D610 Data Engineering Capstone

Task 2: Data Analysis Report

JoAnne Moore

May 9, 2025

A. Research Question

The research question I identified for this project is: Do 911 call times differ significantly between the high-income neighborhood of Canton and the low-income neighborhood of Druid Heights in Baltimore? This question stems from the need to understand how emergency call patterns vary across different socioeconomic areas of the city. By analyzing 911 call times in neighborhoods with contrasting income levels, I can potentially uncover important insights into how resource allocation, emergency response times, and community needs might differ depending on the neighborhood's economic status.

Canton, with its higher median income ($149,999), may experience different patterns of 911 calls compared to Druid Heights, a neighborhood with a significantly lower median income ($25,326). This difference could be driven by a variety of factors, such as the types of incidents being reported, the population density, or the availability of emergency services. By focusing on the timing of 911 calls, specifically whether certain times of day see a higher volume of calls in one neighborhood over the other, this research aims to highlight any disparities that could influence how emergency services are deployed.

The null hypothesis (H_0) for this analysis is that there is not a statistical difference in the distribution of 911 call times between Canton and Druid Heights. My alternative hypothesis (H_1) is that there is a statistical difference in the distribution of 911 call times between Canton and Druid Heights. This hypothesis was tested using a chi-squared test, which examined whether the distribution of 911 calls across different times of day (categorized as Night, Morning, Afternoon, and Evening) differs between the two neighborhoods. The outcome of this analysis can provide

useful insights for local authorities and emergency responders, allowing them to tailor their operations and interventions based on the unique patterns in each neighborhood.

B. Data Collection

For this project, I collected 911 call data from the Open Baltimore dataset, which includes detailed records of emergency calls made throughout Baltimore City in 2024. The dataset contains information on the call times, neighborhoods, incident types, and other variables such as district and ZIP code. I specifically focused on data for two neighborhoods: Canton (high-income) and Druid Heights (low-income). This allowed for a direct comparison of 911 call times between the two neighborhoods with contrasting income levels. The data was filtered to include only relevant columns, such as the call time and neighborhood, to streamline the analysis process.

One advantage of using a datasets that has been made publicly available is the accessibility and comprehensiveness of the data. The dataset contains a large number of entries (over a million rows), which provided a robust sample size for analysis. This enabled me to perform a meaningful comparison between the neighborhoods and identify trends that could support decision-making for emergency services and resource allocation.

However, a disadvantage of using this dataset is that some data may be incomplete or inaccurate. For example, certain call records were missing geographic or time-related information, and there was no explicit column for call duration, which would have been useful in analyzing response times. To overcome these challenges, I relied on data cleaning techniques to address missing or inconsistent values. I also ensured that only the relevant columns were included, reducing the

impact of incomplete data and focusing the analysis on the core variables of interest, like call times and neighborhoods. By carefully filtering and preprocessing the dataset, I was able to overcome these limitations and maintain the integrity of the analysis.

C. Data Extraction and Preparation

I began the data extraction and preparation process by downloading the 911 call dataset from the Open Baltimore platform, which was in CSV format. In Jupyter Lab I primarily used Pandas, which is a powerful Python library for data manipulation. The read_csv() function in Pandas enabled me to efficiently load the data from the CSV file into a DataFrame, where I could easily handle missing values, convert columns into appropriate data types, and filter the dataset. I focused on columns such as callDateTime and Neighborhood, ensuring that only relevant data was retained for analysis. I then transformed the callDateTime column into a usable datetime format and extracted the hour of the call to later categorize the times into bins such as Night, Morning, Afternoon, and Evening. Using Python's apply() function to create the custom bins allowed for a clear comparison of call distributions between neighborhoods. I also used visual methods such as graphs and statistical methods (mean, median, mode) to get a good understanding of my data. This technique helps to identify patterns and detect outliers. This approach provided the flexibility needed to transform the data into a suitable format for the analysis (WGU, 2020).

