Impact of Area on Rates of Medical Readmission

Gerald Burke

Western Governors University

D207: Exploratory Data Analysis

Dr. William Sewell

October 14, 2024

Contents

A. Real World Issue

Medical readmissions are a significant problem in healthcare. The extent of the problem is that a hospital can be fined if readmissions rise above a target threshold. I have been tasked with examining the relationship between readmission rates and other factors to determine what can be done to reduce the rates of readmission and reduce the associated penalties.

A1. Research Question

Does the area that a hospital serves have an impact on the rates of readmission?

Null Hypothesis

The area that the hospital serves does not have an impact on the rate of readmission.

Alternative Hypothesis

The area that the hospital serves does have an impact on the rate of readmission.

A2. Stakeholders Benefits

The availability and quality of healthcare available in rural areas vs. urban or suburban areas is a question of major concern in the country today. For larger hospital chains like ours, we have some flexibility in terms of how and where we apply funding and resources. If we are able to determine a discrepancy in the rates of readmissions between areas, we have a path to changing outcomes for our patients and decreasing the financial and reputational ramifications of receiving large, recurring fines.

A3. Relevant Data

ReAdmis

- Categorical
- Example: 'Yes'
- ReAmis is defined as whether a patient was readmitted within a month of release

Area

- Categorical
- Example: Rural
- Area is defined as the type of area (rural, urban, suburban) the hospital serves

B. Analysis

B1. Chi-square Test in Python

Below is the code I wrote to perform the test:

```
#Chi-square test
import pandas as pd
import scipy.stats as stats

df = pd.read_csv('medical_clean.csv')
```

```
#Establish the alpha
alpha = 0.05

#Create the contingency table
contingency_table = pd.crosstab(df['ReAdmis'], df['Area'])

#Print the contingency table
print(contingency_table)

#Perform the test
chi2, p, dof, expected = stats.chi2_contingency(contingency_table)

#Print the results
print("Chi-square:", chi2)
print("P-value:", p)
print("The finding is significant" if p < alpha else "The finding is not
significant")
```

B2. Results

```
Chi-square: 0.7133125620168337
P-value: 0.7000130641731285
The finding is not significant
```

B3. Justification of Analysis Method

Given the analysis is regarding two categorical variables, I opted to use a chi-square test.

I opted to perform the test in Python, primarily due to my familiarity with the language. I leveraged the SciPy Library to perform the test.

I tested with an alpha of 0.05 as it fit standard definitions of certainty.

C. Univariate Statistics

Marital

The code I wrote to analyze the Marital variable is:

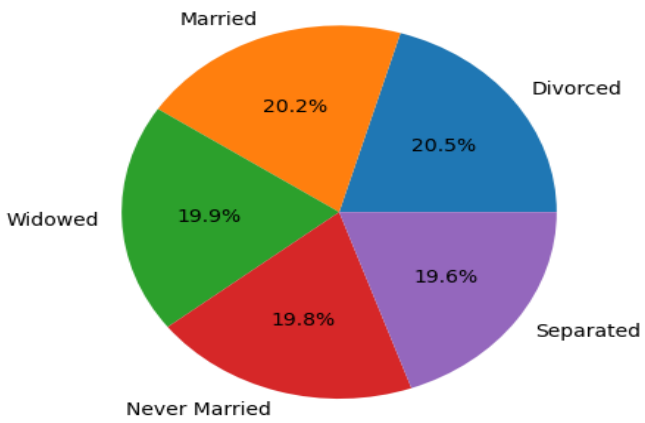
```
#Print relevant information about the variable
print('++++==== Begin ====++++\n')
print('++++==== Description of Marriage ====++++\n')
print(df['Marital'].describe())
print('++++==== Unique Values in Marriage ====++++\n')
print(df['Marital'].unique())
print('++++==== Counts of Values in Marriage ====++++\n')
print(df['Marital'].value_counts())
print('++++==== Percentages of Values in Marriage ====++++\n')
print(np.round(df['Marital'].value_counts() / df['Marital'].count() * 100,
2))
print('++++==== End ====++++\n')

#https://matplotlib.org/stable/gallery/pie_and_polar_charts/pie_features.html
fig, ax = plt.subplots()
ax.pie(df['Marital'].value_counts(), labels=df['Marital'].unique(),
autopct='%1.1f%%')
```

```
plt.show()
```

