SI 206 Final Project Report

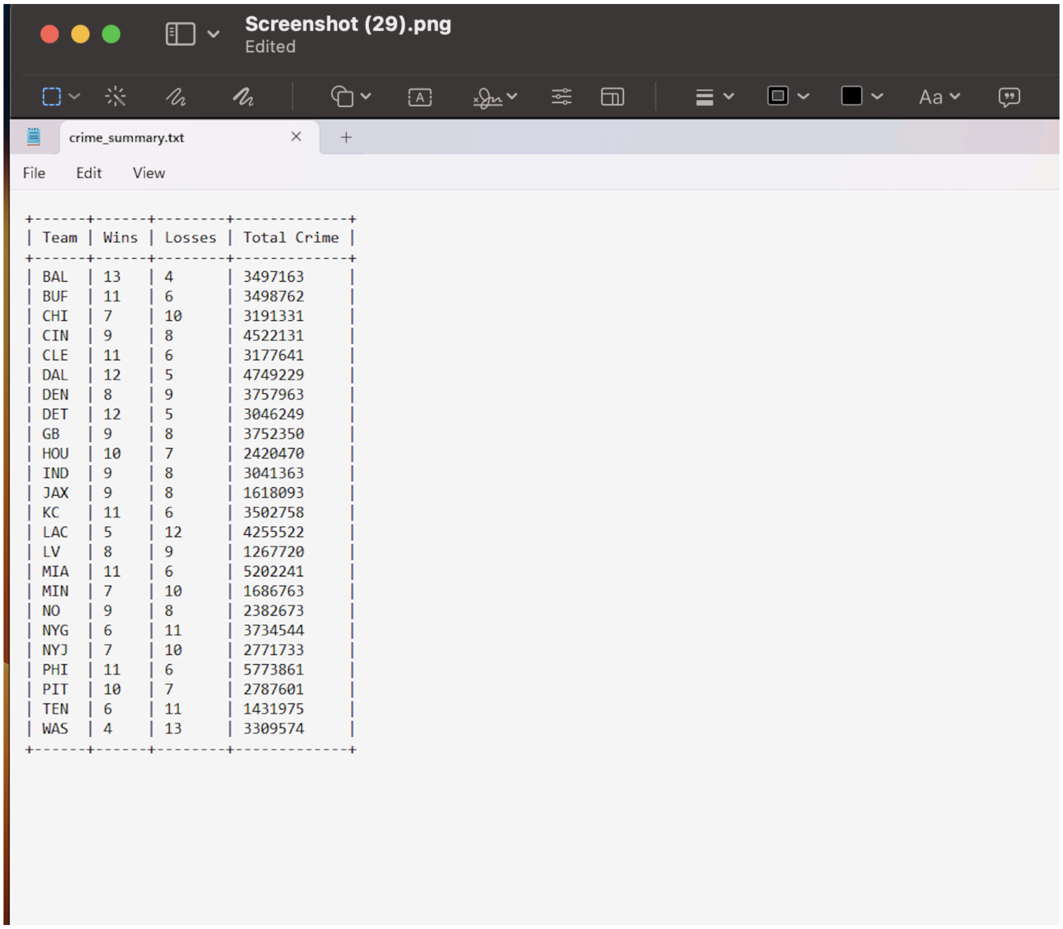
Jonathan Jipping and Connor Nolan

The primary goal of our final project was to determine if there was a correlation between NFL team performance and the level of crime in an area. We aimed to achieve this goal by working with APIs and creating visualizations to derive meaningful insights from our collected data. Specifically, this project aimed to work with the SportsData API to gather data on NFL teams regarding their wins, losses, and team name. We also hoped to use the FBI Crime Data API to collect various crime statistics and connect them to the matching NFL team. Our last goals for this project were to store all of this retrieved data in a well organized database, and to create effective graphs based on this data to demonstrate the correlation visually.

After the project has been completed, we have completed many of our goals listed above. We successfully used the SportsData API to fetch data on NFL teams and stored it in a database table called "NFL_Data." Similarly, we utilized the Crime Data API successfully and stored that data in the "Crime_Data" table. Using these APIs, we did successfully map the crime data to the matching NFL team and visualize the correlation between NFL team performance and level of crime, however the correlation did not seem to be as strong as we had originally believed. So, we did indeed put our data in an organized database, and we created useful graphs such as histograms, scatter plots, and pie charts.