The main advantage of using Pandas for data extraction and preparation is its flexibility and efficiency. With its powerful functions for cleaning, transforming, and manipulating large datasets, Pandas made it easy to perform tasks such as handling missing data, converting time formats, and applying custom functions for binning call times. The ability to manage large

datasets and carry out complex operations with minimal code is a significant benefit, allowing for detailed, customized data preparation (WGU, 2020).

A disadvantage of using Pandas and other Python libraries is the learning curve associated with them. While these tools are incredibly powerful, they can be challenging for beginners, especially when working with large datasets or complex data manipulations. The need to write custom code, such as defining custom functions for binning, can be time-consuming and require a strong understanding of Python programming. Despite this, the trade-off is generally worth it, as these tools offer unmatched flexibility for data extraction and preparation (WGU, 2020).

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# Load the dataset
df = pd.read_csv(r"C:\Users\joann\Documents\WGU\D610\911_Calls_For_Service_2024.csv")

# Display basic info and a sample
print(df.info())
print(df.head())
```

```
# Convert callDateTime to datetime format
df['callDateTime'] = pd.to_datetime(df['callDateTime'], errors='coerce')
```

```
# Drop rows with invalid or missing callDateTime
df = df.dropna(subset=['callDateTime'])
```

```
# Extract the hour from the datetime
df['callHour'] = df['callDateTime'].dt.hour
```

```
# Normalize neighborhood names to avoid casing issues
df['Neighborhood'] = df['Neighborhood'].str.strip().str.title()
```

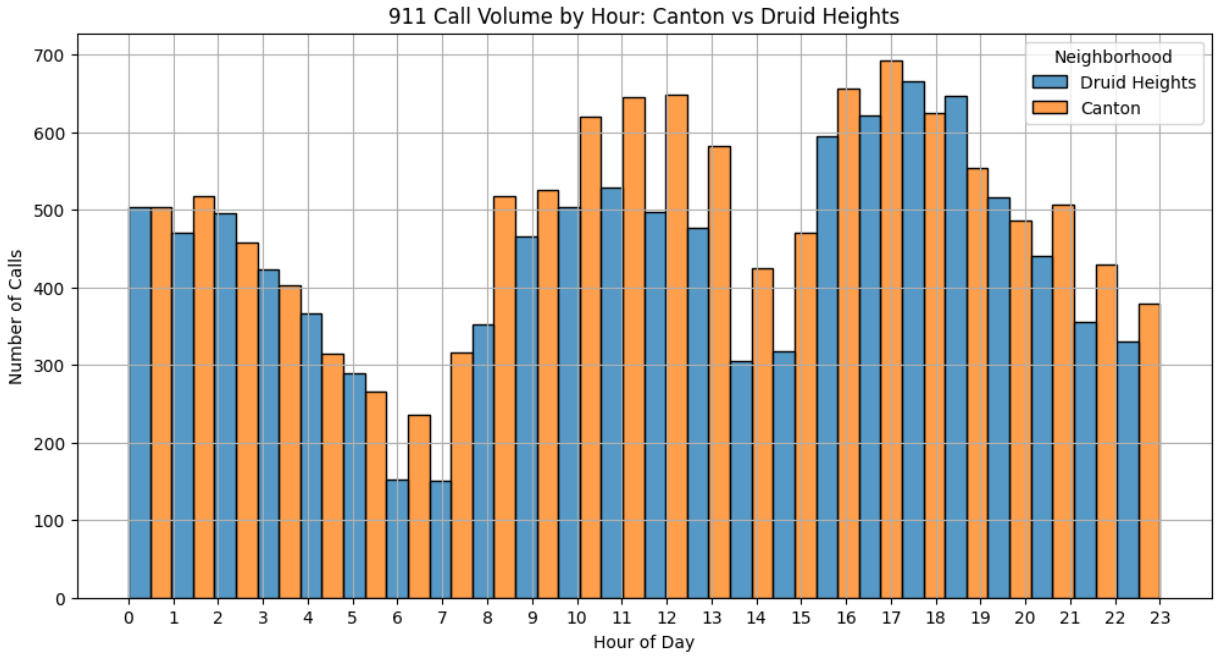
```
# Filter for just Canton and Druid Heights
df_filtered = df[df['Neighborhood'].isin(['Canton', 'Druid Heights'])]
```