The results of the report are:

```
++++==== Begin ====++++

++++==== Description of Marriage ====++++

count       10000
unique          5
top       Widowed
freq         2045
Name: Marital, dtype: object
++++==== Unique Values in Marriage ====++++

['Divorced' 'Married' 'Widowed' 'Never Married' 'Separated']
++++==== Counts of Values in Marriage ====++++

Marital
Widowed          2045
Married          2023
Separated        1987
Never Married    1984
Divorced         1961
Name: count, dtype: int64
++++==== Percentages of Values in Marriage ====++++

Marital
Widowed          20.45
Married          20.23
Separated        19.87
Never Married    19.84
Divorced         19.61
Name: count, dtype: float64
++++==== End ====++++
```

The graph generated is:



Marital is a categorical variable featuring 5 unique values: Widowed, Married, Separated, Never Married, and Divorced. Of these values, Widowed is the most commonly occurring 20.45% in all values with a count of 2,045. The percentage distribution of all values is within roughly 1% of one another, each hovering around 20%.

Initial_Admin

The code I wrote to analyze the Initial_admin variable is:

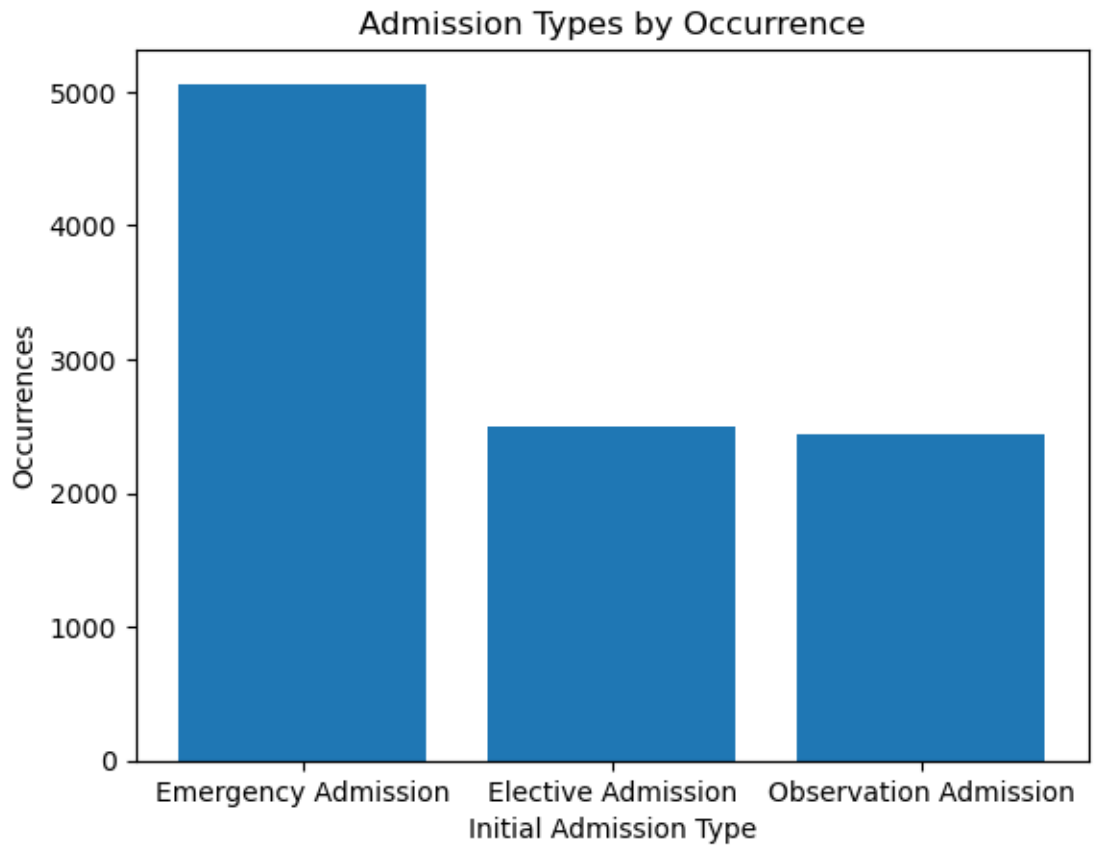
```
#Print relevant information about the variable
print('++++==== Begin ====++++\n')
print('++++==== Description of Initial_admin ====++++\n')
print(df['Initial_admin'].describe())
print('++++==== Unique Values in Initial_admin ====++++\n')
print(df['Initial_admin'].unique())
print('++++==== Counts of Values in Initial_admin ====++++\n')
print(df['Initial_admin'].value_counts())
print('++++==== Percentages of Values in Initial_admin ====++++\n')
print(np.round(df['Initial_admin'].value_counts() /
df['Initial_admin'].count() * 100, 2))
print('++++==== End ====++++\n')

#https://matplotlib.org/stable/gallery/lines_bars_and_markers/bar_colors.html
fig, ax = plt.subplots()
ax.bar(df['Initial_admin'].unique(), df['Initial_admin'].value_counts(),
label=df['Initial_admin'].unique)
ax.set_xlabel('Initial Admission Type')
ax.set_ylabel('Occurrences')
ax.set_title('Admission Types by Occurrence')
plt.show()
```

