Dataset Analysis and Visualization of Offences in the UK Between 2014 and 2015

MSC Data Science

Chidinma Darlington-Njoku

4/27/2021

Module Title: Data Analysis and Visualization Principles

University of Gloucestershire

INTRODUCTION

As a data science student who has recently moved to the UK, I am a female and a mother of two beautiful girls which has led me to having several conversations with my friends and neighbors on crimes and offences in the UK and read several reports on the internet also.

While exploring the information on the Offences in Britain as recorded by the Crown Prosecution Service. A monthly report on the criminal case outcomes by principal offence category and CPS Area were done and I found the information on Sexual Offence Convictions interesting and coincidentally found its correlation with other crimes such as Drug offences and Robbery, I decided to do some explorations with a focus on the years 2014 and 2015. The data is sourced from this URL: https://data.gov.uk/dataset/89d0aef9-e2f9-4d1a-b779-5a33707c5f2c/crown-prosecution-service-case-outcomes-by-principal-offence-category-data

This analysis seeks to answer a number of questions:

- What was trend of sexual offence conviction?
- Which of the months in 2014 or 2015 was the peak or floor period?
- Which county/counties had the highest/lowest sexual offence?
- Is there any relationship between Sexual offences and Drug offences or Robbery Convictions?
- How does their increase or decrease affect sexual convictions?

This report is broken into different stages: Data preprocessing, Data Cleaning, Explorative Data Analysis, Predictive Data Analysis, Hypothesis Testing, Clustering and Classification.

DATA PREPROCESSING

This analysis is being done using the data of the “principal offence category” for two years from January 2014 to December 2015 and requires 24 datasets representing each of the months for the 2 years in focus. One month is missing which is the month of November 2015 and that means I have to start out with the available 23 months datasets.

First things first, I imported the essential libraries and the datasets were loaded on the Jupyter notebook.

```
# importing the needed packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

```
# Loading the data by reading all files into pandas dataframes
Jan2014=pd.read_csv('principal_offence_category_january_2014.csv')
Feb2014=pd.read_csv('principal_offence_category_february_2014.csv')
Mar2014=pd.read_csv('principal_offence_category_march_2014.csv')
Apr2014=pd.read_csv('principal_offence_category_april_2014.csv')
May2014=pd.read_csv('principal_offence_category_may_2014.csv')
Jun2014=pd.read_csv('principal_offence_category_june_2014.csv')
Jul2014=pd.read_csv('principal_offence_category_july_2014.csv')
Aug2014=pd.read_csv('principal_offence_category_august_2014.csv')
Sept2014=pd.read_csv('principal_offence_category_september_2014.csv')
Oct2014=pd.read_csv('principal_offence_category_october_2014.csv')
Nov2014=pd.read_csv('principal_offence_category_november_2014.csv')
Dec2014=pd.read_csv('principal_offence_category_december_2014.csv')
Jan2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Feb2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Mar2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Apr2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
May2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Jun2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Jul2015= pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Datase
Aug2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
Sept2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Datase
Oct2015= pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Datase
Dec2015=pd.read_csv(r'C:\Users\user01\Desktop\Data Science Uniglos\Module 2\Assignment\Dataset for the assignment-20210303\Dataset
```

Figure 1

Using the first dataset which is for January 2014, I took a look to understand the data and draw inferences.

```
#Having a look to obtain infomation about one of the dataset
Jan2014.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 43 entries, 0 to 42
Data columns (total 51 columns):
 #   Column
Non-Null Count  Dtype
---  ------
--------------  -----
 0   Unnamed: 0
43 non-null     object
 1   Number of Homicide Convictions
43 non-null     int64
 2   Percentage of Homicide Convictions
43 non-null     object
 3   Number of Homicide Unsuccessful
43 non-null     int64
 4   Percentage of Homicide Unsuccessful
43 non-null     object
 5   Number of Offences Against The Person Convictions
43 non-null     object
```

Figure 2

```
#Checking for the number of observation
Jan2014.shape
```

```
(43, 51)
```

Figure 3

There are 43 observation rows of 42 locations within the UK and for the national, then 51 columns comprised of several offences convictions, offences unsuccessful, percentages of offences successful and percentages of offences unsuccessful.

I plotted a scatter matrix of all the variable in the dataset to see possible correlations, see figure 4.

