# DATA1002 STAGE 2

*Note: 4 attribute chart produced as a group and will be counted same for each individual

**Topic of interest**

This topic was chosen to investigate the correlation between common socioeconomic factors including income, unemployment and education attainment and crime rates across the United States. A greater understanding of how the dataset varies and impacts each other enables us to make and extract crucial information regarding the prevention of crimes for many groups of people and improve societal well-being.

**Stakeholders**

The findings in this report would be potentially useful for:

Law Enforcement Agencies: Local and state law enforcement agencies have a vested interest in understanding the factors that influence crime rates. This crime factors analysis provides insights that help them allocate resources and develop strategies for crime prevention and reduction.

Government and Policymakers: State and local government officials and policymakers may use our findings to inform policy decisions related to education, economic development, and crime prevention.

Researchers and Academics: Other researchers and academics in the fields of criminology, sociology, economics, and public policy may find this analysis valuable for their own studies and research.

General Public: The general public may also be interested in understanding the factors influencing crime rates in their communities. This analysis can contribute to public awareness and discussions on these topics.

**Overview**

Our analysis aims to decipher the impact of variables like economic conditions, education levels, population density, and income distribution on the variances in crime rates across the U.S states from 2013-2019. By extracting and obtaining visualisations of subsets of the overall data through graphs, we are able to get better insight into the broader trends between external factors and how they have individually or collectively shaped the crime dynamics in various states.

Part A
Individual Sections
530511063 - Jack
Introduction:

The used dataset is the merged dataset from Stage 1, however a subset or more specific dataset was used in this section to explore specific relationships. These attributes are listed below:
“State”: Name of the U.S state
“Percentage with high school completion or higher of people age 25 and over”: %
“Per capita personal income”: in dollars
“Unemployment rate”: %

Grouped-Aggregation 1 (based on binned quantitative attribute):

First line creates a list called ‘bin’ defines the bin edges for binning the values from “Per capita personal income’. ‘np.inf’ is the upper bound for the last bin. Which includes values greater than or equal to 60000. In the second line, a new column ‘binning’ is added to the main data frame by binning the ‘Per capita personal income’ values based on the defined bin edges from the list ‘bin’. The data frame is then grouped by the ‘binning’ column and the mean of ‘Percentage with high school completion or higher (%) total’ for each bin is calculated. ‘reset_index()’ reset the index of the resulting data frame called ‘binned_df’, so the bin labels become regular columns again. A new data frame called ‘binned_df_csv’ is then created. Finally, the column names are renamed to better describe the columns.

```
bin = [30000, 40000, 50000, 60000, np.inf]
df['binning']= pd.cut(df["Per capita personal income"], bins = bin)
binned_df = df.groupby("binning")['Percentage with high school completion or higher (%)  total'].mean().reset_index()
binned_df_csv = pd.DataFrame(binned_df)
binned_df_csv.rename(columns = {"binning": "Per capita personal income",
                                "Percentage with high school completion or higher (%)  total":
                                "Percentage with high school completion or higher of people age 25 and over (average)"
                                })
```

| | Per capita personal income | Percentage with high school completion or higher of people age 25 and over (average) |
|---|---|---|
| 0 | (30000.0, 40000.0] | 85.902222 |
| 1 | (40000.0, 50000.0] | 88.899412 |
| 2 | (50000.0, 60000.0] | 90.557692 |
| 3 | (60000.0, inf] | 90.381579 |

**Table 1**

Grouped-Aggregation 2 (based on nominal attribute):

First line groups the data frame by the ‘Year’ column, (explain why year is an ordinal) creating a ‘GroupBy’ object, which allows for aggregation operations to be performed on the data grouped by the ‘Year’ column. The ‘mean’ function is then used to calculate the mean of ‘Unemployment rate’ and ‘percentage with high school completion or higher (%) total’. This is stored in a data frame called ‘final_df’ with the calculated mean values for each year. The ‘head()’ function is used to display the first 10 rows of the ‘final_df’ data frame (only 7 needs to be displayed since it’s 2013 - 19). The ‘reset_index()’ function is then used to reset the index of the data frame so that ‘Year’ is no longer the index and instead becomes a regular column. Finally, renaming of the columns is done for better readability.

```
grouped_by_year = df.groupby("Year")
final_df = grouped_by_year[["Unemployment_rate", "Percentage with high school completion or higher (%)  total"]].mean()
final_df.head(10)
final_df = final_df.reset_index()
final_df_csv = pd.DataFrame(final_df)
final_df_csv.rename(columns = {"Unemployment_rate": "Average Unemployment Rate"}, inplace = True)
final_df_csv.rename(columns = {"Percentage with high school completion or higher (%)  total":
                               "Average Percentage with High School Completion or higher of people over the age of 25 (%)"}, inplace = True)
final_df_csv
```

Table 2

| | index | Year | Average Unemployment Rate | Average Percentage with High School Completion or higher of people over the age of 25 (%) |
|---|---|---|---|---|
| 0 | 0 | 2013 | 6.749020 | 88.243137 |
| 1 | 1 | 2014 | 5.743137 | 88.545098 |
| 2 | 2 | 2015 | 5.011765 | 88.805882 |
| 3 | 3 | 2016 | 4.650980 | 89.039216 |
| 4 | 4 | 2017 | 4.158824 | 89.509804 |
| 5 | 5 | 2018 | 3.768627 | 89.823529 |
| 6 | 6 | 2019 | 3.582353 | 90.170588 |

**Chart 1:**

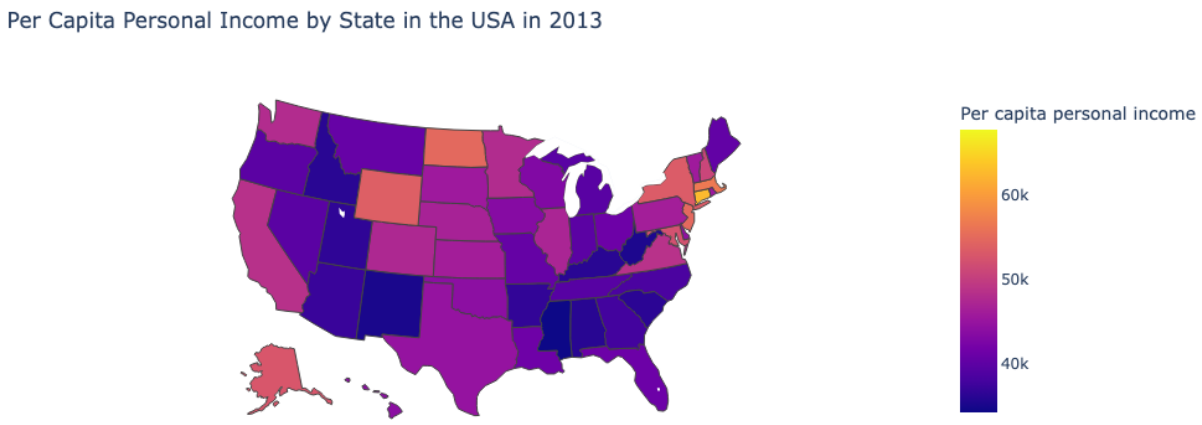
Initially, a new data frame called 'df_2013' is created by filtering the big original data frame. It only selects the rows which have data corresponding to the year 2013 (all the states of which its data is from 2013). Then, creating a choropleth map figure using 'Plotly Express' (px.choropleth). Below are the arguments used:

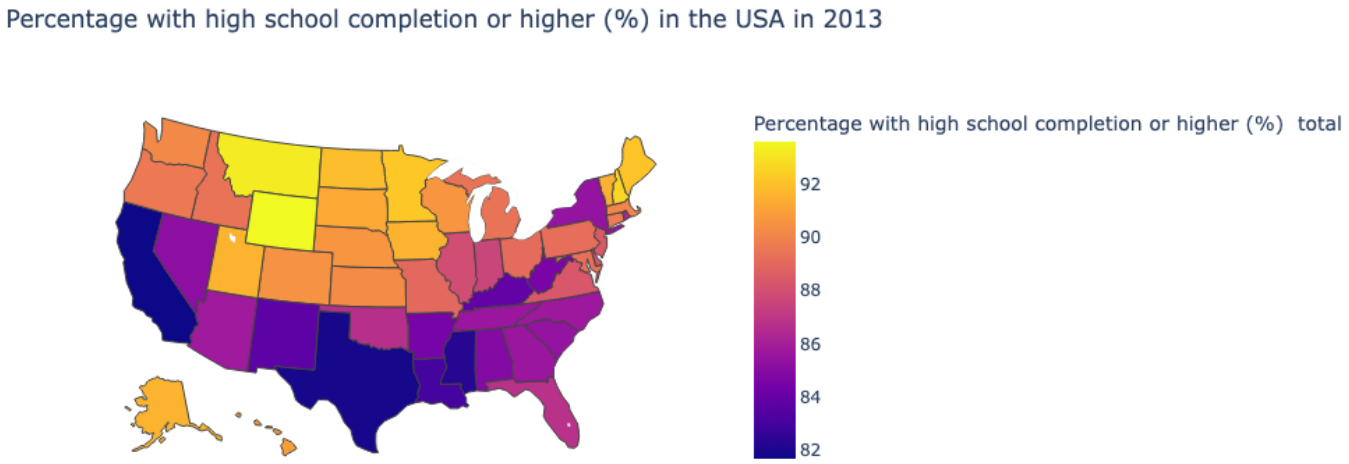
- 'df_2013': Data frame used (previously generated)
- 'locations': The column in the data frame that corresponds to the geographical location of the data, in this it is the postal abbreviations of the different states of the USA.
- 'color': 'Per capita personal income' and 'percentage with high school completion or higher' determines the colour scale for the map (as this is the attribute we are measuring or showing a measurement of)
- 'hover_name': this argument provides additional information when the mouse hovers over a state on the map.
- 'color_continuos_scale': defines the colour scale used for the choropleth map. In this case, the 'Plasma' colour scale is used on both sub-maps so that relationship(s) can be better and easier picked-up and identified.
- 'scope': This argument defines the geographical scope of the map, in this case "usa" is used for the entire United States.
- 'title': the title of the map.