The results were:

```
++++==== Begin ====++++

++++==== Description of Initial_admin ====++++

count                   10000
unique                      3
top       Emergency Admission
freq                     5060
Name: Initial_admin, dtype: object
++++==== Unique Values in Initial_admin ====++++

['Emergency Admission' 'Elective Admission' 'Observation Admission']
++++==== Counts of Values in Initial_admin ====++++

Initial_admin
Emergency Admission      5060
Elective Admission       2504
Observation Admission    2436
Name: count, dtype: int64
++++==== Percentages of Values in Initial_admin ====++++

Initial_admin
```

```
Emergency Admission      50.60
Elective Admission       25.04
Observation Admission    24.36
Name: count, dtype: float64
++++==== End ========
```

The graph generated was:



Admission_type is a categorical variable represented by 3 unique values: Emergency Admission, Elective Admission, and Observation Admission. Emergency Admission is by far the most common, representing more than half of all values with a count of 5,060. The other two values fell within 1% of one another, with each hovering around 25%.

Income

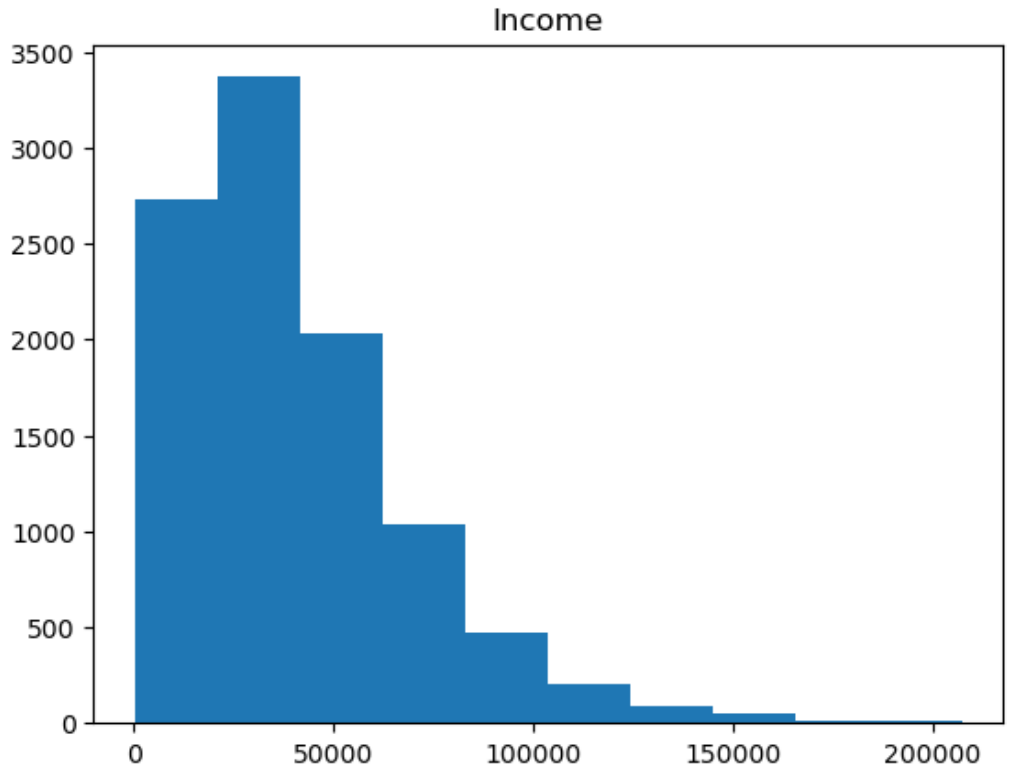
Below is the code I used to generate my analysis of Income:

```
print('++++==== Begin ====++++\n')
print('++++==== Description of Income ====++++\n')
print(df['Income'].describe())
print('++++==== Median of Income ====++++\n')
print(f'Median: {df['Income'].median()}')
print('++++==== End ====++++\n')

plt.hist(df['Income'])
plt.title('Income')
plt.show()
```