```
#Plotting the scateer matric to see possible correlations
from pandas.plotting import scatter_matrix

scatter_matrix(Jan2014.loc[:,:],diagonal="kde")
axes = pd.plotting.scatter_matrix(Jan2014, alpha=0.2)
for ax in axes.flatten():
    ax.xaxis.label.set_rotation(90)
    ax.yaxis.label.set_rotation(0)
    ax.yaxis.label.set_ha('right')

#plt.tight_layout()
plt.gcf().subplots_adjust(wspace=0, hspace=0)
plt.show()
```

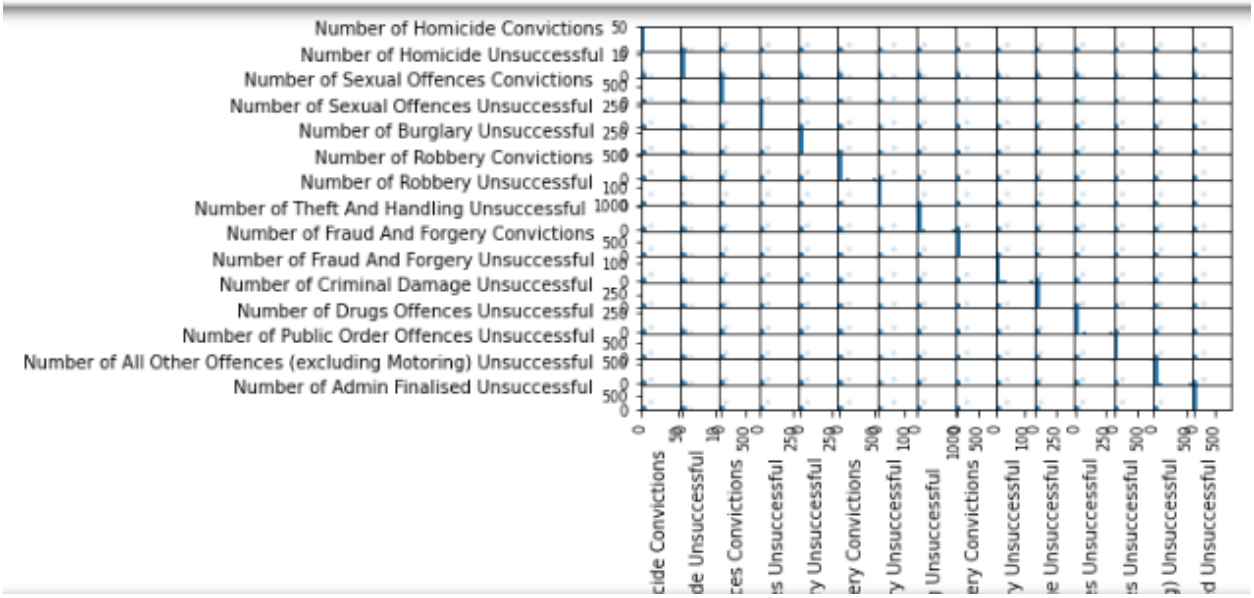


Figure 4

The data has only objects and integers (Figure 5) and this report, the columns of interest which have shown possible correlations are the “Number of Sexual Offences Convictions”, “Number of Drugs Offences Convictions” and the “Number of Robbery Convictions”; therefore, I dropped all columns not needed, see figure 6.

```
# checking to see how many records are in the dataset and checking for the missing data
merged2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 989 entries, 0 to 988
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Month_Year  989 non-null    object
 1   Location    989 non-null    object
 2   S_offences  989 non-null    object
 3   R_offences  946 non-null    float64
 4   D_offences  989 non-null    object
dtypes: float64(1), object(4)
```

Figure 5

```
#dropping columns for June 2014:
Jun2014.drop(Jun2014.columns[[1,10,34,2,3,4,5,6,7,8,15,16,18,19,20,21,22,23,24,25,26,27,28,29,30,31,
Jun2014.head(2)
```

	Unnamed: 0	Number of Sexual Offences Convictions	Number of Robbery Convictions	Number of Drugs Offences Convictions
0	National	778	512	4,563
1	Avon and Somerset	25	4	125

Figure 6

After dropping the 47 columns in each dataset, I merged the 23 datasets and got the output in fig7.

```
#merging the dataframes for 2014 and 2015 to create a new dataframe
merged2 = pd.concat([Jan2014,Feb2014,Mar2014,Apr2014,May2014,Jun2014,Jul2014,Aug2014,Sept2014,Oct2014,Nov2014,Dec2014,Jan
#creating a new CSV file
merged2.to_csv('merged2.csv')
merged2=pd.read_csv('merged2.csv')
merged2.head(2)
```

	Unnamed: 0	Unnamed: 1	Unnamed: 0.1	Number of Sexual Offences Convictions	Number of Robbery Convictions	Number of Drugs Offences Convictions
0	Jan2014	0	National	736	522.0	4,988
1	Jan2014	1	Avon and Somerset	35	8.0	148

Figure 7

Checked for the summary of the statistics of the newly created dataset of 988 rows and 5 columns (Figure 8), the standard deviation in the plot for the three numerical variables (127.15, 77.27 and 661.73) are far higher than their mean(39.34, 23.13 and 202.73) which shows that the how the data is spread out with a lot of anomalies.