Finally, 'fig.show()' is used to display the map in the python environment or notebook, allowing the user to interact with it.

```
df_2013 = df[df["Year"] == 2013]
fig = px.choropleth(df_2013,
                    locations = "Postal Abbr.",
                    locationmode = "USA-states",
                    color = "Per capita personal income",
                    hover_name = "State",
                    color_continuous_scale = px.colors.sequential.Plasma,
                    scope = "usa",
                    title = "Per Capita Personal Income by State in the USA in 2013")
fig.show()
```

```
fig = px.choropleth(df_2013,
                    locations = "Postal Abbr.",
                    locationmode = "USA-states",
                    color = "Percentage with high school completion or higher (%)  total",
                    hover_name = "State",
                    color_continuous_scale = px.colors.sequential.Plasma,
                    scope = "usa",
                    title = "Percentage with high school completion or higher (%) in the USA in 2013")
fig.show()
```






This graph provides effective visualisation of the higher and lower income/percentage of high school completion based on the colour density. To effectively distinguish between all the states from each other, the map of the USA is extremely effective in showing geographical locations of each state. By setting two maps each identifying each of the attributes, the reader can effectively visualise the connection between the two attributes. The use of the same colour scale furtherly allows for easier visualisation when comparing the two maps. Finally, the interactive nature of the graph allows the user to zoom in and out and more effectively navigate the maps to allow for a more personal experience/interpretation of the maps.

**Chart 2:**

This chart used data from the data frame 'final_df' which is the 2nd grouped aggregation.
The first line creates a Matplotlib figure 'fig' and a set of axes 'axis1'. The 'figsize' parameter then sets the size of the figure. The next line creates a bar plot on 'axis1'. Years from 2013 to 2019 are plotted along the x-axis and the corresponding unemployment rates (average of each year) is plotted on the y-axis. The colour of the bars is selected to be blue with transparency of 0.6, finished with a label.
The second set of axes 'axis2' is then created to share the same x-axis with 'axis1'. This action overlays a line plot on top of the bar plot. Similarly, the percentage with high school completion or higher (on the y-axis) is plotted against the corresponding years (from 2013-2019 on the x-axis). To avoid confusion and for better readability, it uses an orange line with markers "o" and a solid line style "-". Finally, the side is labeled.
To finish off the details on the chart, all 3 sides are labelled with the appropriate title, while a title is added and the 'set_label_coords()' is used for better readability.

```
fig, axis1 = plt.subplots(figsize = (8, 6))
axis1.bar(final_df["Year"], final_df["Unemployment_rate"], color = "Blue", alpha = 0.6, label = "Unployment rate")
# xtick = axis1.set_xticklabels(axis1.get_xticklabels(), rotation = 90, fontsize = 8)
axis2 = axis1.twinx()
axis2.plot(final_df["Year"], final_df["Percentage with high school completion or higher (%)  total"], color = "Orange",
    marker = "o", linestyle = "-", label = "Percentage with high school completion or higher (%) of people over the age of 25")
axis1.set_xlabel("Years")
axis1.set_ylabel("Unployment rate (average across all states)", color = "Blue")
axis2.set_ylabel("Percentage with high school completion or higher (%) \n of people over the age of 25 (average across all states)",
    color = "Orange", rotation = 270)
axis2.yaxis.set_label_coords(1.15, 0.5)
plt.title("Unemployment rate vs Percentage with high school completion or higher (%) of people over the age of 25 from 2013 to 2019 across all states")
plt.show()
```

By plotting the two metrics on the same time scale, this chart effectively displays the relationship between unemployment rate and percentage of high school completion over the years, allowing viewers to quickly compare and understand the relationship between these two variables over time. Though a line graph is usually used with a time series, here a different coloured bar graph overlaps with a line graph so that each attribute can be easier and identified more clearly from each other.

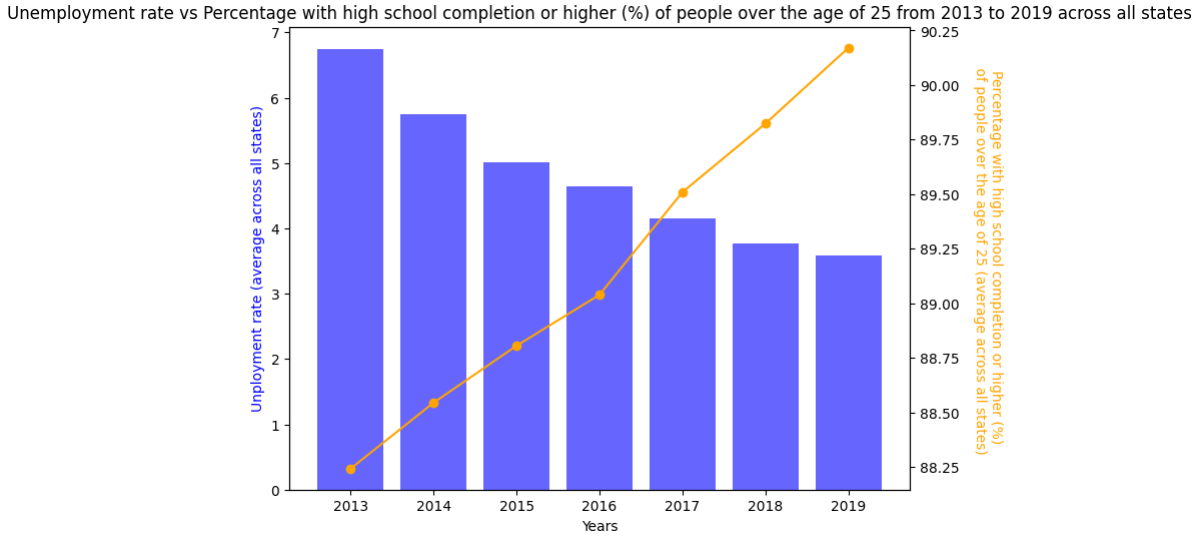



**Chart 1:**

Choropleth map works well for visualising regional data (in this case, per capita personal income by state and percentage of high school completion by state). The choropleth map provides a geographical context for the data and allows viewers to see how two sets of data values differ in the same state, along with the visual encoding of darker colours indicating higher income/percentage and lighter colours indicating lower income/percentage.
The nature of the choropleth map disallows for later-obtained data to be incorporated in the first place. For example, it would not make sense to change “Per capita personal income by state in the USA in 2013” because it is specifically designed so that it only presents the demographics for 2013. However, if more data from other years was obtained and considered, more choropleth maps can be drawn and together constructed as an animation. This can be useful for spotting trends and changes in data across different time periods, while still maintaining the detailed geographical visualisation of the data.

**Chart 2:**

A combination of a bar plot and a line plot is clear and easy to interpret while it is well-suited for comparing two variables that may have different scales, (in this case the “unemployment rate” and “percentage of high school completion”) but share a common time dimension (in this case, from 2013 to 2019). This combination is especially effective when trying to see a trend or a relationship of these two variables over time, as seen in the graph, a clear relationship can be seen between the two attributes, this will be explored in part B of this paper.
Since both the “unemployment rate” and “percentage of high school completion” takes an average across all states of each year, adding more will not necessarily affect the usability of the chart. However if more years (x axis values) were added, potential binning of the time frame (e.g. binned by decades) is necessary to ensure readability and prevent overcrowding of the data.

Individual Section
530508591 - Emily

Dataset description:
Chosen dataset is the merged dataset followed from Stage 1, where a subset of it was used in this particular analysis.

**'State'** (String/Object) : Name of U.S state
**'Year'** (Integer) : Year of data (ranging from 2013-2019)
**'Population'** (Integer) : Number of people
**'Disposable PI*'** (Float) : Income available to a person for spending or saving. Equal to personal income without personal current taxes.
'**Real GDP**'** (Float) : Inflation-adjusted measure of each state's GDP based on national prices for goods and services in the state
**'Real PI'** (Float) : Personal income divided by the RPPs and national PCE price index
**'Total [crime]'** (Integer) : Number of the specific crime type recorded

*Personal income is abbreviated as P.I
**Gross Domestic Product is abbreviated as GDP

For convenience, a new dataframe containing a subset of the merged dataset (i.e. the columns of interest) was created; using `'.iloc()'` to slice across an index range and followed by `'pd.concat'` to merge the specific columns. Code is displayed below, with the filtered dataset named `'new_df'.`

```
#select a subset of the dataset with the relevant data
state_year_columns = df.iloc[:, [1, 2]]
income_columns = df[['Population','Disposable personal income','Real GDP','Real personal income']]
crime_columns = df.iloc[:, 32:41]
new_df = pd.concat([state_year_columns, income_columns, crime_columns], axis=1)
new_df
```

## Grouped aggregate summary

a) Grouped aggregate based on nominal attribute

The filtered dataset is first organised by grouping it based on the nominal attribute 'State' using the `'groupby()'` function. This procedure returns a condensed representation of the data for each unique state. Since our initial dataset spans several years (2013-2019) for each state, we take it a step further by computing aggregate statistics using the `'agg()'` function. This function allows us to specify which columns we want to include in our analysis and the specific aggregation methods we want to apply. We chose to use functions like `'mean'`, `'min'`, and `'max'` on columns such as 'Disposable Personal income', 'Real GDP', 'Real personal income' and the crime-related data. These aggregation functions provide a comprehensive overview of the dataset, summarizing key aspects of the attributes for each US state. They enable us to gain valuable insights into how these factors, both individually and collectively, influence trends in crime data by capturing central tendencies and variations within the data. Final output illustrated in Table 1.

```
# Grouping the dataset by the 'State' column
state_grouped = new_df.groupby('State')

# Calculate aggregate summaries for the grouped data
aggregate_income_crime = state_grouped[new_df.columns[3:]].agg(['mean','min','max'])

# Reset the index to have 'State' as a regular column, not an index
aggregate_income_crime = aggregate_income_crime.reset_index()

# Center-align columns using HTML-style formatting
aggregate_income_crime = aggregate_income_crime.style.set_properties(**{'text-align': 'centre'})

aggregate_income_crime
```