These are the results of the report:

```
++++==== Begin ====++++

++++==== Description of Income ====++++

count     10000.000000
mean      40490.495160
std       28521.153293
min         154.080000
25%       19598.775000
50%       33768.420000
75%       54296.402500
max      207249.100000
Name: Income, dtype: float64
++++==== Median of Income ====++++

Median: 33768.42
++++==== End ====++++
```

This is the graph generated:



Income is a continuous variable with a mean of 40,490.50 and a median of 33,768.42. The interquartile range is 34,697.62. Given that the median is much lower than the mean, I was able to infer that the distribution was skewed right. The histogram generated confirmed this inference visually.

TotalCharge

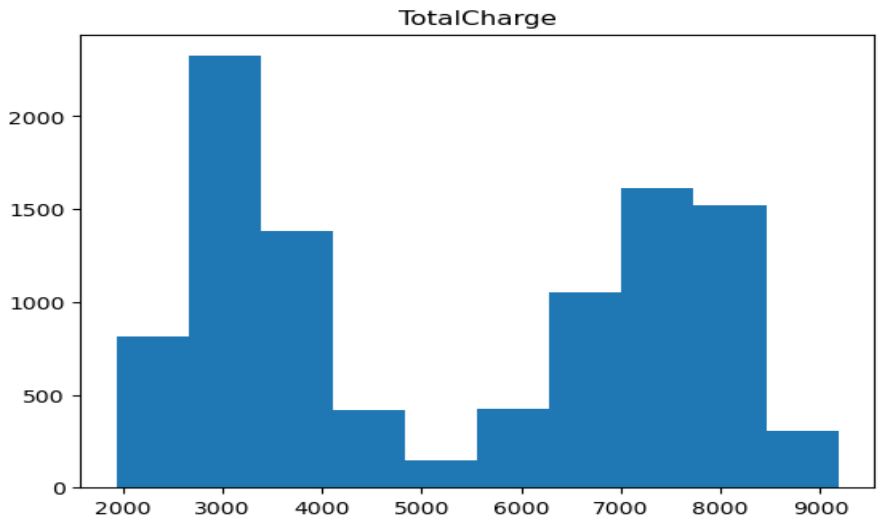
The code I used to generate the report is below:

```
print('++++==== Begin ====++++\n')
print('++++==== Description of TotalCharge ====++++\n')
print(df['TotalCharge'].describe())
print('++++==== Median of TotalCharge ====++++\n')
print(f'Median: {df['TotalCharge'].median()}')
print('++++==== End ====++++\n')

plt.hist(df['TotalCharge'])
plt.title('TotalCharge')
plt.show()
```

The results of the report:

```
++++==== Begin ====++++

++++==== Description of TotalCharge ====++++

count    10000.000000
mean      5312.172769
std       2180.393838
min       1938.312067
25%       3179.374015
50%       5213.952000
75%       7459.699750
max       9180.728000
Name: TotalCharge, dtype: float64
++++==== Median of TotalCharge ====++++

Median: 5213.952
++++==== End ====++++
```

The graph generated:



TotalCharge is a continuous variable with a mean of 5,312.17 and a median of 5,213.95. The interquartile range is 4,280.33. Given the closeness of the mean and median, I was anticipating a normal or uniform distribution. Visual analysis revealed two distinct 'peaks' in the data, signifying a bi-modal distribution.