```
#Finding the summary of the statistics
merged2.describe()
```

	S_offences	R_offences	D_offences
count	989.000000	946.000000	989.000000
mean	39.344793	23.135307	202.738119
std	127.154388	77.267683	661.731283
min	0.000000	0.000000	8.000000
25%	8.000000	3.000000	42.000000
50%	14.000000	6.000000	71.000000
75%	27.000000	11.750000	114.000000
max	1011.000000	650.000000	4988.000000

Figure 8

Checking for the Outliers

Before carrying out any analysis on the merged dataset, I decided to examine the it and found that there are lots of extreme values present which I need to treat appropriately. Figure 9 –figure 12 are the boxplots of the numerical columns; the Drug Offences Conviction, Sexual Offences Conviction and Robbery offences Conviction.

```
sns.boxplot(merged2['D_offences'],data=merged2)
```

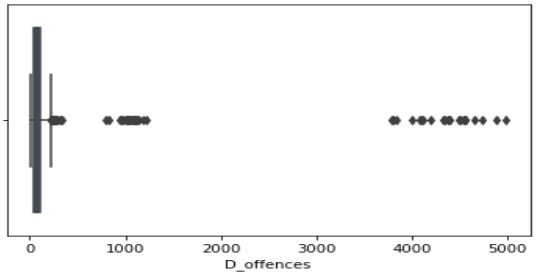


Figure 9

```
sns.boxplot(merged2['S_offences'],data=merged2)
```

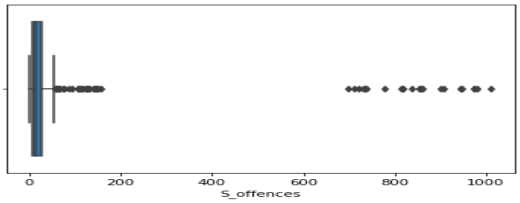


Figure 10

```
sns.boxplot(merged2['R_offences'],data=merged2)
```

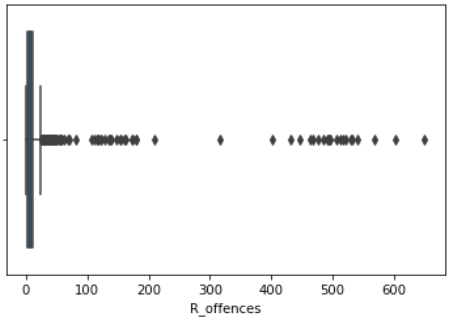


Figure 11

```
#comparing the outliers in the 3 variables
merged2.boxplot(['S_offences','D_offences','R_offences'])
plt.xlabel('Selected attributes from dataset')
plt.ylabel('Values')
plt.title('Boxplots of Sexual Offence and Drug Offence')
plt.show()
```

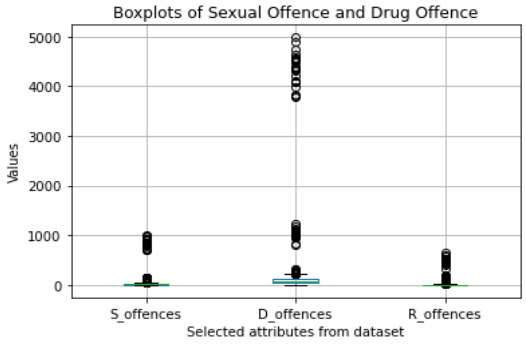


Figure 12

I used the box plot for outlier detection and for each of the three convictions to show the extreme values outside of the maximum mark. The bubbles (figure 10) indicate the unusual trend in the observation, for the sexual offence most of the data was concentrated maximally around 30 but the bubbles revealed that this column has extreme values which is up 1020 mark that are away from the maximum number of sexual convictions. For the Drug offences Conviction (Figure 9) showed that the data was concentrated around the 130 mark but there were value points from around the 135 mark to about 5000 that are beyond the maximum points while the robbery conviction (Figure 11) also shows that the maximum is around 20 but we have extreme numbers up to the 650 mark. The box plot has gone a long way to show the presence of outliers which would be treated to have a clean dataset.

The Histogram:

The histogram was plotted to further investigate the distribution of the data, fig 13-15 shows that the distributions are right-skewed as the values for the Sexual Conviction has more cases below 180 on the x-axis and a lot of the values with high magnitude are distinct and lying on the extreme right of the plot. For the Drug offences, the number of convictions is concentrated below 500 on the x-axis with a lot of value which are higher in magnitude being away from the region of concentration. For the robbery offences, most of the conviction cases are below 50 on the x-axis which shows that the dataset is really bias.