```
# Check how many records we have for each neighborhood
print(df_filtered['Neighborhood'].value_counts())
```

```
Neighborhood
Canton          11779
Druid Heights    10467
Name: count, dtype: int64
```

```
# Optional: Save the cleaned and filtered dataset
df_filtered.to_csv('filtered_911_calls_canton_druid.csv', index=False)
```

```
plt.figure(figsize=(12, 6))
sns.histplot(data=df_filtered, x='callHour', hue='Neighborhood', multiple='dodge', bins=24, kde=False)
plt.title('911 Call Volume by Hour: Canton vs Druid Heights')
plt.xlabel('Hour of Day')
plt.ylabel('Number of Calls')
plt.xticks(range(0, 24))
plt.grid(True)
plt.show()
```



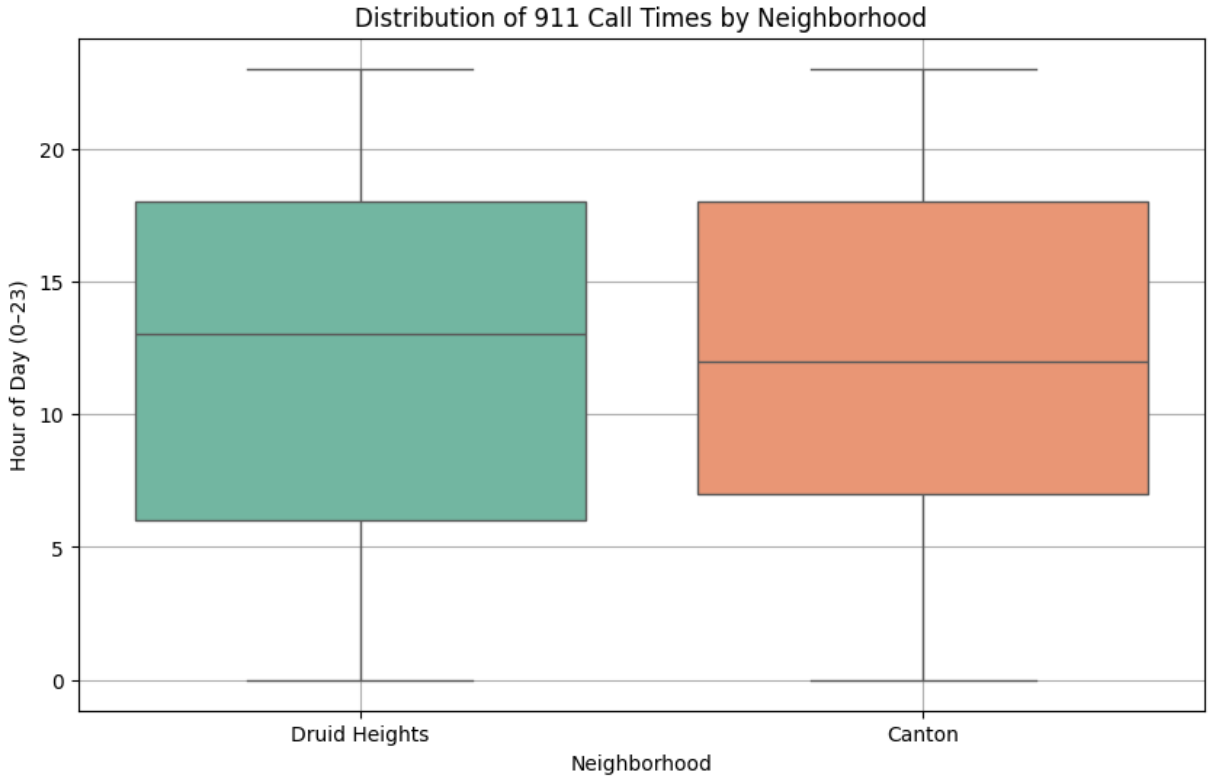