Table 1: Small section of the aggregate table based on U.S states, with data for mean, min, and max.

| | State | Disposable personal income | | | Real GDP | | | | Real personal income | | Total crimes | | | Total burglary | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | mean | min | max | mean | min | max | mean | min | max | mean | min | max | mean | min | max |
| 0 | Alabama | 176250.642857 | 159063.100000 | 196622.700000 | 195379.328571 | 189886.300000 | 203432.700000 | 206157.728571 | 190336.300000 | 223626.500000 | 145312.571429 | 131133 | 161993 | 33980.142857 | 26079 | 42429 |
| 1 | Alaska | 38802.242857 | 35601.400000 | 41593.300000 | 54224.200000 | 53327.000000 | 55354.300000 | 38758.871429 | 37516.900000 | 39945.800000 | 22723.285714 | 20334 | 26204 | 3620.428571 | 2916 | 4171 |
| 2 | Arizona | 258123.914286 | 219881.600000 | 305173.100000 | 295444.557143 | 273481.900000 | 325395.300000 | 278451.057143 | 243833.100000 | 316996.300000 | 204021.285714 | 177638 | 225243 | 37949.428571 | 28699 | 48533 |
| 3 | Arkansas | 110254.085714 | 98053.700000 | 121026.400000 | 113499.657143 | 110752.400000 | 117126.200000 | 130333.157143 | 118709.100000 | 138377.800000 | 95238.857143 | 86250 | 106613 | 22976.571429 | 18095 | 30485 |
| 4 | California | 1897720.642857 | 1603552.400000 | 2198931.700000 | 2447446.028571 | 2179229.000000 | 2729225.800000 | 1918062.771429 | 1694852.200000 | 2108670.200000 | 977561.285714 | 921114 | 1024914 | 187759.000000 | 152555 | 232058 |

a) Grouped aggregate based on binned quantitative attribute

We selected the year 2019 from the six-year dataset as our reference year, aiming for a more current representation of the data. To categorize the data based on disposable personal income (the chosen quantitative attribute), we utilized the ‘pd.qcut’ function, which discretizes the data into equally sized bins determined by sample quantiles (in this case, a total of five bins). We set ‘labels=False’ to obtain integer bin labels instead of categories, and ‘duplicates=drop’ to eliminate duplicate bins. The function returned a series of bin labels for each row in the new dataframe. To make these bin labels more informative, we printed the bin edges of the generated quantiles with their corresponding bin number as seen in the code to the right .

```
# Filter data for the year 2019
df_2019 = df[df['Year'] == 2019]

# Specify the number of bins
num_bins = 5

# Use pd.qcut to create quantile-based bins
bins, bin_edges = pd.qcut(df_2019['Disposable personal income'], q=num_bins, retbins=True, \
                          labels=False, duplicates='drop')

# Get the bin labels based on the bin edges
bin_labels = [f'Bin {i}' for i in range(1, num_bins + 1)]

# Print the bin edges and labels
print("Bin Edges:")
print(bin_edges)
print("\nBin Labels:")
print(bin_labels)
```

```
Bin Edges:
[  31911.    70507.   144102.   275745.2  442540.1 2198931.7]

Bin Labels:
['Bin 1', 'Bin 2', 'Bin 3', 'Bin 4', 'Bin 5']
```

Consequently, we assigned meaningful value ranges as the bin names and introduced a new column named ‘Disposable Personal Income Bins’, which associates bin labels with the respective rows. Once again, we employed the ‘groupby()’ function, this time to group the 2019 data according to income bins. We proceeded to calculate aggregate summaries, including mean and median values, specifically for the crime data. This approach enables us to effectively compare variations in the crime data across specific income ranges, deducing any particular correlation between the attributes when one is increasing/decreasing. Custom bin labels and code for grouped summaries are shown above. Table output is presented in Table 2.

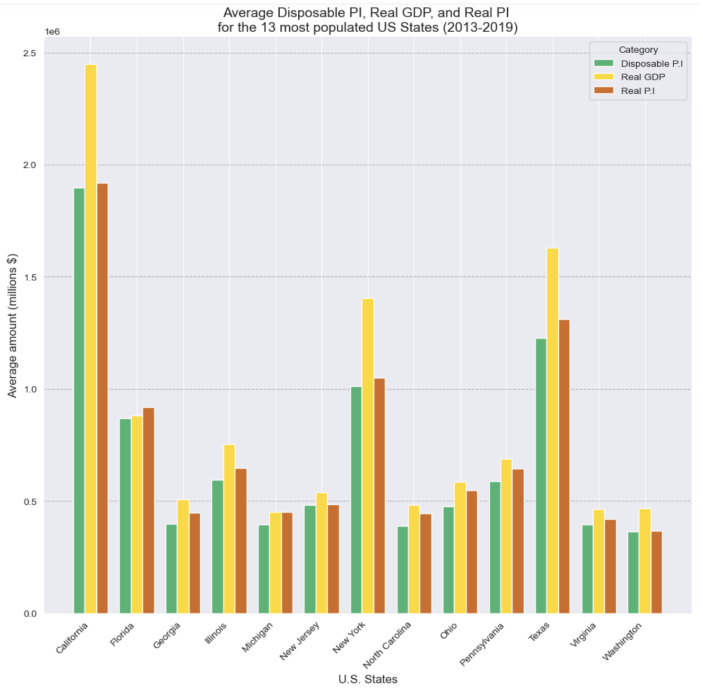
```
income_bins = pd.qcut(df_2019['Disposable personal income'], q=5, labels=False, duplicates='drop')

#modify bin labels to correspond approx. values from returned by q.cut
bin_labels = ['< $70K', '\$70K-$144K', '\$144K-$275K', '\$275K-$442K', '> $442K']

# Create a new column 'Disposable Personal Income Bins' based on the bins
df_2019['Disposable Personal Income Bins'] = pd.Categorical.from_codes(income_bins, categories=bin_labels)

# Group the data based on bins
grouped_data = df_2019.groupby('Disposable Personal Income Bins')

# Calculate aggregate summaries for crime numbers (e.g., mean, sum) within each income bin
crime_summaries = grouped_data[['Total crimes','Total burglary','Total motor','Total violent crimes',\
                                'Total violent assault','Total violent murder','Total violent rape',\
                                  'Total violent robbery']].agg(['mean', 'median'])
crime_summaries
```

Table 2: Small section of the aggregate table binned in terms of Disposable Personal income, with data computed for mean and median.

| | Total crimes | | Total burglary | | Total motor | | Total violent crimes | | Total violent assault | | Total violent murder | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | mean | median | mean | median | mean | median | mean | median | mean | median | mean | median |
| Disposable Personal Income Bins | | | | | | | | | | | | |
| < $70K | 18870.727273 | 16743.0 | 2704.363636 | 2608.0 | 1635.727273 | 1756.0 | 3633.818182 | 3530.0 | 2512.272727 | 2676.0 | 45.272727 | 25.0 |
| $70K-$144K | 53181.100000 | 59984.0 | 9765.900000 | 9427.5 | 5398.600000 | 5500.5 | 8723.300000 | 7912.5 | 6262.500000 | 5013.5 | 115.300000 | 66.0 |
| $144K-$275K | 111204.600000 | 113878.5 | 19823.400000 | 20794.5 | 12270.800000 | 11918.5 | 18489.400000 | 17078.0 | 13017.500000 | 11913.0 | 295.900000 | 243.5 |
| $275K-$442K | 145986.100000 | 144701.0 | 22256.900000 | 20929.5 | 15419.000000 | 15733.5 | 26787.400000 | 23772.0 | 18011.500000 | 16203.5 | 344.900000 | 371.0 |
| > $442K | 361438.100000 | 249742.5 | 56948.600000 | 41700.0 | 37300.800000 | 18723.5 | 66543.700000 | 45394.5 | 42063.200000 | 29852.0 | 836.600000 | 661.5 |

**Chart 1**

Before creating the visualizations, we undertook a series of data pre-processing steps. Initially, we computed the average population values for each state across the six years in the ‘new_df’ dataset. Subsequently, we divided these populations into three distinct groups based on quantile values, where the top 25% were classified as high, the bottom 25% as low, and the rest as middle. Within these population bins, we extracted the corresponding state names, which allowed us to generate charts focusing on specific states within a given category. This approach enabled us to discern trends within states that share similar population characteristics and isolate any population-related factors when later making comparisons with other attributes.



This first chart explores income data and GDP for the top 25% (13) most populous states. It employs a multi-series bar chart, with columns colour-coded to represent different categories. To create this chart, we first filtered the 'new_df' dataset to keep only the states of interest using the 'isin()' function. The mean values for the columns 'Disposable personal income,' 'real GDP,' and 'real personal income' were then calculated after grouping data by state. Utilising a bar plot, we can effectively showcase categorical data, specifically US states, with each state being represented by a set of three differently coloured bars corresponding to the numerical values of personal income and GDP. Bar plots also facilitate visual comparison of the data, allowing viewers to quickly identify disparities in personal income and GDP among states through a straightforward comparison of bar heights.