```
sns.histplot(x='R_offences',data=merged2, label="Robbery offences")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

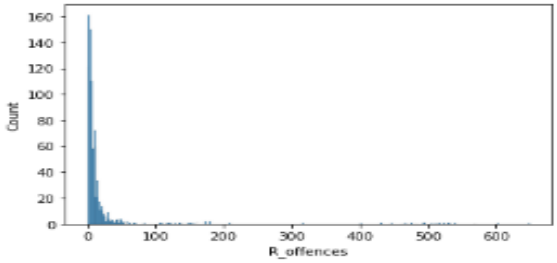


Figure 13

```
sns.histplot(x='D_offences',data=merged2, label="Drug Convictions")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

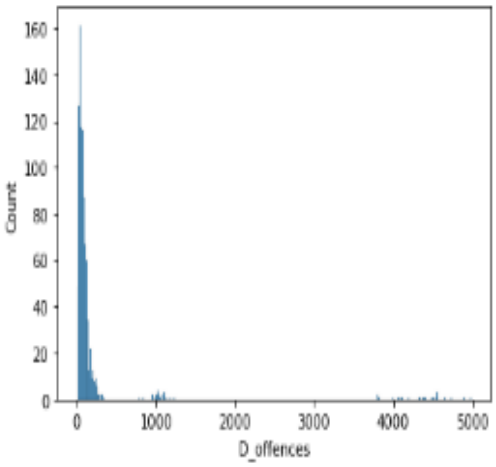


Figure 14

```
sns.histplot(x='S_offences',data=merged2, label="Sex Convictions")
plt.show
```

```
]: <function matplotlib.pyplot.show(close=None, block=None)>
```

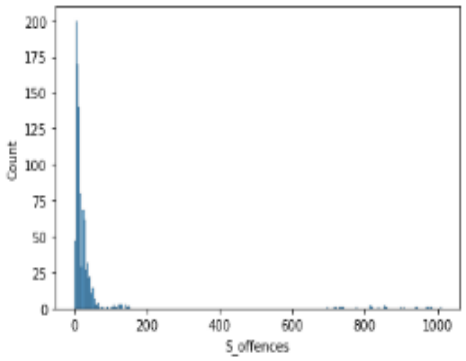


Figure 15

```
sns.histplot(data=merged2, x="S_offences", color="blue", label="S_offences",bins=5, kde=True)
sns.histplot(data=merged2, x="D_offences", color="red", label="D_offences",bins=5, kde=True)
sns.histplot(data=merged2, x="R_offences", color="green", label="R_offences",bins=5, kde=True)

plt.legend()
plt.show()
```

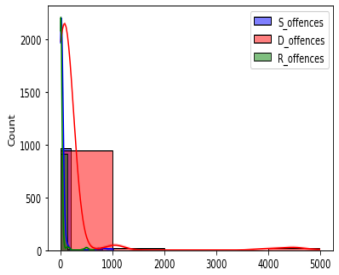


Figure 16

The Scattermatrix:

I used the Scatterplot (Figure 17 and Figure 18) to visualize the relationship between the dependent variable, sexual offences conviction and the other two variables which are the explanatory variables for this analysis. Sex conviction plotted against the Drug offence (Figure 17) is moving in the same direction at the points below 500 on the y-axis, showing that there is a relationship. Robbery against Sex offence is also showing a relationship as they are moving in the same direction around the 850 mark on the y-axis on the plot below. A further check for the relationship between sex and robbery offences shows they are moving in the same direction (Figure 18), for the drug and sex offence; the relationship is positive because as drug offence increases, sexual offence also increases. The pink and turquoise bubbles in the plot below represent the offences at the National and Metropolitan and City locations respectively, it is evident that they account for the high magnitude of the offences that are distinct from the vast majority of locations with much lower number of convictions and they cause the anomaly in the data set.

Figure 17

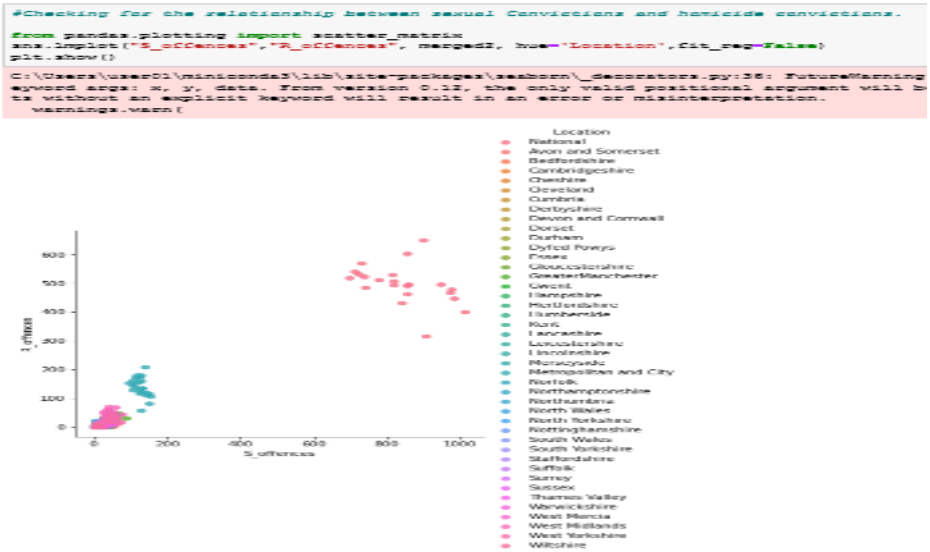


Figure 18

Data Cleaning:

The goal of this analysis is to determine if sexual offence convictions are related to drug and robbery convictions, hence the data has been resampled to 28 select counties to represent the UK's drug and sexual offence Convictions while excluding the data for the regions, Manchester and National.

