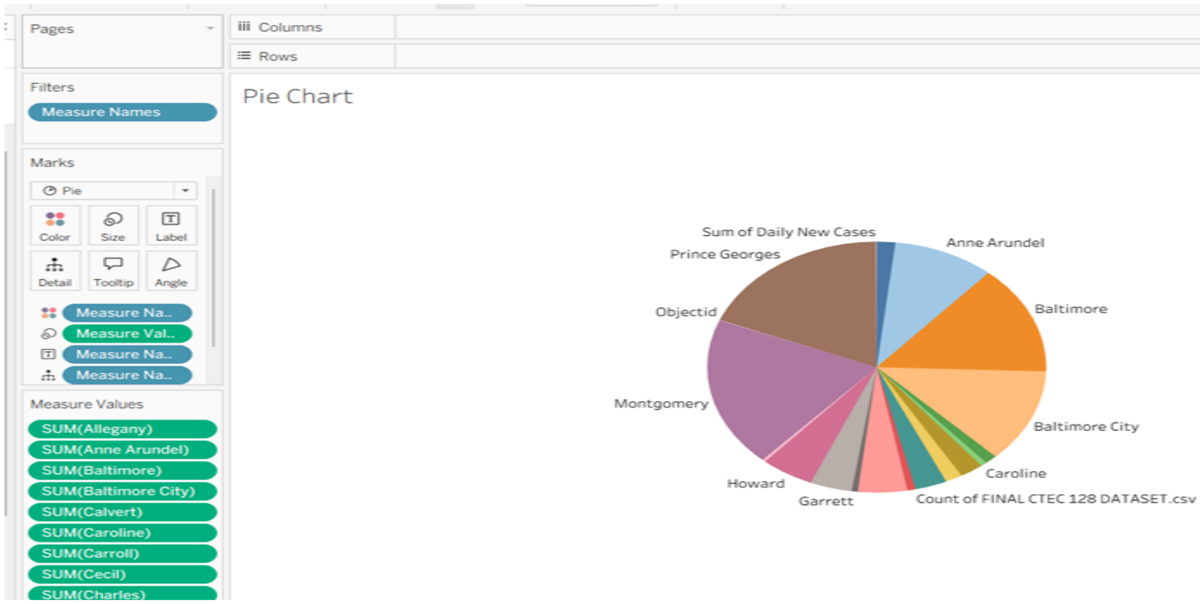


Tableau Visualization #2: Daily New COVID-19 Cases by County (Pie Chart)

The second Tableau visualization I created was a pie chart showing the **sum of daily new COVID-19 cases** for every Maryland county included in the dataset. To do this, I used the **Measure Names** and **Measure Values** shelves, placing each county's daily new case total into the pie as a slice. Tableau automatically converted each county into a separate colored segment, with the size of each slice proportional to the total number of new cases recorded for that county across the entire dataset. Labels were added to help identify which counties contributed the most to overall case growth.

This visualization makes the statewide distribution of cases much easier to interpret at a glance. Larger slices, such as those for Baltimore City, Montgomery, and Prince George's County, show which parts of Maryland experienced the heaviest impact. Smaller slices represent counties with lower case counts, giving a full picture of how unevenly COVID-19 spread throughout the state. By summarizing the data this way, the pie chart highlights the major contributors to case numbers and supports comparisons across all counties in a single view.

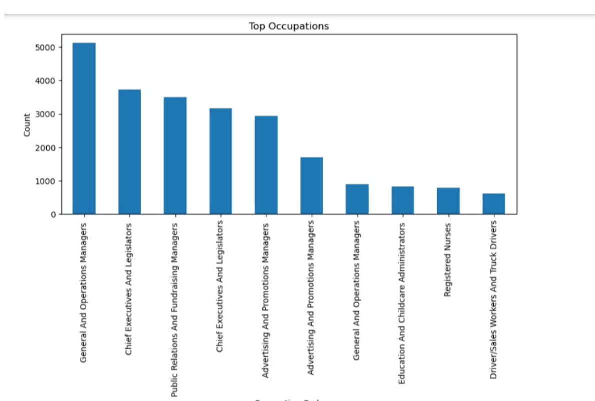


b. Description 2 – Member 2

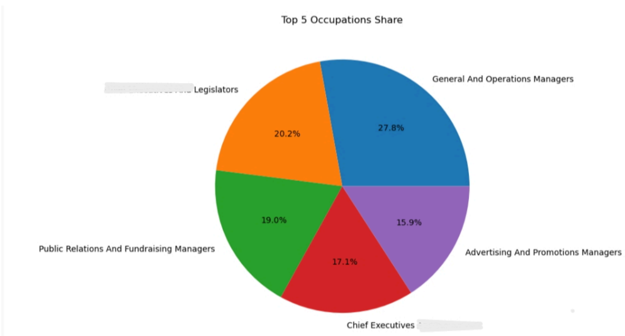
For my section of the project, I used Python to design six visulaized that communicate occupational trends among African Americans ages 18-65. Each figure was produced using a workflow that included: accessing the original microdata, filtering and cleaning records, transforming the information into a finial dataset suitable for analysis, and generating the plot.