The bar plot itself was generated using Matplotlib, which setups the figure and axes for the plot. The x-axis positions of the bars were determined by an `'index'` variable, which calculated the total number of positions required based on the number of states. Additionally, we defined parameters specifying the width of each bar (`'bar_width'`) and the spacing between bars (`'bar_spacing'`). We stored the calculated average values for Disposable Personal Income (DPI), Real GDP, and Real Personal Income (Real PI) in separate variables and proceeded to create three sets of bars on the same plot using the `'bar()'` function. Each set of bars represented one of the three variables for the selected states. Furthermore, we set customized x and y-axis labels, rotating the x-axis labels to improve readability. To aid interpretation, a legend was incorporated to elucidate the color code corresponding to the various categories.

```
# Filter the dataset to include only specific U.S. states
states_of_interest = ['California', 'Florida', 'Georgia', 'Illinois', 'Michigan', 'New Jersey', 'New York', \
                      'North Carolina', 'Ohio', 'Pennsylvania','Texas', 'Virginia', 'Washington']

filtered_df = new_df[new_df['State'].isin(states_of_interest)]

# Group the data by state and calculate the mean DPI and GDP
grouped_data = filtered_df.groupby('State')[['Disposable personal income', 'Real GDP','Real personal income']].mean()

# Apply a Seaborn style
sns.set_style("darkgrid")

# Create a bar plot with grouped bars
fig, ax = plt.subplots(figsize=(10, 10))

index = np.arange(len(states_of_interest))  # X-axis index positions for the bars
bar_width = 0.25  # Width of each bar
bar_spacing = 0.05  # Additional spacing between bars

dpi_data = grouped_data['Disposable personal income']
gdp_data = grouped_data['Real GDP']
real_pi_data = grouped_data['Real personal income']

bar1 = ax.bar(index, dpi_data, bar_width, label='Disposable P.I', color='mediumseagreen')
bar2 = ax.bar([i + bar_width for i in index], gdp_data, bar_width, label='Real GDP', color='gold')
bar3 = ax.bar([i + 2 * bar_width for i in index], real_pi_data, bar_width, label='Real P.I', color='chocolate')

# Set x-axis labels and title
ax.set_xlabel('U.S. States', fontsize=12)
ax.set_ylabel('Average amount (millions $)', fontsize=12)
# Set the title text and configure it for a two-line title
title_text = 'Average Disposable PI, Real GDP, and Real PI \nfor the 13 most populated US States (2013-2019)'
ax.set_title(title_text, fontsize=14, loc='center')
```

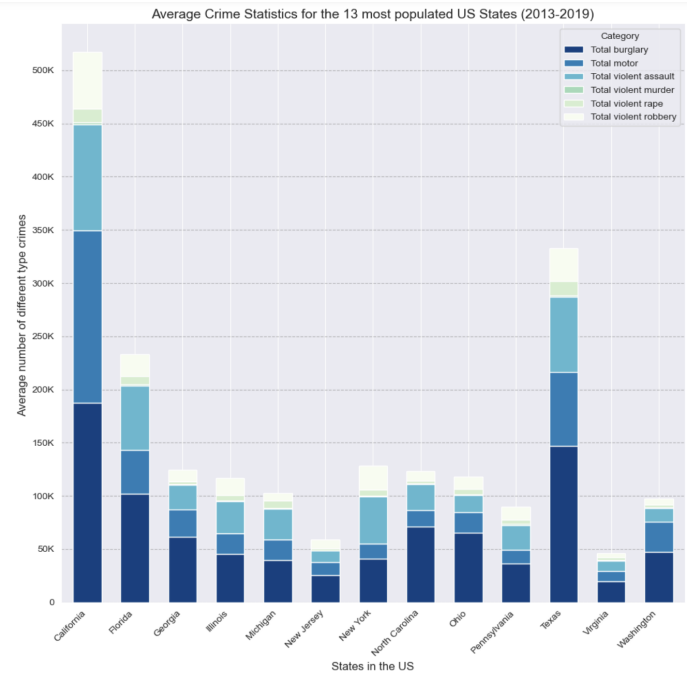
```
# Set x-axis tick positions and labels
ax.set_xticks([i + bar_width for i in index])
# Rotate x-axis labels
ax.set_xticklabels(states_of_interest, rotation=45, ha="right", fontsize=10)

# Customize the legend
ax.legend(title='Category', fontsize=10)
ax.get_legend().set_title('Category')

# Add grid lines
ax.grid(color='gray',axis='y', linestyle='--', alpha=0.5)

# Remove the top and right spines
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# Customize the y-axis ticks and labels
plt.yticks(fontsize=10)  # Increase font size for y-axis labels

# Adjust the space between x-axis labels
plt.subplots_adjust(bottom=0.2)

# Show the plot
plt.tight_layout()
plt.show()
```

Bar plots are most impactful when viewers can easily make meaningful comparisons among different categories. However, overwhelming the plot with excessive data can decrease readability and effective communication of a particular message. In our case, we chose to incorporate data related to overall personal income and overall GDP, ensuring that the information presented doesn't create confusion or misinterpretations regarding scales of measurement. Expanding the data would complicate the interpretation of inter-category relationships and necessitate more intricate axis labelling.

Other Chart design decisions:

- The plot's style was defined using Seaborn's `'set_style()'` function. We opted for a 'darkgrid' design to enhance the contrast between our chosen colour palette and the background, thereby enhancing data legibility.
- Instead of keeping the values of the y-axis ticks as 7 digit numbers, we applied appropriate scaling with units in millions to improve the clarity and readability of the large numbers, also making data interpretation much more straightforward. Applying such scaling is valid in this case as our focus is on the trends and patterns in the data rather than their specific magnitudes.
- The green, yellow and brown colours chosen for the different attributes provide good contrast that enhances readability for viewers. The colour combination also makes the data accessible for individuals with colour vision impairments.

## Chart 2




Similarly to Chart 1, we collected data for the same set of 13 states, focusing this time on different crime metrics. The filtered dataset was again further grouped by state, where within each state, we calculated the mean values for the selected crime columns that encompassed total counts for different crime types such as burglary, assault, murder etc.

Unlike the bar plot used for Chart 1 which displayed data individually, we opted for a stacked bar chart. This choice provided a more effective visualisation of the crime data by illustrating the individual components that contribute to the total crime numbers in each state. This approach is valuable for further analysis because it allows us to visually assess how the composition of crime types varies from state to state. Consequently, it provides insights into which specific types of crimes are more or less prevalent, enabling us to make informed observations about data relationships.

We used the `'plot()'` function to easily customize the appearance of the bars by providing parameters to the arguments. A legend was included in the upper right corner of the chart to explain the colour scheme used to represent different crime types.
However, one issue arose during chart generation: the default y-axis had very large increments, making it challenging to interpret data values accurately. To address this, we manually specified custom tick labels for the y-axis, ensuring that the increments were evenly spaced and enhancing the readability of the chart.

```
# Group the data by state and calculate the mean total crimes
grouped_data = filtered_df.groupby('State')[['Total burglary', 'Total motor', \
                                             'Total violent assault', 'Total violent murder', \
                                             'Total violent rape','Total violent robbery']].mean()

# Apply a Seaborn style
sns.set_style("darkgrid")

# Stacked bar chart for crime statistics
ax = grouped_data.plot(kind='bar', stacked=True, figsize=(10, 10),colormap='GnBu_r',width=0.6,edgecolor=None)

# Set labels and title
ax.set_xlabel('States in the US', fontsize=12)
ax.set_ylabel('Average number of different type crimes ', fontsize=12)
ax.set_title('Average Crime Statistics for the 13 most populated US States (2013-2019)',fontsize=14)

# Customize colors
ax.set_prop_cycle('color', sns.color_palette("Paired"))

# Add a legend with a title
ax.legend(title='Category',loc='upper right',fontsize=10)

# Set x-axis labels to the states of interest
plt.xticks(range(len(states_of_interest)), states_of_interest, rotation=45, ha="right", fontsize=10)


# Change Y-Axis Increments
# Specify custom ticks and labels here
custom_y_ticks = [0, 50000, 100000, 150000, 200000, 250000, 300000, 350000, 400000, 450000, 500000]
custom_y_labels = ['0', '50K', '100K', '150K', '200K','250K','300K','350K','400K','450K','500K']
plt.yticks(custom_y_ticks, custom_y_labels)

# Add grid lines
ax.grid(color='gray',axis='y', linestyle='--', alpha=0.5)

# Remove the top and right spines
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

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

Adding significantly more data into this stacked bar plot wouldn't necessarily reduce its effectiveness, unlike the simple bar plot depicted in Chart 1. Instead, it would just provide a further breakdown of total crimes into smaller subcategories. Given that our focus revolves around understanding the trends in crime numbers for each state and external factors might influence them, providing excessive detail regarding the individual components contributing to these numbers wouldn't be necessary. Additionally, including more data might also require a larger range of colours, potentially complicating visual comprehension if the colours are less distinguishable from one another within a sequential palette.

Other chart designs:

- A sequential colour palette was chosen to make it easier for viewers to understand the relationship of the attributes and make it clear that they belong to the same overarching category (in this case, total crimes).
- The y-axis labels were shortened using 'K' instead of 000's for thousands to reduce complexity in data comprehension and crowding of values.

Part A individual section
SID: 530023821 Yining Wang

The merged dataset is chosen as it contains the most information and therefore is the best for demonstrating relationships and correlations between attributes. We focus on the socioeconomic factors and crime rates of all type in this section. The columns chosen with the dataset include: rates of all crimes, year, unemployment rate, personal disposable income, % with high school completion or higher.

## 1. Aggregate summary (Using ***df.groupby()*** and ***df.agg()*** )

### (a) Group by Nominal Data

Firstly, we group the data by State to obtain information about the merged dataset. Within the grouped range, we compute the average crime rates, education level and unemployment rate between 2013 and 2019. Such a grouping is meaningful as we aim to investigate factors that affect crimes, and information on each state can be considered an individual data point. The code and output are shown below.

```
#Group data by State
state_stat=df.groupby('State')
'''
Calculate aggregate summary: 2013-2019 the average unemployment rate,% of education completion,
income and rates of all crime types in different states in the US
'''
state_crime_summary=state_stat[df.columns[4:50]].agg(['min','max','mean','median'])
state_crime_summary.head()
```

| | Unemployment_rate | | | | Disposable personal income | | | | Gross domestic product (GDP) | | ... | Percentage with high school completion or higher (%) Female | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | min | max | mean | median | min | max | mean | median | min | max | ... | mean | median |
| State | | | | | | | | | | | | | |
| Alabama | 3.2 | 7.3 | 5.371429 | 5.9 | 159063.1 | 196622.7 | 176250.642857 | 173653.1 | 194786.9 | 231561.9 | ... | 86.642857 | 86.3 |
| Alaska | 5.6 | 7.0 | 6.385714 | 6.5 | 35601.4 | 41593.3 | 38802.242857 | 38714.9 | 50727.7 | 57247.7 | ... | 92.400000 | 92.3 |
| Arizona | 4.8 | 7.8 | 5.828571 | 5.5 | 219881.6 | 305173.1 | 258123.914286 | 253015.4 | 278591.6 | 372393.5 | ... | 87.171429 | 87.4 |

Table 1

### (a) Group by Quantitative Data

We take 2019 as the year of interest and bin data based on the unemployment rate. ***pd.cut*** gives us the 7 distinct levels of unemployment rates. We then create a new column 'Unemployment_level' that contains the binned labels based on the unemployment rate. Finally, we group the data by 'Unemployment_level' and calculate summary statistics (minimum, mean, maximum, and median) for the columns related to crime rates (columns 24 to 33) and display the results.