```
#resizing the samples by filtering and creating a new data frame with the select counties.
County=['Bedfordshire','Cambridgeshire','Cheshire','Cumbria','Derbyshire','Dorset','Durham',
        'Dyfed Powys','Essex','Gloucestershire','Gwent','Hampshire','Hertfordshire','Kent',
        'Lancashire','Leicestershire','Lincolnshire','Merseyside','Norfolk','Northamptonshire',
        'Northumbria','Nottinghamshire','Staffordshire','Suffolk','Surrey','Sussex',
        'Warwickshire','Wiltshire']
merged2.loc[County]
county_merged2= merged2.loc[County]
county_merged2.to_csv('county_merged2.csv')
county_merged2=pd.read_csv('county_merged2.csv')
county_merged2.head()
```

	Location	Month_Year	S_offences	R_offences	D_offences
0	Bedfordshire	01/2014	2	16.0	31
1	Bedfordshire	02/2014	6	7.0	19

Figure 19

The dataset was resampled because it is would help to reduce error by focusing on mainly counties because the scatterplot revealed that the values for the National and the Metropolitan and city strayed away from the area of concentration of the data. Moreso, the boxplot showed that this locations were the cause of the the unusual trend in the observation thus making the histogram was rightly skewed because the value of these variables.

Figure 20 is the new boxplot after resampling and it shows that there is still a strong presence of the outliers; for the sexual offences, there are still bubbles between points 38 to 55 on the y-axis which are the extreme values above the maximum level, meaning that there is more than 14 sex convictions beyond the maximum number. For the drug offences also, the bubbles between point 160 and 380 are the extreme values above the maximum, also indicating that there is more than 160 drug convictions that are also beyond the maximum number of cases. The robbery offences shows a maximum point of about 20 but there are still values outside of the maximum boundary up to about 40. This box plot shows that there is still a need for the further cleaning of the dataset.

```
#Checking and comparing the outliers in the 3 variables for the resized dataframe
county_merged2.boxplot(['S_offences','D_offences','R_offences'])
plt.xlabel('Selected attributes from dataset')
plt.ylabel('Values')
plt.title('Boxplots of Sexual Offence and Drug Offence')
plt.show()
```

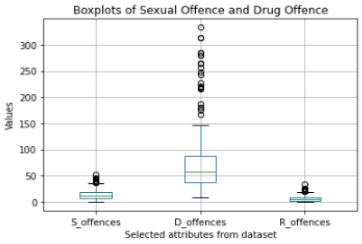


Figure 20

```
# Plotting a histogram for the resized sample to confirm if it is still right skewed

sns.histplot(data=county_merged2, x="S_offences", color="blue", label="S_offences",bin
sns.histplot(data=county_merged2, x="D_offences", color="red", label="D_offences",bins
sns.histplot(data=county_merged2, x="R_offences", color="green", label="R_offences",bi

plt.legend()
plt.show()
```

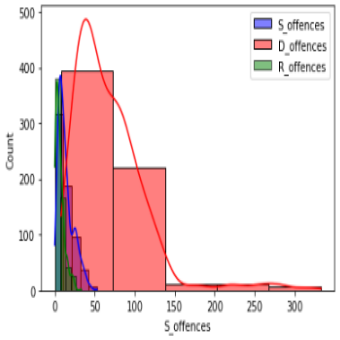


Figure 21

Outlier Identification and Treatment

```
#Getting the statistical overview of the data
county_merged2.describe()
```

	S_offences	R_offences	D_offences
count	644.000000	616.000000	644.000000
mean	13.737578	6.379870	68.704969
std	9.627805	5.536738	46.939025
min	0.000000	0.000000	8.000000
25%	6.000000	2.000000	37.000000
50%	11.000000	5.000000	58.000000
75%	18.000000	9.000000	88.000000
max	53.000000	33.000000	333.000000

Figure 22