D. Bivariate Statistics

Additional_charges vs. Gender

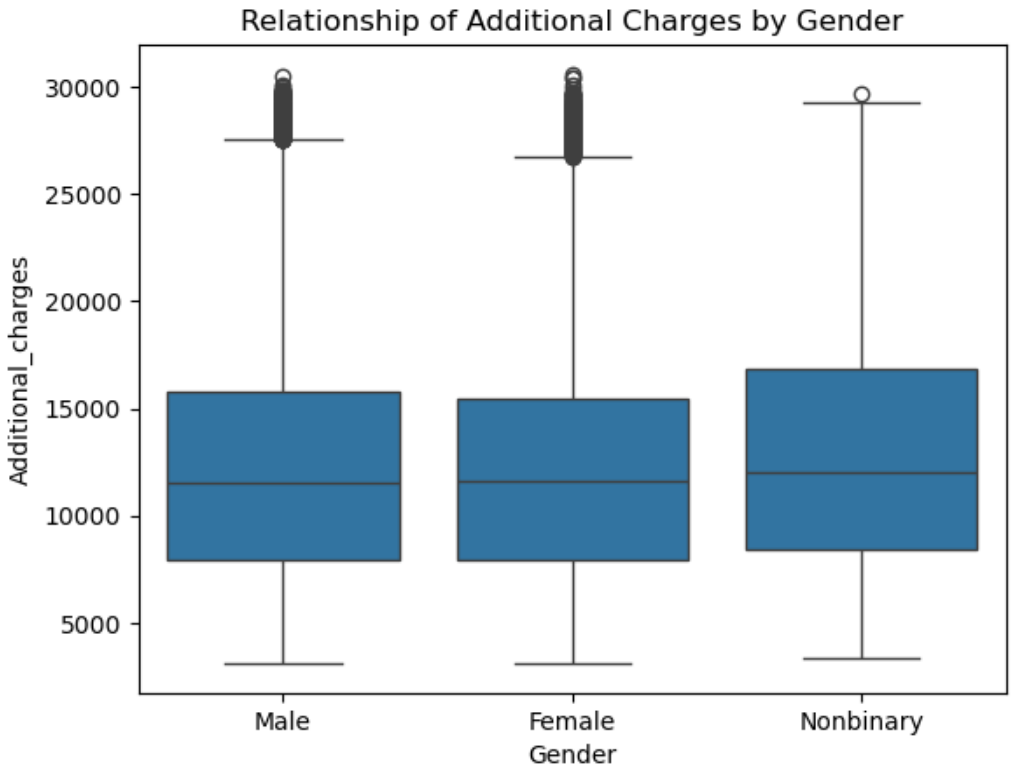
The code I wrote to generate the report:

```
sns.boxplot(data=df, x='Gender', y='Additional_charges')
plt.title('Relationship of Additional Charges by Gender')
plt.show()

#Group by Asthma and calculate stats for TotalCharge
gender = df.groupby('Gender').agg(
    Mean=('Additional_charges', 'mean'),
    Median=('Additional_charges', 'median'),
    Max=('Additional_charges', 'max'),
    Min=('Additional_charges', 'min'),
    IQR=('Additional_charges', calculate_iqr)
).reset_index()

print(gender)
```

The results of the report:

```
      Gender          Mean        Median       Max          Min          IQR
0     Female  12896.066069  11615.191865  30566.07  3139.049369  7511.782573
1       Male  12953.426237  11540.609295  30466.93  3125.703000  7819.419063
2  Nonbinary  13415.374018  11984.095000  29626.42  3369.832673  8413.311500
```

The generated graph:



There appears to be correlation between the values of Gender(Nonbinary) and the other values in the set. The mean, median, min, and IQR of Gender(Nonbinary) are all higher than those of Gender(Male) or Gender(Female). The only exception is max, which is high than both Gender(Male) and Gender(Female). This is likely due to the high occurrence of outliers in those categories, compared to the relatively few in Gender(Nonbinary).