```
# Group by Neighborhood and calculate descriptive stats
grouped_stats = df_filtered.groupby('Neighborhood')['callHour'].agg(['mean', 'median', pd.Series.mode])

# Reset index for cleaner display
grouped_stats = grouped_stats.reset_index()
grouped_stats.columns = ['Neighborhood', 'Mean Call Hour', 'Median Call Hour', 'Mode Call Hour']

print(grouped_stats)
```

```
    Neighborhood  Mean Call Hour  Median Call Hour  Mode Call Hour
0         Canton       12.054419              12.0              17
1  Druid Heights       12.034680              13.0              18
```

```
plt.figure(figsize=(10, 6))
sns.boxplot(data=df_filtered, x='Neighborhood', y='callHour', hue='Neighborhood', palette='Set2', legend=False)
plt.title('Distribution of 911 Call Times by Neighborhood')
plt.ylabel('Hour of Day (0-23)')
plt.xlabel('Neighborhood')
plt.grid(True)
plt.show()
```



D. Analysis

For the analysis, I applied the chi-squared (χ^2) test of independence to determine if there is a statistical difference in the distribution of 911 call times between two neighborhoods: Canton and Druid Heights. I first categorized the call times into bins (Night, Morning, Afternoon, and Evening) based on the hour of the call, and then compared the frequency of calls in each time bin between the two neighborhoods. The formula for the chi-squared statistic is $\chi^2 = \Sigma[(O - E)^2 / E]$, where *O* is the observed frequency in each cell of the contingency table, and *E* is the expected frequency, calculated as (*row total* × *column total*) ÷ *grand total*. I repeated this calculation for each of the eight cells (4 time bins × 2 neighborhoods) and summed the results to produce the

final chi-squared value. The result from the Chi-squared test showed a value of $\chi^2 = 94.99$ with 3 degrees of freedom, resulting in a p-value less than 0.001 (0.0000), indicating a statistically significant difference between the call times of the two neighborhoods (WGU, 2020).

The reason I selected the Chi-squared test is that it is a well-suited method for comparing categorical variables to assess whether the distribution of call times in different neighborhoods follows the same pattern. Given that I was working with binned call times (a categorical variable), the Chi-squared test was appropriate because it helps determine whether differences in observed frequencies are due to chance or a real statistical difference. It is a straightforward, widely used technique for hypothesis testing when dealing with categorical data.

One advantage of using the Chi-squared test is that it is simple to implement and provides a clear indication of whether the differences between groups are statistically significant. Since it is a non-parametric test, it does not require assumptions about the distribution of the data. This makes it useful when the data may not follow a normal distribution. This allows for greater flexibility when working with categorical data like the binned call times in this analysis.

A disadvantage of the Chi-squared test is that it requires a sufficient sample size to produce reliable results. Small sample sizes in any of the categories can lead to expected frequencies that are too low, which can invalidate the test. In this analysis, while the large dataset helped mitigate this issue, it’s still something to be mindful of when working with smaller datasets or when categories have few observations. Additionally, the Chi-squared test doesn't provide insights into the magnitude or direction of the differences, so further analysis might be necessary to understand the practical significance.