```
'''
Bin data based on unemployment rate(Quantitative data)
We label the bins according to 5 levels of unemployment rate:
extremely low, very low, low, medium, high, very high, extremely high
'''
df1=df[df['Year']==2019]
df1['Unemployment_level']=pd.cut(df1['Unemployment_rate'],bins=7,
labels=['extremely low','very low', 'low', 'medium', 'high', 'very high','extremely high'])
unemployment_rate_binned=df1.groupby(['Unemployment_level'])

'''Calculate aggregate summary:
2013-2019
the average unemployment rate and rates of all crime types
in different states in the US
'''
binned_crime=unemployment_rate_binned[df1.columns[24:33]].agg(['min','mean','max','median'])
```

| | Rates of all crimes | | | |
|---|---|---|---|---|
| | min | mean | max | median |
| Unemployment_level | | | | |
| extremely low | 1209.2 | 1924.220000 | 2841.2 | 1977.00 |
| very low | 1179.8 | 1920.790000 | 2940.3 | 1752.35 |
| low | 1335.7 | 2133.194118 | 2858.0 | 2193.20 |
| medium | 1373.3 | 2001.611111 | 2730.6 | 1897.40 |
| high | 1403.4 | 2325.750000 | 3162.0 | 2368.80 |
| very high | 1583.4 | 2378.866667 | 3112.7 | 2440.50 |
| extremely high | 2375.8 | 3217.900000 | 4367.1 | 2910.80 |

Table 2: Aggregate summary of data based on binned 'Unemployment' numbers

## 2. Chart production

### (a) Simple data visualisation

**State (qualitative) vs. Average unemployment rate (quantitative variable)**

```
state_crime_avg=state_crime_avg.reset_index()

plt.figure(figsize=(10, 6))
sns.barplot(x="State", y="Unemployment_rate",data=state_crime_avg)
plt.title("Average unemployment rate by state ")
plt.xlabel("State")
plt.ylabel("Average unemployment rate")
ax=plt.gca()
plt.xticks(rotation=90, ha='right')
plt.show()
```

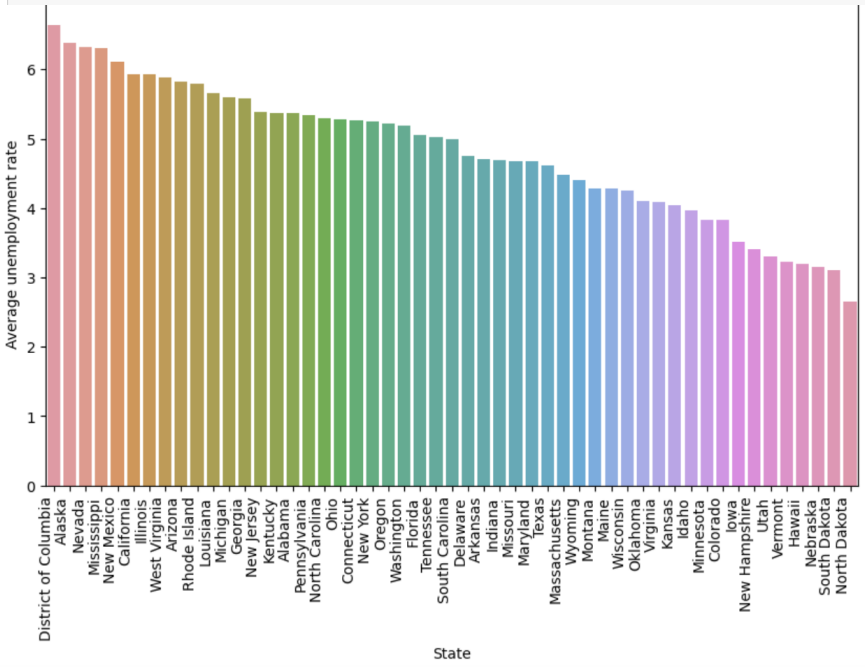



**Figure 1**

- To have a brief first look at the unemployment information in different States, we first ***reset the index*** and then a bar plot with the x-axis being states and the y-axis being the average unemployment rate is used. ***Matplotlib.pyplot***(as plt) and ***Seaborn***(as ***sns.barplot***) are the two modules we import to produce the chart. Using ***plt.xlabel*** and ***plt.ylabel,*** we label the axis according to its corresponding value.The x-axis labels are rotated for better readability (***plt.xticks(rotation=90, ha='right'***), which also improves the interpretability of the plot.
- Effectiveness: The colored bar plot makes it easy to compare the unemployment rates of different states at a glance. From the chart, we can easily see the hierarchy of the unemployment data in different states. D.C. Alaska and Nevada have the highest rates, all above 6, while South and North Dakota witness the lowest statistics (less than 3.5).
- Limitation: As there are a fixed number of state in the US, the obtainment of more data would not affect the effective of the chart. However, ths plot doesn't show any correlation or causation between unemployment rates and other factors. To understand the factors influencing unemployment rates, we would need to perform statistical analysis, such as regression analysis or correlation studies. Meanwhile, it doesn't provide insights into trends or changes over time.

### (b) Crime rates vs. unemployment rate, median income, education attainment (2013-2019 respectively)

```
#Group data by Year and State
year_stat=df.groupby(['Year','State'])
'''
Calculate aggregate summary: Unemployment rates,education attainment,income from 2013 to 2019 respectively
and rates of all crime types in different states in the US
'''
year_crime=year_stat[df.columns[7:50]].median()
year_crime.head()
```

```
year_crime=year_crime.reset_index()
sns.lmplot(x='Unemployment_rate',y='Rates of all crimes',hue='Year',data=year_crime,
fit_reg=True).set(title='Rates of all crimes in 2013-2019 repectively vs. Unemployment rates in States ')
#3 attributes: Unemployment rate, year, Rates of all crimes
```

```
sns.lmplot(x='Disposable personal income',y='Rates of all crimes',hue='Year',data=year_crime,
           fit_reg=True).set(title='Rates of all crimes in 2013-2019 repectively vs. Disposable income ')
```

```
sns.lmplot(x='Percentage with high school completion or higher (%)  total',
          y='Rates of all crimes',hue='Year',data=year_crime,
          fit_reg=True).set(title=' Rates of all crimes in 2013-2019 repectively vs. % of high school completion ')
```

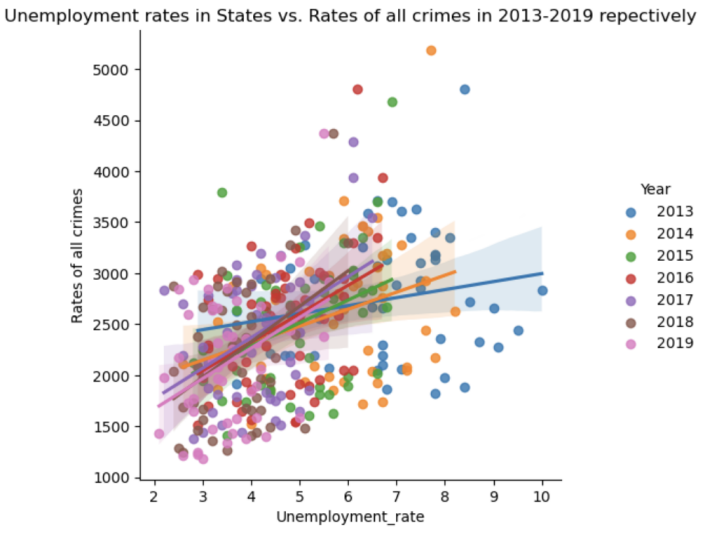
Firstly, we group data by both "Year" and "State" using ***df.groupby()*** to calculate aggregate summary statistics for various variables, including unemployment rates, education attainment, income, and crime rates over the years 2013 to 2019. The code is using the ***median() function*** to calculate the median values for these variables within each group. Then scatter plots with hue and regression fit (***fit_reg=True***) were chosen to demonstrate the three-way relationship regarding socioeconomic factors (unemployment rate, disposable income and percentage of high school completion), crime rates and year. This is a powerful visualization technique that provides insights into how these variables are correlated over time.

**Encoding Between Data Attributes and Visual Attributes:**

The charts were produced using ***seaborn*** (***sns.lmplot***) and scatter data from the different years is distinguished by distinct colours (***hue='year'***). For each year, the corresponding regression line is reflected by the same palette, illustrating the correlation between variables on both axes.The ***x-axis*** consistently represents a key socioeconomic variable, such as unemployment rate or the percentage of high school completion, while the ***y-axis*** consistently represents crime rates, allowing viewers to understand the relationships between these variables easily.

**Scale and labels:**

The scales used on the axes are well-constructed and appropriate, while unemployment rate and % with high school completion use percentage, rates of all crimes are the number of reported offenses per 100,000 population, and personal disposable income is actual number. For each graph, labels and titles providing information regarding corresponding axes using ***set(title=' ')***.




- **Correlation between unemployment rate and crime rates**

it is clear that the average unemployment rate is decreasing yearly as the slope of the regression fit is gradually becoming steeper (**Figure 2**). Meanwhile, the regression line slants upward from left to right, demonstrating that there is a positive correlation between unemployment rates and crime rates–that is, the larger the unemployed population, the greater the risk of crime incidence.

**Figure 2: Average unemployment rate vs. Crime rates vs. Year**

- **Correlation between disposable personal income and crime rates**

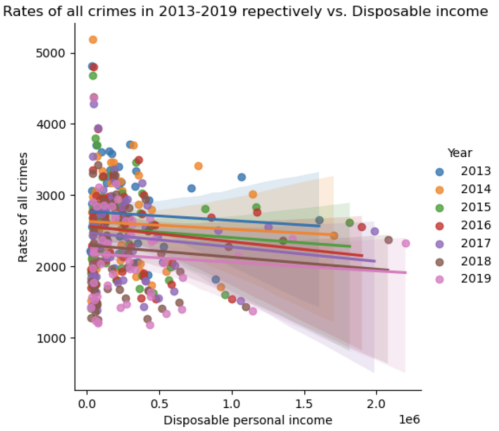



Figure 3

In **Figure 3**, While the regression line does display a negative slope, suggesting a potential inverse correlation between income and crime rates, it's important to note that the magnitude of this slope is quite small and nearly horizontal. As a result, it's challenging to definitively conclude whether a significant relationship exists between income and crime rates based solely on this visualization. For a more robust analysis, further statistical tests and the consideration of additional variables may be necessary to provide a more comprehensive understanding of this relationship.

- **Correlation between percentage of high school completion and crime rates**

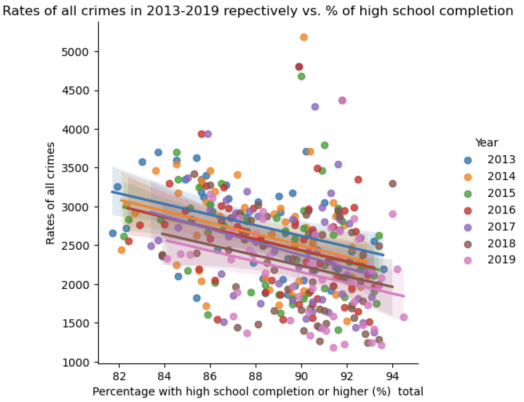



Figure 4

**Figure 4** provides compelling evidence of a substantial and distinct negative correlation between the percentage of high school completion and crime rates. The regression line's pronounced negative slope underscores the inverse relationship between these two variables—higher high school completion percentages are associated with lower crime rates.

**Effectivness of communication**: The presented 3 graphs effectively illustrate the relationships between key socioeconomic factors—unemployment rate, education attainment, disposable income—and crime rates over the years. They provide a visual representation of potential correlations and trends. The inclusion of regression lines and distinct colors for different years enhances clarity and facilitates the understanding of these complex relationships. Scatter plots with regression fits effectively highlight trends, clusters, and outliers in the data.

**Limitations**: For all three graphs, it is important to know that there are a large number of outliners which can skew interpretations and introduce uncertainty into the observed trends. As we only have 51 data points(each representing a state in the US), the chart may not accurately represent the true underlying relationship between the variables. Small fluctuations or outliers in the data can have a disproportionate impact on the regression line. Therefore, while and further analysis is required, more data obtained would contribute greatly to quantify the correlation.

# Individual sections

## 510644297 - Danica

### Grouped Aggregation 1 based on quantitative attribute

We selected three attributes, including years, real GDP, and rates of all crimes from the whole dataset, in order to find the relationship between real GDP and rates of all crimes from 2013 to 2019. We utilized the ‘pd.filter’ function, which extracts the data. We used the ‘groupby()’ function to group data. We proceeded to calculate summaries, including mean, standard values, minimum, maximum, and others. The picture below is the table output.