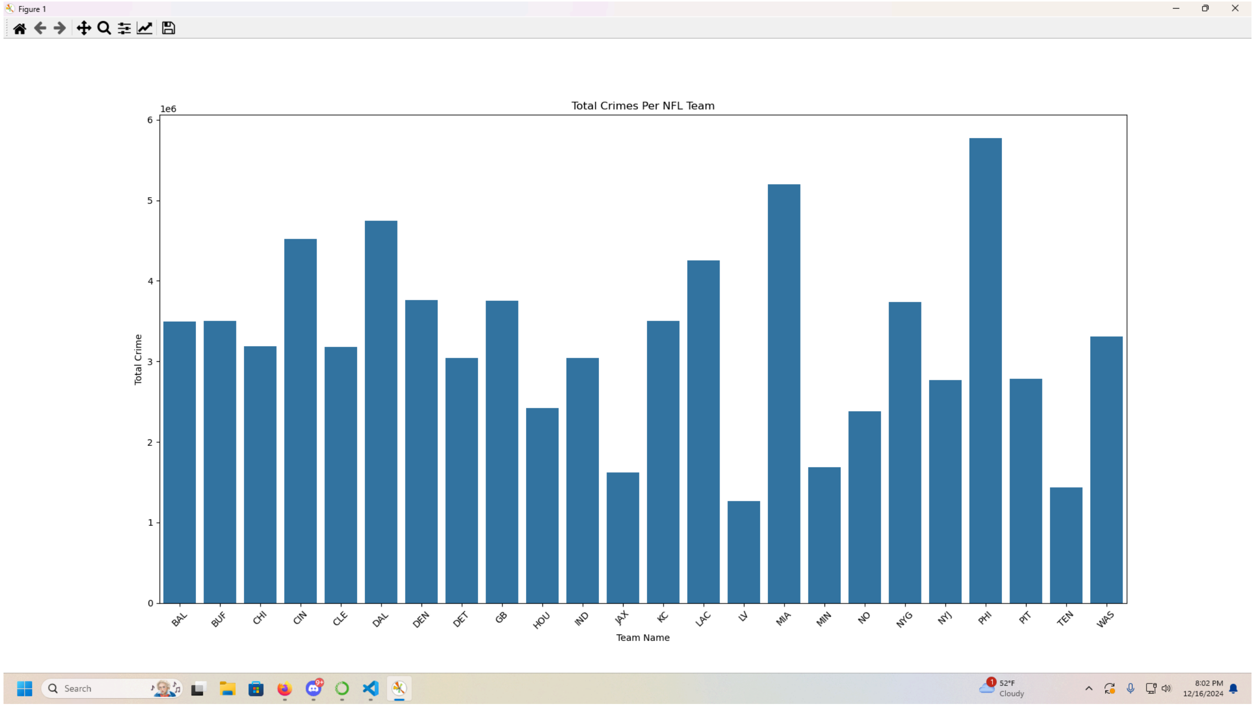
Although many of our goals were completed successfully, we did run into quite a few problems along the way of developing this project. First off, we struggled to find an API that did exactly what we wanted to, as the SportsData API could only be used for one season's data at a time, so we could not analyze the full spread of data that we had hoped to. Additionally, we realized a bit too late that our API would only output 32 items as there are only 32 teams, so we struggled to successfully hit 100 pieces of data in that API pull. The FBI crime data API would also randomly fail on us even if it had worked moments ago, which was a very frustrating process to work through. Another API issue was that the crime API we found could only group crime on a monthly basis, so we were not able to get specific enough to look at a specific spike right after a game or during the game week. We also struggled to format the graphs properly as it was difficult to figure out what visualizations would best demonstrate the correlation we were looking for. Overall, we did experience many obstacles on the path of completing this project, but eventually we were able to overcome them and present a successfully running code.