```
county_merged2.mode(axis=0, numeric_only=True)
```

	S_offences	R_offences	D_offences
0	6	2.0	46

Figure 23

```
#replacing values greater than the upper quantile (75%) with the mode
county_merged2['S_offences']=np.where(county_merged2['S_offences']>18.0,6,county_merged2['S_offences'])
county_merged2['D_offences']=np.where(county_merged2['D_offences']>88.0,46,county_merged2['D_offences'])
county_merged2['R_offences']=np.where(county_merged2['R_offences']>9.0,2,county_merged2['R_offences'])
county_merged2.describe()
```

	S_offences	R_offences	D_offences
count	644.000000	616.000000	644.000000
mean	8.281056	3.558442	48.381988
std	4.008570	2.500698	18.323346
min	0.000000	0.000000	8.000000
25%	6.000000	2.000000	37.000000
50%	6.000000	3.000000	46.000000
75%	11.000000	5.000000	58.250000
max	18.000000	9.000000	88.000000

Figure 24

For the cleaning of the dataset, given the fact that each observation row in the dataset is the value of a county for offence convictions of interest for the years 2014 and 2015, substantial information would be lost if the if the outlier is treated by deleting the row with the extreme values . Figure 22 shows the maximum value, 75% quantile was used as the ceiling, so values above the 75% quantile were treated as the extreme values which needed to be replaced, replacing the extreme values with the mean was considered, but not used because the setback is that the mean is easily influenced by the outlier, tried the treatment with the median which was (11), (58) and (5) for the sexual, drug and robbery offence convictions respectively (Figure 22), the result was that it could not remove some of the extreme values for the Sexual and robbery Offence Convictions but the drug offence was cleaned. I tried the mode which was (6), (46) and (2.0) Figure 23, for the sexual, drug and robbery offence convictions respectively; it was adopted because the outliers in the three convictions were successfully treated. Now we have an absolutely normal looking boxplot below that is void of outliers and the histogram also, follows a normal distribution for each of the variables.

```
#Checking to confirm the absence of outliers
county_merged2.boxplot(['S_offences','D_offences','R_offences'])
plt.ylabel('Values')
plt.title('Boxplots of Sex, Drug and Robbery Conviction')
plt.show()
```

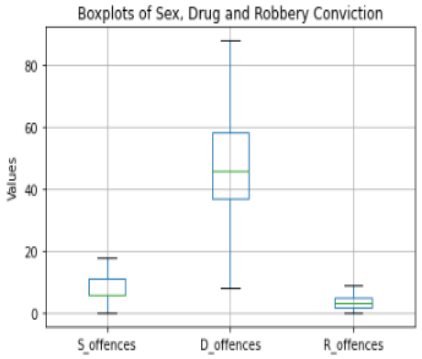


Figure 25

```
#Reconfirming the distribution of the histogram
sns.histplot(data=county_merged2, x="S_offences", color='
sns.histplot(data=county_merged2, x="D_offences", color='
sns.histplot(data=county_merged2, x="R_offences", color='

plt.legend()
plt.show()
```

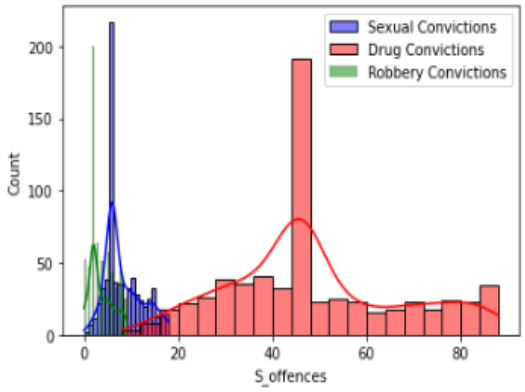


Figure 26

Filling up the missing values.

Now that the dataset is clean, a new empty dataframe was created for the missing month (November 2015) and the mean of the columns would be used to fill it up after it has been merged with the original dataframe (Figure 29). I chose to use the mean because it captures the midpoint of the long-term trend of the variable even though its weakness is it's susceptibility to outliers. Figure 27 and Figure 28 show the check for skewness before and after the cleaning of the data, figure 29 and figure 30 show the empty data frame created for the missing month, the merger of the new and original dataframe to be filled with the mean, then fig 31 shows the final check for missing values.