TotalCharge vs. Asthma

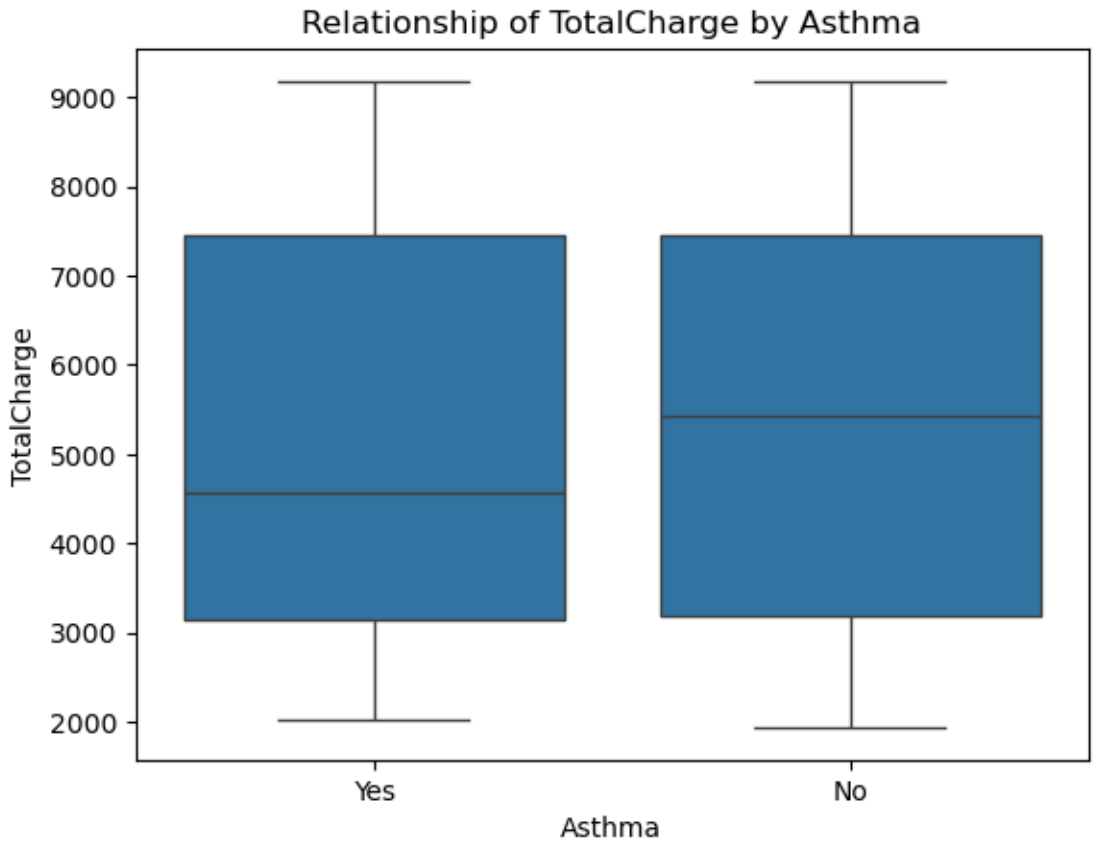
The code I wrote to generate the report:

```
sns.boxplot(data=df, x='Asthma', y='TotalCharge')
plt.title('Relationship of TotalCharge by Asthma')
plt.show()

#Group by Asthma and calculate stats for TotalCharge
asthma = df.groupby('Asthma').agg(
    Mean=('TotalCharge', 'mean'),
    Median=('TotalCharge', 'median'),
    Max=('TotalCharge', 'max'),
    Min=('TotalCharge', 'min'),
    IQR=('TotalCharge', calculate_iqr)
).reset_index()

print(asthma)
```

The results of the report:

	Asthma	Mean	Median	Max	Min	IQR
0	No	5332.051476	5428.338000	9180.728	1938.312067	4270.372278
1	Yes	5263.338351	4568.911956	9169.248	2022.650007	4313.228213

The generated graph:



The two categories, Asthma(Yes, No), appear to have an even distribution of values across the range of TotalCharge. The exception being the median of Asthma(No) seems to be significantly higher than Asthma(Yes). Given my earlier analysis of TotalCharge, I'm led to believe this is due to the bi-modal distribution of the variable itself.

E. Implications

E1. Results of the Hypothesis Test

In the analysis, I achieved this result:

```
Chi-square: 0.7133125620168337
P-value: 0.7000130641731285
The finding is not significant
```

Given the established alpha of 0.05, the findings were not determined to be significant enough to reject the null hypothesis. The result is found that the area that a hospital serves does not impact the rate of readmissions.

E2. Limitations of the analysis

The area collected within the survey data is based on unofficial census data. When and how this data was collected could present some potential for inaccuracy in results. The area classification is broad, and the parameters are not adequately described in the data dictionary. A larger sample size, more narrowly and explicitly defined categories, and data sources could improve both the quality of the data set and the analysis.

E3. Proposed Action

Given the acceptance of the null hypothesis, I would advise both the analytics team and the stakeholders to explore new avenues in determining the factors that impact the rates of readmission. Perhaps testing readmission rates against the raw population numbers, rather than the obliquely defined area categories.

F. Panopto Video

See link for Panopto Video

Code included as D207_Task1.ipynb

G. Code References

The following were used to generate charts I was not familiar with.

https://matplotlib.org/stable/gallery/pie_and_polar_charts/pie_features.html

https://matplotlib.org/stable/gallery/lines_bars_and_markers/bar_colors.html