Here is the calculation file created:

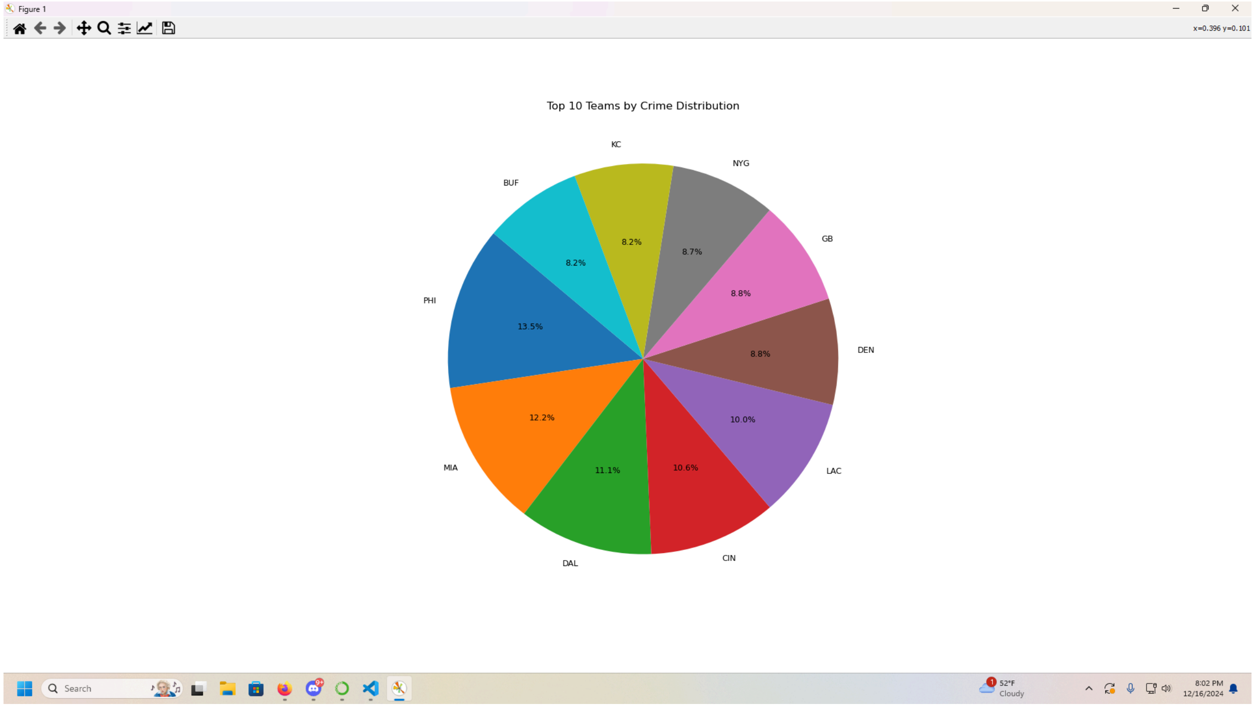


```
+------+------+--------+--------------+
| Team | Wins | Losses | Total Crime  |
+------+------+--------+--------------+
| BAL  | 13   | 4      | 3497163      |
| BUF  | 11   | 6      | 3498762      |
| CHI  | 7    | 10     | 3191331      |
| CIN  | 9    | 8      | 4522131      |
| CLE  | 11   | 6      | 3177641      |
| DAL  | 12   | 5      | 4749229      |
| DEN  | 8    | 9      | 3757963      |
| DET  | 12   | 5      | 3046249      |
| GB   | 9    | 8      | 3752350      |
| HOU  | 10   | 7      | 2420470      |
| IND  | 9    | 8      | 3041363      |
| JAX  | 9    | 8      | 1618093      |
| KC   | 11   | 6      | 3502758      |
| LAC  | 5    | 12     | 4255522      |
| LV   | 8    | 9      | 1267720      |
| MIA  | 11   | 6      | 5202241      |
| MIN  | 7    | 10     | 1686763      |
| NO   | 9    | 8      | 2382673      |
| NYG  | 6    | 11     | 3734544      |
| NYJ  | 7    | 10     | 2771733      |
| PHI  | 11   | 6      | 5773861      |
| PIT  | 10   | 7      | 2787601      |
| TEN  | 6    | 11     | 1431975      |
| WAS  | 4    | 13     | 3309574      |
+------+------+--------+--------------+
```