```
#Checking for the skewness before removing the outlier
print(county_merged2['S_offences'].skew())
print(county_merged2['D_offences'].skew())
print(county_merged2['R_offences'].skew())
```

```
1.0899891482350335
2.242685366497213
1.3793061538128875
```

Figure 27

```
#Rechecking for the skewness after removing the outlier
print(county_merged2['S_offences'].skew())
print(county_merged2['D_offences'].skew())
print(county_merged2['R_offences'].skew())
```

```
0.7440094733938574
0.4065181537817532
0.6356634998972694
```

Figure 28

```
#Creating a new Data frame for the missing month: November 2015
df = pd.DataFrame({'Month_Year':['11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015'
                                 '11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015'
                                 '11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015','11/2015'
                                 '11/2015'],
                   'Location':['Bedfordshire','Cambridgeshire','Cheshire','Cumbria','Derbyshire','Dorset','Durham','Dyfed
                               ,'Essex','Gloucestershire','Gwent','Hampshire','Hertfordshire','Kent','Lancashire','Leicest
                               'Lincolnshire','Merseyside','Norfolk','Northamptonshire','Northumbria','Nottinghamshire','S
                               'Suffolk','Surrey','Sussex','Warwickshire','Wiltshire'],
                   'S_offences':['','','','','','','','','','','','','','','','','','','','','','','','','','',''],
                   'R_offences':['','','','','','','','','','','','','','','','','','','','','','','','','','',''],
                   'D_offences':['','','','','','','','','','','','','','','','','','','','','','','','','','','']})
df.head()
```

	Month_Year	Location	S_offences	R_offences	D_offences
0	11/2015	Bedfordshire			
1	11/2015	Cambridgeshire			
2	11/2015	Cheshire			
3	11/2015	Cumbria			
4	11/2015	Derbyshire			

Figure 29

```
#merging the new and old dataframes
frames = [county_merged2,df]
NC_merged1 = pd.concat(frames)
NC_merged1.to_csv('NC_merged1.csv')
NC_merged1=pd.read_csv('NC_merged1.csv')
NC_merged1.tail()
```

	Unnamed: 0	Location	Month_Year	S_offences	R_offences	D_offences
667	23	Suffolk	11/2015	NaN	NaN	NaN
668	24	Surrey	11/2015	NaN	NaN	NaN
669	25	Sussex	11/2015	NaN	NaN	NaN
670	26	Warwickshire	11/2015	NaN	NaN	NaN
671	27	Wiltshire	11/2015	NaN	NaN	NaN

Figure 30

```
#Using the mean to fill up the missing values.
NC_merged1.fillna(NC_merged1.mean(),inplace = True)
NC_merged1.isnull().sum()
```

```
Unnamed: 0    0
Location      0
Month_Year    0
S_offences    0
R_offences    0
D_offences    0
dtype: int64
```

Figure 31

Explorative Data Analysis

Now that I have a clean dataset of 4 columns and 672 rows, I want to explore the data with nice looking graphs using seaborn and matplotlib for my analysis.

```
#checking for the yearly mean of each variable
NC_merged1.groupby(pd.Grouper(freq='Y')).mean()
```

	S_offences	R_offences	D_offences
2014-12-31	8.440476	3.571429	48.363095
2015-12-31	8.121636	3.545455	48.400880

Figure 32

```
#Checking for the yearly median of each variable
NC_merged1.groupby(pd.Grouper(freq='Y')).median()
```

	S_offences	R_offences	D_offences
2014-12-31	7.0	3.000000	46.0
2015-12-31	6.0	3.558442	46.0

Figure 33

I would like to start by analyzing the mean of the variables by grouping the dataset into the two years in focus; 2014 and 2015 (Figure 32 & figure 33). The three convictions had different patterns for both years; sex conviction has reduced average by (0.31884) while the median reduced from (7) to (6). For robbery the average reduced by (0.025) but on the other hand, the median increased by (0.558); the drug offence average increased by (0.038) with a static median of (46). In comparism, Sexual conviction had a highest change of the three variables.

```
#Checking for the info of the new clean datatset
NC_merged1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 672 entries, 2014-01-01 to 2015-11-01
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Location    672 non-null    object
 1   S_offences  672 non-null    float64
 2   R_offences  672 non-null    float64
 3   D_offences  672 non-null    float64
dtypes: float64(3), object(1)
memory usage: 26.2+ KB
```

Figure 34

```
NC_merged1.describe()
```

	S_offences	R_offences	D_offences
count	672.000000	672.000000	672.000000
mean	8.281056	3.558442	48.381988
std	3.924042	2.394074	17.936967
min	0.000000	0.000000	8.000000
25%	6.000000	2.000000	37.000000
50%	7.000000	3.000000	46.000000
75%	11.000000	5.000000	58.000000
max	18.000000	9.000000	88.000000

Figure 35

The focus of this explorative analysis would be on the dependent variable (sexual Convictions), the 28 counties selected have shown a variety of trends as the mean value of sexual offence for the aggregate of the 28 counties for the time frame is 8.28 , the standard deviation is now 3.92 is reduced and much lower than the mean and the median is 7 showing that for this conviction, about 50% of the counties have cases lower than 7 while for the other half, the number of their cases is greater than 7. See figure 35.

24 Months Moving Average for Sexual Convictions