```
df = pd.read_excel('/Users/mac/Downloads/Merged Dataset.xlsx')
print(df.dtypes)

df = pd.DataFrame(df)
print(df.columns)

# Filter out labels of interest, group the data on state values and make a statistic description
grouped_sn = df.filter(['Real GDP', 'Year', 'Rates of all crimes', 'Personal consumption expenditures', 'Per capita personal income'])
grouped_sn = grouped_sn.groupby(['Year']).describe()
grouped_sn = grouped_sn.reset_index()
print(grouped_sn)
```

Table 1

| | Year | Real GDP | | | | | | | ... | Per capita personal income | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | count | mean | std | min | 25% | 50% | 75% | ... | count | mean | std | min | 25% | 50% | 75% | max |
| 0 | 2013 | 51.0 | 321398.521569 | 398186.672302 | 28681.5 | 79959.90 | 181505.9 | 419805.75 | ... | 51.0 | 44426.666667 | 7312.678584 | 34259.0 | 39510.0 | 43931.0 | 47847.5 | 67774.0 |
| 1 | 2014 | 51.0 | 328893.880392 | 410092.740119 | 28912.2 | 81251.80 | 189360.0 | 429390.40 | ... | 51.0 | 46428.745098 | 7738.805817 | 34896.0 | 40970.5 | 46015.0 | 50557.5 | 71469.0 |
| 2 | 2015 | 51.0 | 338024.982353 | 426395.958413 | 29119.0 | 82785.40 | 192019.9 | 442891.65 | ... | 51.0 | 47995.941176 | 7965.129723 | 35533.0 | 42601.0 | 46548.0 | 52288.5 | 75623.0 |
| 3 | 2016 | 51.0 | 343727.378431 | 436359.287252 | 29408.1 | 83227.65 | 198005.7 | 455294.50 | ... | 51.0 | 48610.450980 | 8178.553156 | 36021.0 | 42984.5 | 46586.0 | 52493.0 | 78186.0 |
| 4 | 2017 | 51.0 | 351386.378431 | 451099.166660 | 29490.6 | 83955.15 | 202512.4 | 462488.00 | ... | 51.0 | 50238.843137 | 8389.012763 | 36902.0 | 44735.0 | 48758.0 | 54572.5 | 79984.0 |
| 5 | 2018 | 51.0 | 361451.719608 | 468418.451377 | 29602.2 | 85389.45 | 207101.2 | 472237.30 | ... | 51.0 | 52339.941176 | 8727.974366 | 37900.0 | 46635.0 | 50908.0 | 56579.0 | 82708.0 |
| 6 | 2019 | 51.0 | 369775.507843 | 482654.075338 | 29940.7 | 87055.40 | 213237.8 | 478153.05 | ... | 51.0 | 54724.823529 | 9023.148286 | 39445.0 | 48745.0 | 52893.0 | 58808.0 | 84671.0 |

### Grouped Aggregation 2 based on nominal attribute

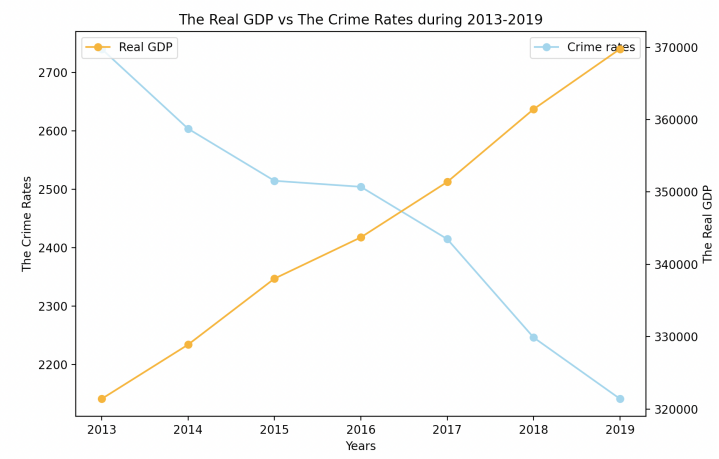
We grouped two attributes, including personal consumption expenditures and per capita personal income, by State to compare personal consumption expenditures and per capita personal incomes.

```
# Filter out labels of interest, group the data on state values and make a statistic description
grouped_sp = df.filter(['State', 'Personal consumption expenditures', 'Per capita personal income'])
grouped_sp = grouped_sp.groupby(['State']).describe()
grouped_sp = grouped_sp.reset_index()
print(grouped_sp)