```
df_filtered = df_filtered.copy()
df_filtered['TimeBin'] = df_filtered['callHour'].apply(categorize_time)
```

```
# Add binned times
def categorize_time(hour):
    if 0 <= hour < 6:
        return 'Night'
    elif 6 <= hour < 12:
        return 'Morning'
    elif 12 <= hour < 18:
        return 'Afternoon'
    else:
        return 'Evening'

# Apply the categorization to the 'callHour' column
df_filtered.loc[:, 'TimeBin'] = df_filtered['callHour'].apply(categorize_time)
```

```
# Create contingency table
contingency_table = pd.crosstab(df_filtered['Neighborhood'], df_filtered['TimeBin'])
print(contingency_table)
```

```
TimeBin         Afternoon  Evening  Morning  Night
Neighborhood
Canton               3477     2980     2861   2461
Druid Heights        2813     2953     2152   2549
```

```
from scipy.stats import chi2_contingency

# Perform the chi-squared test
chi2, p, dof, expected = chi2_contingency(contingency_table)

# Output results
print(f"Chi-squared: {chi2:.2f}, p-value: {p:.4f}")
```

```
Chi-squared: 94.99, p-value: 0.0000
```

E. Data Summary and Implications

The results of the Chi-squared test revealed a statistically significant difference in the distribution of 911 call times between Canton and Druid Heights, with the p-value of 0.0000 indicating that the differences were unlikely to have occurred by chance. Because the chi-squared test resulted in a p-value of less than 0.05, we reject the null hypothesis and conclude that there is a statistically significant difference in the distribution of 911 call times between Canton and Druid Heights. Based on the contingency table, Canton showed a higher volume of calls in the afternoon, with 3,477 calls compared to 2,813 in Druid Heights. Conversely, Druid Heights had a

more evenly distributed call volume across all time bins, with relatively similar numbers of calls in the afternoon, evening, and night. This suggests that while Canton's call volume is more concentrated in specific parts of the day, Druid Heights experiences 911 activity more consistently throughout the day and night. This information could be critical for resource allocation and emergency response planning, indicating that certain neighborhoods might require different approaches to ensure optimal response times.

A limitation of this analysis is that it only focuses on call times, which may not fully capture the complexity of 911 call patterns in different neighborhoods. It does not account for the types of incidents or the severity of those calls, which could provide a more comprehensive understanding of emergency service needs across neighborhoods.

Based on these results, I recommend that emergency services consider tailoring their resources based on the time-specific demands in different neighborhoods. For example, if certain time periods have higher volumes of calls in Canton, additional staffing or quicker response mechanisms could be implemented during those times. In Druid Heights, where calls are spread out more evenly, a more generalized approach could be effective.

For future studies, two potential directions include:

1. **Analyzing the impact of incident type** on call times: By including the types of calls (e.g., medical emergencies, noise complaints, etc.), a deeper understanding of the relationship between neighborhood demographics and emergency service demand could be gained. This could inform even more targeted intervention strategies.
2. **Expanding the time frame of the analysis**: Analyzing 911 call data over a longer period or across multiple years could help identify any trends or shifts in emergency response

patterns, potentially offering more insights into seasonal or long-term changes in call behavior across neighborhoods.

References

Western Governors University - WGU (2020) *Course* |. (n.d.). Retrieved April 29, 2025, from https://apps.cgp-oex.wgu.edu/learning/course/course-v1:WGUx+OEX0399+v01/block-v1:WGUx+OEX0399+v01+type@sequential+block@64a5dec90eb34cd39ff0ae54ddf1de32/block-v1:WGUx+OEX0399+v01+type@vertical+block@dc1ec1f7a21840b4805a59e629eb1701

Western Governors University - WGU (2020) *Course* |. (n.d.). Retrieved March 10, 2025, from https://apps.cgp-oex.wgu.edu/learning/course/course-v1:WGUx+OEX0395+v01/block-v1:WGUx+OEX0395+v01+type@sequential+block@a908f775b256412e8918b824e9ef98f9

Appendix

Dashboard for Stakeholders: 911 Call Time Analysis: Canton vs Druid Heights

https://public.tableau.com/views/NeighborhoodsCantonvsDruidHeights/Dashboard1?:language=en-US&:sid=&:redirect=auth&:display_count=n&:origin=viz_share_link

Dataset: Open Baltimore

https://data.baltimorecity.gov/datasets/5d378673c8f4427fb9d02de362d5b634_0/explore