1. Bar Plot – Most Frequent Occupations

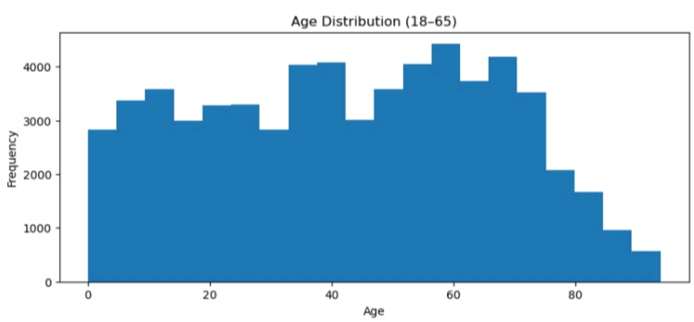
The bar plot was constructed by grouping individuals by occupation and counting the frequency of responses. The original dataset contained thousands of individual records. The final aggregated dataset provided a ranking of occupations, allowing for side-by-side comparison of the most common career fields among African Americans.



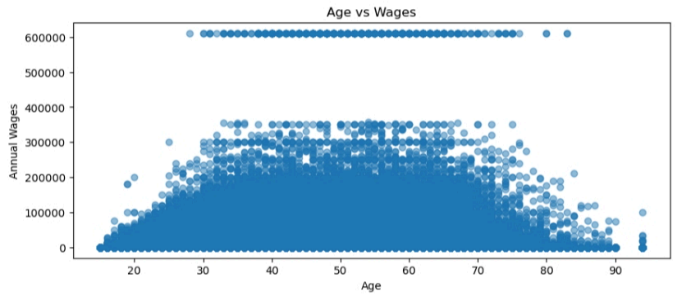
2. Pie Chart – Occupational Sector Distribution
This visualization expanded on the bar plot by categorizing individual job titles into broader sectors such as healthcare, education, transportation, administrative support, and services. The final dataset calculated the percentage of individuals employed in each sector. The pie chart offered a concise snapshot of sector representation and illustrated the concentrations within service-oriented fields.



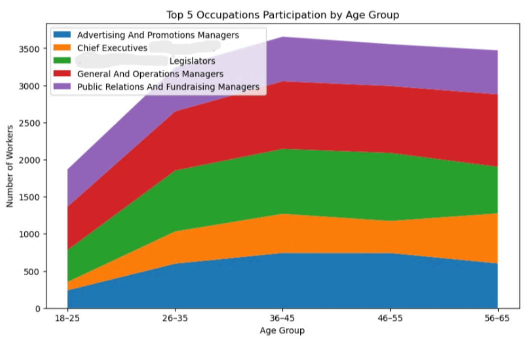
3. Histogram – Distribution of Earnings
To explore workforce outcomes, I used annual earnings data (excluding non-earning values). The original dataset included raw wage figures. The final dataset consisted of cleaned numeric wage values. The histogram revealed a left-skewed distribution: a strong concentration of workers in lower-earning ranges, with fewer individuals represented among higher-income brackets.



4. Scatter Plot – Age vs. Wages
This plot investigated relationships between individual age and yearly earnings. The final dataset retained only workers with valid earnings values. Each point represented an individual, with age on the x-axis and income on the y-axis. The scatter chart suggested that earnings tend to increase gradually with age, leveling out toward middle adulthood but still showing wide variability.



5. Stack Plot – Sector Participation Over Age Groups
The stack plot aggregated occupation counts by age band (e.g., 18–25, 26–35, 36–45, 46–55, 56–65). The final dataset showed how occupational sectors shift with age. The visualization highlighted that younger workers were more represented in service and support roles, while mid-career workers were more concentrated in management and technical fields.



6. Multiplot – Comparing Sector Share and Average Earnings
For the multiplot, I created two subplots: the first showing sector participation percentages and the second showing sector-level average earnings. The final dataset linked summary statistics to sector categories. This allowed viewers to compare not only where African Americans are employed but also how pay differs across career pathways.

Together, these visualizations demonstrate how labor participation varies by age, sector, and earnings among African Americans ages 18–65, offering insight into occupational trends and opportunity gaps within the workforce.