states = grouped_sp['State']
consumption_expenditures = grouped_sp['Personal consumption expenditures', 'mean']
per_capita_income = grouped_sp['Per capita personal income', 'mean']
```

Table 2

| | State | Personal consumption expenditures | | | | | ... | Per capita personal income | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | count | mean | std | min | 25% | ... | std | min | 25% | 50% | 75% | max |
| 0 | Alabama | 7.0 | 1.591581e+05 | 12037.911993 | 144027.9 | 150722.65 | ... | 2524.372190 | 36014.0 | 37793.0 | 39014.0 | 40881.0 | 43288.0 |
| 1 | Alaska | 7.0 | 3.455147e+04 | 2133.413513 | 31709.4 | 33158.70 | ... | 2732.555389 | 52812.0 | 56072.5 | 57189.0 | 58596.5 | 61316.0 |
| 2 | Arizona | 7.0 | 2.486482e+05 | 25748.957307 | 215979.0 | 230529.35 | ... | 3848.695697 | 37139.0 | 39545.0 | 41473.0 | 44481.5 | 48124.0 |
| 3 | Arkansas | 7.0 | 9.473760e+04 | 6931.271153 | 85290.2 | 89868.85 | ... | 2651.319887 | 36677.0 | 39358.5 | 40873.0 | 42637.0 | 44324.0 |
| 4 | California | 7.0 | 1.668061e+06 | 171996.778654 | 1443067.6 | 1546342.75 | ... | 5721.308529 | 48502.0 | 52906.0 | 56560.0 | 60156.0 | 64919.0 |
| 5 | Colorado | 7.0 | 2.343943e+05 | 27131.096759 | 197806.0 | 215181.80 | ... | 4940.124010 | 47404.0 | 51568.0 | 52390.0 | 56852.0 | 62124.0 |
| 6 | Connecticut | 7.0 | 1.709788e+05 | 10059.793606 | 157135.5 | 164265.35 | ... | 4371.130191 | 62647.0 | 66409.5 | 68680.0 | 71468.5 | 75533.0 |
| 7 | Delaware | 7.0 | 4.097734e+04 | 3153.497677 | 36934.9 | 38833.65 | ... | 3515.459545 | 44404.0 | 46941.5 | 48734.0 | 51566.5 | 54217.0 |
| 8 | District of Columbia | 7.0 | 4.352287e+04 | 3974.844792 | 38295.1 | 40730.05 | ... | 6042.315931 | 67774.0 | 73546.0 | 78186.0 | 81346.0 | 84671.0 |
| 9 | Florida | 7.0 | 8.414587e+05 | 86881.111371 | 722610.0 | 782871.45 | ... | 4635.027041 | 41162.0 | 44587.5 | 46454.0 | 50287.5 | 54560.0 |
| 10 | Georgia | 7.0 | 3.651412e+05 | 35773.800531 | 317225.2 | 340186.85 | ... | 3920.693834 | 37794.0 | 40955.0 | 43033.0 | 45924.0 | 49083.0 |
| 11 | Hawaii | 7.0 | 5.966950e+04 | 4836.731483 | 53570.0 | 56330.65 | ... | 3706.373243 | 43931.0 | 46910.5 | 49122.0 | 51639.0 | 54700.0 |
| 12 | Idaho | 7.0 | 5.475580e+04 | 6382.456609 | 46862.6 | 50050.65 | ... | 3328.125899 | 36095.0 | 38533.0 | 40098.0 | 42337.0 | 45924.0 |

### Chart 1




We collected the data from 2013 to 2019, focusing on real GDP and rates of all crimes. We filtered the data by years. For each year, we calculated and used the mean values for real GDP and rates of all crimes in the line chart.

The line chart can show a clear trend for attributes. For chart 1, the

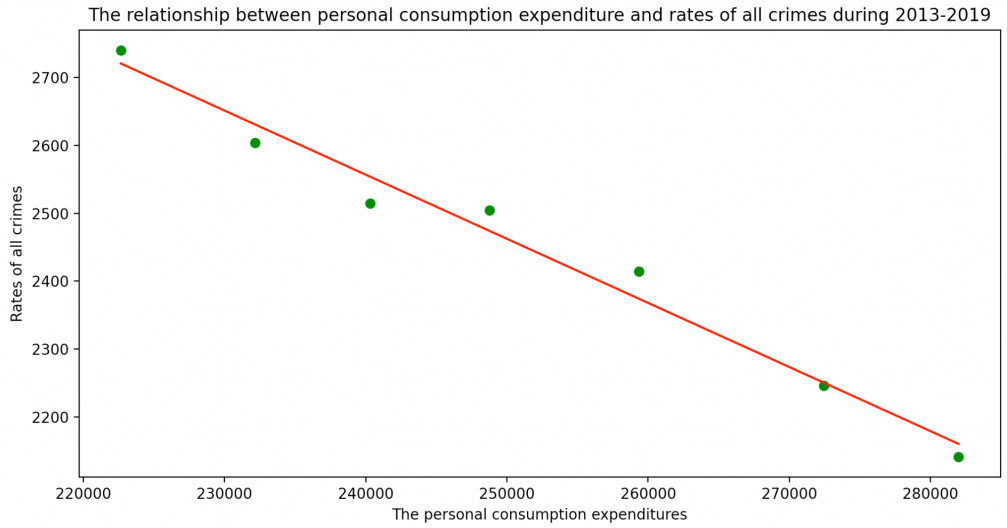
line chart displayed the relationship between real GDP and rates of all crimes from 2013 to 2019. The real GDP increased and the crime rates decreased during 2013-2019. We can further use a scatter plot to show the correlation between these two attributes, which the real GDP was plotted on x-axis and the rate of all crimes was plotted on y-axis.

For the code part of creating a line chart, the first line created a figure ‘fig’ and a set of axes ‘ax1’. Then, we set the ‘figsize’ parameter and the size of the figure. The next line created a bar plot on ‘ax1’. The x-axis is the years from 2013 to 2019. The crime rates were plotted on the y-axis.  Similarly, we created another bar plot on ‘ax2’. The real GDP were plotted on the y-axis. We used ‘twinx()’ to put these two bar plots in on graph by using different scales on each side of the graph. Finally, a title was added.

```
# Extract the year, crime rates and real GDP
years = grouped_sn['Year']
crime_rates = grouped_sn['Rates of all crimes', 'mean']
real_GDP = grouped_sn['Real GDP', 'mean']

# Create a line plot to show the relationship between the real GDP and the crime rates
fig, ax1 = plt.subplots(figsize = (8, 6))

ax1.plot(years, crime_rates, color = 'skyblue', marker='.', markersize=12, label = 'Crime rates')
ax1.set_xlabel('Years')
ax1.set_ylabel('The Crime Rates')
ax1.legend()

ax2 = ax1.twinx()

ax2.plot(years, real_GDP, color = 'orange', marker = '.', markersize = 12, label = 'Real GDP')
ax2.set_xlabel('Years')
ax2.set_ylabel('The Real GDP')
ax2.legend()

plt.title('The Real GDP vs The Crime Rates during 2013-2019')
plt.show()
```

**Chart 2**

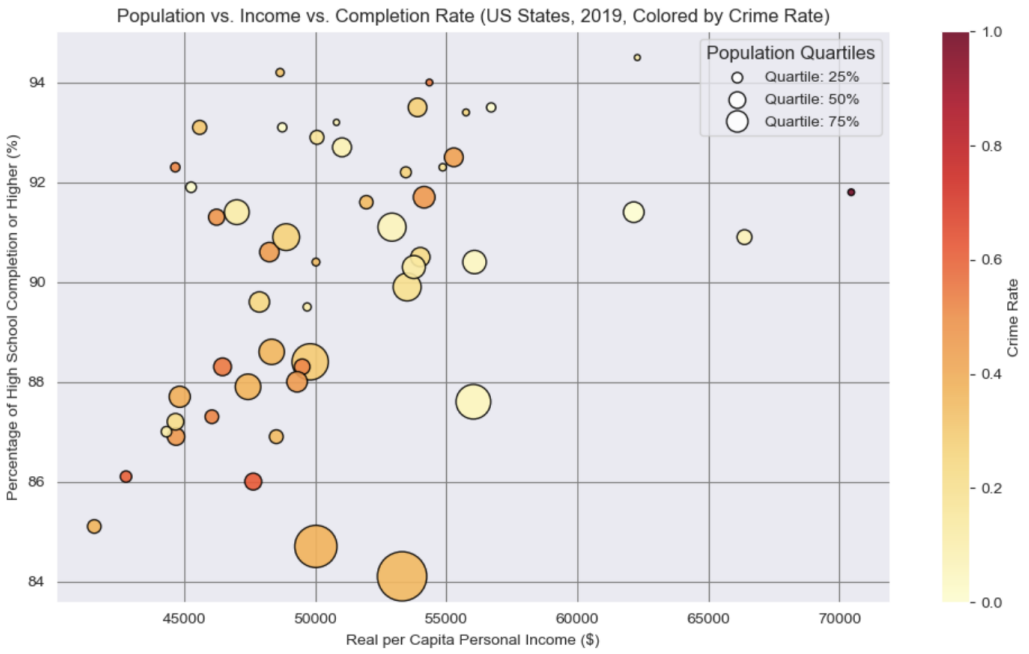
We collected the personal consumption expenditure and rates of all crimes from 2013 to 2019. For each year, we calculated the mean values for the personal consumption expenditure and the crime rate. Then, we created a scatter plot (chart 2), in which the personal consumption expenditure was plotted on the x-axis, and the rate of all crimes was plotted on the y-axis. For chart 2, the scatter plot displayed the correlation between personal consumption expenditure and the rate of all crimes from 2013 to 2019. When personal consumption expenditure increases, the rate of all crimes decreases. Thus, it means that the rate of all crimes was high in 2013, and people had low consumption due to several reasons such as income. As economic and other conditions improve, individual consumption levels increase year by year, and then the crime rate decreases. This approach is valuable for further analysis because it allows us to visually assess the correlation between personal consumption expenditure and rates of all crimes.




For the coding part of graphing the scatter plot, the first two lines named two attributes on the x-axis and y-axis. Then we used 'plt.scatter()' to create a scatter plot. The personal consumption expenditures were plotted on the x-axis. The rates of all crimes were plotted on the y-axis. A title was added, which named the relationship between personal consumption expenditure and rates of all crimes from 2013 to 2019. Finally, a regression line was added.

```
z = grouped_sn['Personal consumption expenditures', 'mean']
e = grouped_sn['Rates of all crimes', 'mean']

plt.scatter(z, e, color = 'green')
plt.xlabel('The personal consumption expenditures')
plt.ylabel('Rates of all crimes')
plt.title('The relationship between personal consumption expenditure and rates of all crimes during 2013-2019')

h = np.polyfit(z, e, 1)
p = np.poly1d(h)
plt.plot(z, p(z), color = 'red')
plt.show()
```

**4 attributes chart - GROUP SECTION**




From our merged dataset, named 'df', we first filtered it to select data specifically for the year 2019 before extracting the columns of our interest (i.e. 'State', 'Population', 'Rates of all crimes', 'Real per capita personal income', 'Percentage with high school completion or higher (%) total') and assigning each a distinct variable.

```
#Filter dataset for year 2019, and extract needed columns
df_2019 = df[df['Year'] == 2019]
state = df_2019['State']
pop = df_2019['Population']
crime_rate = df_2019['Rates of all crimes']
income = df_2019['Real per capita personal income']
completion_rate = df_2019['Percentage with high school completion or higher (%)  total']
```

We normalised the population data to use as bubble size references by calculating the maximum population value using 'max()', dividing each population value through and multiplying by 1000 as an adjustment factor. The scaled values are then stored into a list. The normalising procedure is similarly applied for the crime rate data but using min-max scaling, where each crime rate is scaled between 0 and 1 to ensure its usage as bubble determinant colours.

```
# Normalize the data for bubble sizes
max_pop = max(pop)
scaled_pop = [1000 * (p / max_pop) for p in pop]  # Adjust the scaling factor (1000)

# Normalize the crime rates between 0 and 1
crime_rate_min = min(crime_rate)
crime_rate_max = max(crime_rate)
normalized_crime_rates = [(c - crime_rate_min) / (crime_rate_max - crime_rate_min) for c in crime_rate]
```

To accurately represent the range of crime rates with our chosen colour palette 'YlOrRd', we used the 'Normalize()' function which maps the normalised crime rates to colours within the specified colourmap; with parameters of 0 and 1 given to the 'vmin' and 'vmax' arguments.