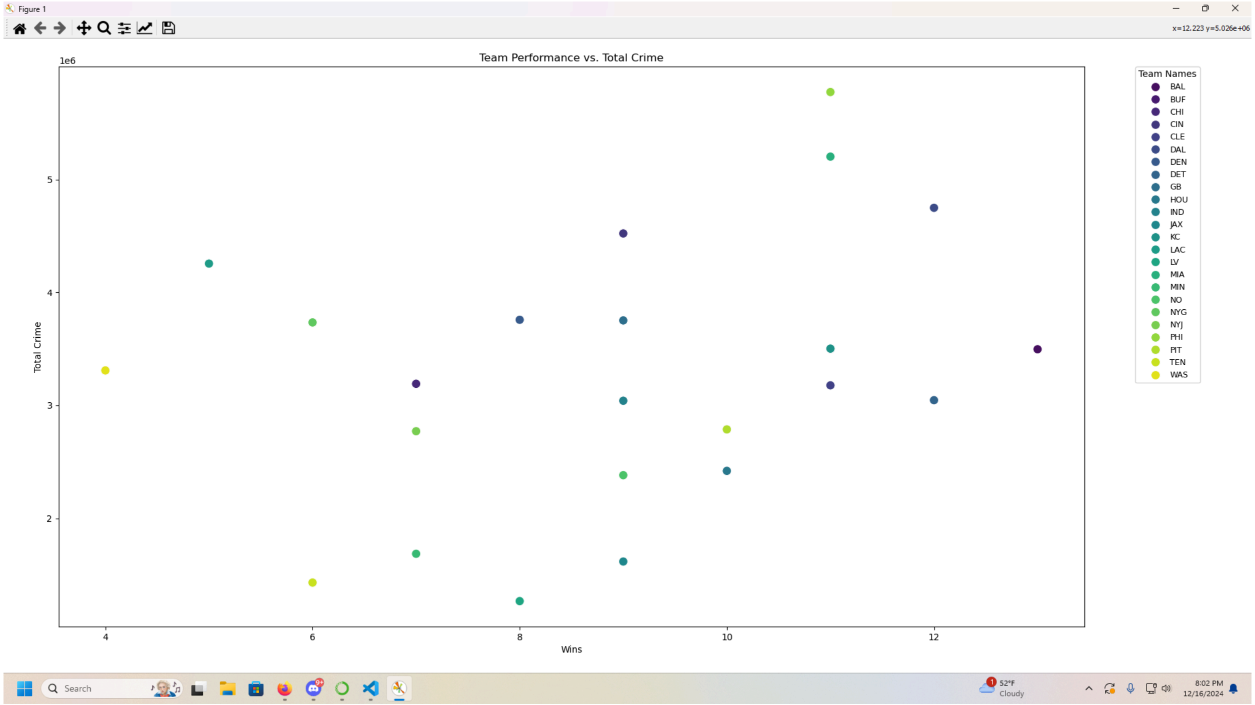
Here are all of the graphs created:

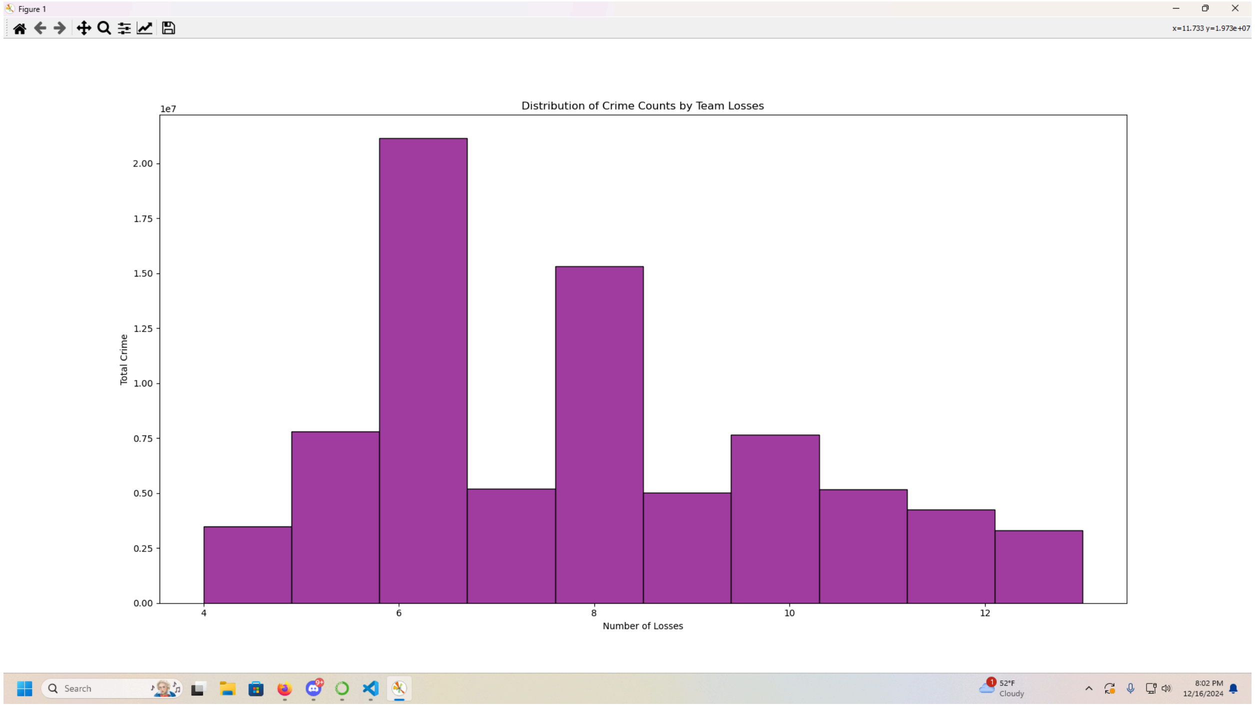


Total Crimes Per NFL Team



Top 10 Teams by Crime Distribution





Now that all that has been shown, we will go through a few easy steps to run our code. First, ensure Python is installed along with the following libraries: sqlite3, pandas, matplotlib, seaborn, and requests. Then, save the provided code script to a python file and run it. Once it is run, the code will execute and the various visualizations will be displayed, each one can be closed out to view the next. If you do wish to change the database name to see if it populates on its own, there is a bit of a process to ensuring the database will correctly change. Currently, the name of the database is “sports_crime.db” and if you wish to change this you must change that name anywhere you see on the code to whatever you like, as long as the .db remains. The lines in which this name will need to be changed are as follows: 11, 48, 84, 147, 172, 198, 229, and 276. Once the name is changed in each location, the database should populate as you run the code now. That is all there is to running this code, as the main output is simply the visualizations which will automatically pop up once the code is run.

We have 8 different functions that we will now go over and discuss the input and output of each function. The first is the setup_database() function which effectively sets up the framework for our database. This requires no input and creates the sports_crime db and tables NFL_data and Crime_data as an output. The fetch_and_store_nfl_data() function has no input. This function fetches NFL data from the API and populates the NFL_data table as an output. Next up is a similar function, fetch_and_store_crime_data() which again has no input, and it results in a populated crime_data table after fetching crime data from the crime API. The calculate_and_write_summary() function, like the rest, has no input. It does output with a written file summary of team performance and the total crimes in that area in crime_summary.txt. The last four functions we will discuss are all visualization functions. So, starting off with plot_crime_counts(), this function again has no input, and it displays a bar chart of the total crimes by NFL team once it is executed. This bar chart organizes crime by location and maps it to its matching NFL team. The next graph is plot_crime_distribution by team, which (with no input) displays a pie chart of the crime distribution of the top 10 NFL teams with the most crime. It demonstrates which of the top 10 areas with the most crime are taking up the biggest proportion of that crime based on NFL team. Next is plot_team_performance_vs_crimes() which has no required input and produces a scatter plot of a teams performance against total crime count, so it helps visualize what correlation there is between a teams wins and how much crime is occuring. The last graph, and last function as a whole is plot_crime_distribution_by_losses(), which again has no input. This function outputs a histogram of the amount of crime counts by team losses. For example, it would show the total amount of crime counts by teams with 1 loss, 2 losses, 3 losses and so on.

Resources Used:

Date	Issue Description	Location of Resource	Result
12/1/2024	Crime Data API issue successfully pulling data	https://cde.ucr.cjis.gov/LATEST/webapp/#/pages/docApi	Through a bit of struggle, eventually got it working
12/6/2024	Could not figure out how to properly format our desired graphs in the way we wanted	https://matplotlib.org/	Got graphs looking as desired
12/7/2024	Errors with data parsing correctly	https://pandas.pydata.org/docs/	Solved using pd.to_datetime
12/9/2024	Random API errors with SportsData API functioning	https://stackoverflow.com/	Eventually trouble shooted and fixed issue