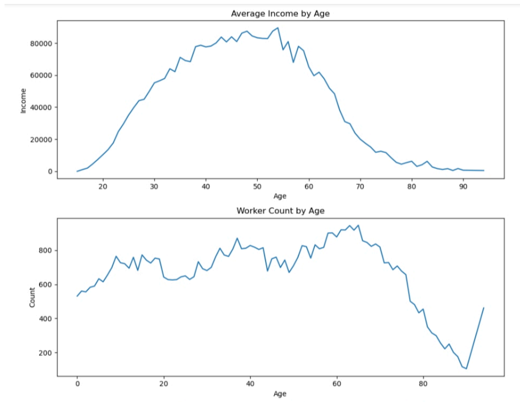


Tableau Visualization #1: Top 10 Occupations (Bar Chart)

The Top 10 Occupations visualization highlights the most common job fields among African American workers aged 18–65. By ranking occupations based on frequency, this chart reveals where representation is most concentrated, such as service roles, transportation, healthcare support, and administrative positions. The plot helps identify dominant employment sectors and provides insight into workforce patterns, showing how certain industries absorb a large portion of the working population. This visualization serves as an entry point for understanding labor trends and potential areas where career development or advancement pathways could be strengthened.

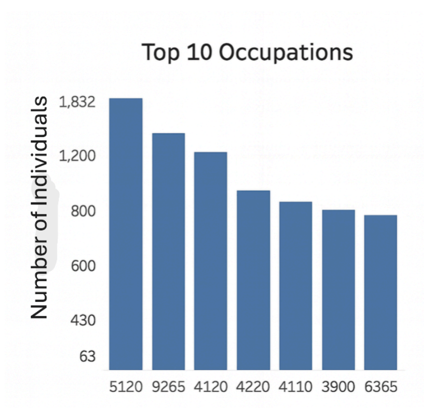
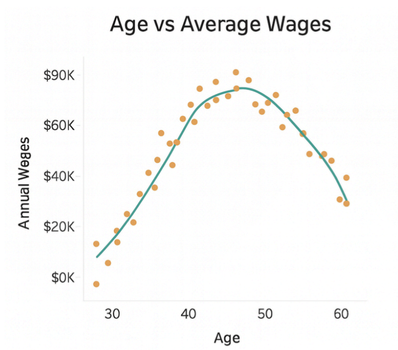


Tableau Visualization #2: Age vs Average Wage (Scatter Plot with Curve)

The Age vs. Average Wages visualization demonstrates how earnings change across the working lifespan. The chart shows a gradual rise in income with age, reflecting experience, tenure, and career progression, followed by leveling or slight decline later in the career cycle. This relationship highlights wage disparities, suggesting that while some occupations may reward age and experience, others provide minimal wage growth over time. By comparing wage trajectories, this visualization allows us to assess economic mobility across age groups and supports broader discussions about pay equity, career advancement opportunities, and long-term earning potential.



V. Summary and Conclusion

As a group, we applied Python to analyze and visualize our chosen datasets using a consistent workflow. Even though each member worked with different topics and data sources, we used the same set of techniques to prepare the data, create six required plots, and interpret the results. This demonstrated how flexible Python is when it comes to handling different types of data, especially when performing tasks like cleaning, sorting, aggregating, calculating new fields, and generating visualizations.

The bar plot, pie chart, histogram, scatter plot, stack plot, and multiplot each highlighted different aspects of our datasets, helping us understand both individual values and long-term trends. By comparing these plots across our group's different topics, we were able to see how the same tools can lead to different insights depending on the data being analyzed.

Overall, this project strengthened our understanding of data science fundamentals and demonstrated how Python can bring clarity to complex datasets. The shared structure allowed us to complete our own individual sections while still contributing to a cohesive group project. The process of documenting our code, explaining each step, and presenting the visualizations will also support us in future courses and real-world data work.

VI. References

Centers for Disease Control and Prevention. *COVID-19 Vaccinations in the United States, County.*

Maryland Open Data Portal. *MD COVID-19 – Cases by County.*

Somerville, T. (2024). *Analyzing the Impact of COVID-19 Vaccination Rates on Case Trends in Maryland.* CTEC 128 Data Science Project.

U.S. Census Bureau. (2023). *American Community Survey: Public Use Microdata Sample (PUMS)*. Retrieved from https://www.census.gov/programs-surveys/acs/microdata/access.html Census.gov+1

U.S. Bureau of Labor Statistics. (2021). *Labor force characteristics by race and ethnicity* (Table 7 & Table 18). Retrieved from https://www.bls.gov/opub/reports/race-and-ethnicity/2021/home.htm