```
# Create a Normalize object to map normalized crime rates to colors in the colormap
normalize = Normalize(vmin=0, vmax=1)
```

The actual bubble chart is generated using '.scatter()', with arguments specifying the data values to be used for the x-axis, y-axis, bubble sizes and bubble colours, which in our case are the data for 'Real per cap PI', 'high-school completion %', scaled population, and normalised crime rates respectively.

```
# Create a scatter plot with bubble sizes and shading
plt.figure(figsize=(10, 6))
scatter = plt.scatter(income, completion_rate, s=scaled_pop, c=normalized_crime_rates,
                      cmap=cmap, alpha=0.9, edgecolors='k')
```

We set the chart style as dark using 'sns.set_style', with grey grid lines added to both the x and y-axis for visualisation purposes and easier data extraction.

```
sns.set_style("dark")
ax = plt.gca()
ax.grid(color='grey', axis='y')
ax.grid(color='grey', axis='x')
```

For the bubble scaling legend (based on population), we calculated the values representing the 1st, 2nd, and 3rd quartiles within the data population using NumPy's 'percentile()' function. This provides viewers with a reference guide to the relative bubble sizes as plotted on the chart and make appropriate estimates when comparing sizes. In terms of appearance, aside from providing a title 'Population Quartiles', we created custom lists containing the colours to be used for the 3 bubble sizes (white) and the names for legend markers (25%, 50%, 75%).

```
# Calculate quartiles for the population data
quartiles = np.percentile(pop, [25, 50, 75])  # Calculate 1st, 2nd (median), and 3rd quartiles

# Define a single custom color for the legend markers (all red)
legend_colors = ['white', 'white', 'white']  # Red color for all legend markers

# Create proxy artists for the legend with custom sizes and labels
legend_elements = []
for size, color, label in zip(quartiles, legend_colors, ['25%', '50%', '75%']):
    scaled_size = 1000 * (size / max_pop)  # Scale the size for the legend
    legend_elements.append(plt.scatter([], [], s=scaled_size, label=f'Quartile: {label}',
                                       c=color, edgecolors='k', alpha=0.9))

# Add the legend
legend = plt.legend(handles=legend_elements, loc='upper right', title='Population Quartiles')
plt.setp(legend.get_title(), fontsize='12')  # Set the font size of the legend title
```

Finally, appropriate labels were added to the x and y axes for context and a default colorbar was included on the right side of the plot for the normalised crime rates.

```
# Add labels and a color bar
plt.xlabel('Real per Capita Personal Income ($)')
plt.ylabel('Percentage of High School Completion or Higher (%)')
plt.title('Population vs. Income vs. Completion Rate (US States, 2019, Colored by Crime Rate)')
```

**Effectiveness of Communication**:

The bubble chart provides a comprehensive visualization of multiple data dimensions simultaneously—population, income, high school completion rates, and crime rates, enabling readers to easily compare and understand relationships among these variables. Meanwhile, the inclusion of a custom legend representing population quartiles allows viewers to interpret the data and understand the distribution of states by population while the colormap effectively represents crime rates using shades of color, with darker colors indicating higher crime rates. The 'YlOrRd' color scheme was chosen as it helps viewers quickly identify states with different crime rate levels. With the normalized population and crime rates and the addition of gridlines , we ensure that the bubble sizes and colors accurately represent the relative values within their respective datasets and aids in fair comparisons. The plot includes clear labels for the x-axis and y-axis, as well as a descriptive title, which enhances readability, clarity and interpretability of the visualization.

**Based on the code and data provided, we can observe the following potential correlations:**

Correlation between Income and High School Completion Rate:

From the chart, there appears to be a positive correlation between real per capita personal income and the percentage of high school completion or higher, as observed from the rising bubbles from bottom left to top right. The increase in personal income followed by % high-school completion rate suggests that states with higher income levels tend to have higher rates of high school completion or education.

Correlation between income and crime rate:

The color of the bubbles is determined by the crime rate, with darker colors indicating higher crime rates. In the visualization, states with higher incomes tend to have lighter-colored bubbles, indicating lower crime rates. Conversely, states with lower incomes tend to have darker-colored bubbles, indicating higher crime rates. From the color distribution, we can observe a negative relationship between income and crime rate.

**Evaluations**:

- Data quality: It is important to note that the quality and representativeness of the visualization heavily depend on the quality and representativeness of the dataset. Given our initial dataset had undergone a cleaning and filtering process beforehand, the chart produced can be justified to be reliable to a large extent based on the quality of data values.
- Population Size Influence: Although our chart reliably documents our data values through differently sized bubble plots (sizes of which are directly proportional to population), potential challenges to data interpretation may arise from the design principle of a bubble chart. Since more visual weight and attention is given to objects of greater surface area, it is likely that smaller bubble plots showcasing lower populated states are overlooked in comparison to higher populated states.
- Statistical Interpretation: Although the visualization provides a visual correlation between variables, it does not offer statistical analysis or measures of correlation coefficients. Viewers can only rely on visual inspection to assess relationships.
- Outliners and external factors: Plenty of outliers can be observed in the chart, indicating that external factors such as regional disparities, policy Interventions, and demographic differences can all contribute to the skewness of the distribution. Meanwhile, given that the chart only analyzes one year of data (2019), extra uncertainty is added to our investigation. However, the effect of the outliners and external factors can be minimized If more data is obtained.

Part B Group section

Intended audience:

A data analysis investigating the factors that affect crime rates across individual U.S states would captivate primarily viewers including policymakers, government officials, and law enforcement agencies, both at state and federal levels. This exploration of insights into the underlying trends and determinants stand to guide the development of effective strategies for crime prevention and intervention, implementation of resources, and formulation of public policies targeted at prioritising public safety and lower crime risks.

**Issue No.1 (Yining): Correlation between unemployment rate and crime rates**
**Table 2** indicates that regions characterized by higher levels of unemployment, particularly those rated as 'high' to 'extremely high,' exhibit significantly elevated risks of crime incidence. This observation underscores the pivotal role of economic conditions in influencing crime dynamics within individual U.S. states. It suggests that areas with higher unemployment rates may face additional challenges related to public safety and crime prevention.

Meanwhile, in **Figure 2**, supported by the steeper slope of the regression fit line, we can observe a downward trend in unemployment rates over time. The upward slant of the regression line indicates a positive correlation between unemployment rates and crime rates, which further demonstrates that as the unemployment rate increases, there is a corresponding increase in crime rates. Therefore, it is suggested that the US government develops initiatives that stimulate the economy, encourage entrepreneurship, and attract businesses which increase employment opportunities, thereby reducing the motivation for criminal activity due to economic hardship.

**Limitations and uncertainties**: Although our finding does illustrate that unemployment rate is associated with crime rates, it is worth emphasizing that correlation does not imply causation, and numerous external factors may influence crime rates beyond unemployment levels. Meanwhile, it is crucial to acknowledge the presence of limited data points and outliers, both of which introduce elements of uncertainty into the observed trends.

**Issue No.2 (Emily)**
The issue in this section involves an investigation into the relationship among personal income, GDP, and crime statistics specific states in the US. A side-by-side analysis of the bar plots (**Chart 1 & 2** ) representing income and crime data reveals various trend discrepancies across multiple states. Whereas particular states such as California, Florida, and Texas, show a positive correlation between their economic and crime numbers (higher incomes and GDP are associated with elevated crimes), regional nuances are also evident across the remaining states. This is illustrated particularly in the case of New York, in which it maintains strong economic metrics while displaying relatively lower total crime numbers compared to similar states. Such regional variations suggest that while economic prosperity can contribute to higher crime, there are other potential factors that may influence the relationship, leading to divergent patterns in some states where increasing income or GDP coincides with reduced crime rates.

For further analysis, when considering the economic status of the entire US as opposed to its individual states, we can observe from **Table 2** that a definitive positive correlation lies between disposable personal income and total crime numbers.

**Limitations/uncertainties:**

- While economic statistics are significant, it's essential to consider multiple factors like income inequality, demographics, population density, and policy decisions affecting crime variations. High-density states like California, Florida, and Texas, especially in major metropolitan areas, offer more opportunities for crime due to increased competition for resources and social challenges.
- It's crucial to remember that correlation doesn't imply causation when interpreting the table. While higher DPI can lead to increased wealth and susceptibility to theft in some cases, this relationship varies across regions and doesn't apply uniformly to all crime types.

**Issue No.3 (Average unemployment rate vs Average percentage with high school completion or higher - Jack)**

It is evident that throughout the years, when the average unemployment rate is relatively high, the average percentage of people with high school completion or higher tends to be relatively low. This displays a negative correlation between the two attributes (**Table 1 and Chart 2**). It can be said that the higher the education level, the lower the unemployment rate. Mentioned before, as the unemployment rate increases, crime rate also increases. As a result, it is important for both government and non-profit organisations interventions to take place to ensure the continuous increase of the level of education across all states of the country, especially in the states where there is a large discrepancy in areas based on different socio-economic levels. Explored previously, it is important and necessary to take into consideration the income disparity, demographics, population density of all states, and develop a specialised and targeted plan to most effectively improve the level of education in different states.

**Issue No.4 (Danica)**

A scatter plot reveals the correlation between personal consumption expenditure and rates of all crimes from 2013 to 2019 (**Chart 2, Table 1**). When the average personal consumption expenditure is high, the average percentage of crime rates will be relatively low. This displays a negative correlation between these two attributes. For the table and chart, they displays that the real GDP increased from 2013 to 2019, and the rate of crime rate decreased. Therefore, it is vital for the government to develop the economy to ensure citizens' incomes for living, the increasing numbers of public facilities, and so on.

**Conclusion**:

In summary, our research reveals a strong correlation between higher unemployment rate and increased crime rates in the United States. This highlights the importance of addressing unemployment as a potential factor in reducing crime rates and improving overall societal well-being. Furthermore, the data suggests an inverse relationship between unemployment rates and education levels, emphasising the importance of educational interventions, particularly in states with significant socio-economic disparities, to potentially reduce unemployment and associated crime rates. To achieve this, tailored strategies addressing income disparities, demographics, and population density should be developed and implemented .