```
#Checking for the monthly frequency of the Sex Convictions for all locations
NC_merged1.S_offences.resample('M').mean().plot()
```

```
<AxesSubplot:>
```

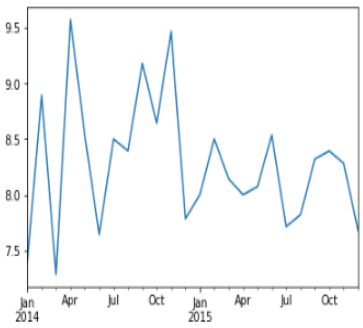


Figure 36

```
#Plotting the average of the sexual offences for each of the locations
NC_merged1.groupby('Location')['S_offences'].mean().sort_values(ascendi
plt.ylabel('Averages')
plt.xlabel('Location')
plt.title('Average of Sexual Offence Conviction for UK 28 Counties')
plt.show()
```

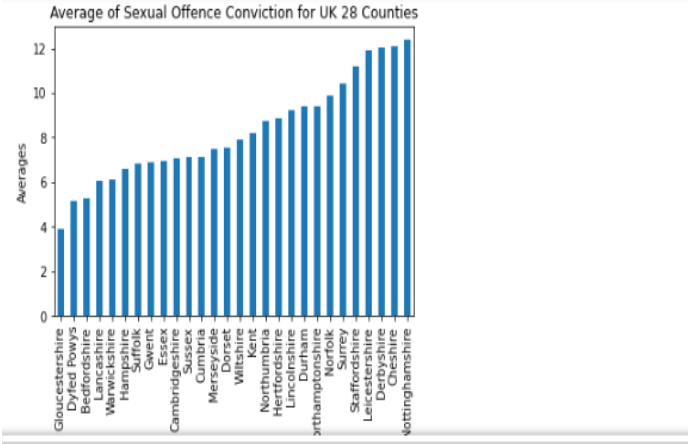


Figure 37

```
#Checking for the total number of sexual offences Conviction
NC_merged1['S_offences'].sum()
```

```
5564.869565217391
```

Figure 38

The monthly average number of sexual offences for the 24 months in focus showed that the lowest number of the convictions 6.6 was in March 2014 while the highest of 7.2 was for April 2014 and the other months spanned in between. Out of the 28 counties, 19 which is 67% of the sample size have the number successful sexual convictions that is greater or equal to 7 which is the group's median value. The average for the sex

offence convictions by location depicts Nottinghamshire as the county with the strongest trend having an average of 12.39(figure 37), the highest average is more than the aggregate mean by 4.11 and Gloucestershire has 3.89 which is the least average in the group, much lower than the aggregate mean by 4.39. Nottinghamshire has the highest sum of 297, while Gloucestershire has the lowest of 93 for the time period, out of (5565) which the total sum of the Sexual offence convictions for the 28 counties.

Finding the Minimum and Maximum Values:

```
#Plotting for the counties with the hightest number of sexual offence Convictions
NC_merged1.groupby('Location')['S_offences'].max().sort_values(ascending=True).plo
plt.ylabel('Sex Conviction')
plt.xlabel('Location')
plt.title('Counties with the highest number of the sexual offence Convictions')
plt.show()
```

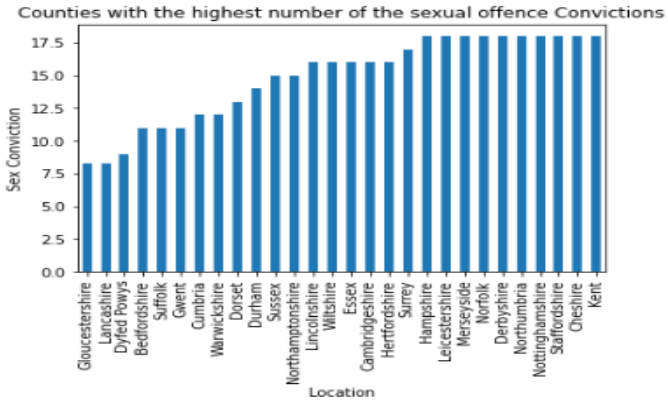


Figure 39

```
#Plotting for the counties with the lowest number of sexual offence Convictions
NC_merged1.groupby('Location')['S_offences'].min().sort_values(ascending=True).plot(kind='bar')
plt.ylabel('Sex Conviction')
plt.xlabel('Location')
plt.title('Counties with the lowest number of the sexual offence Convictions')
plt.show()
```

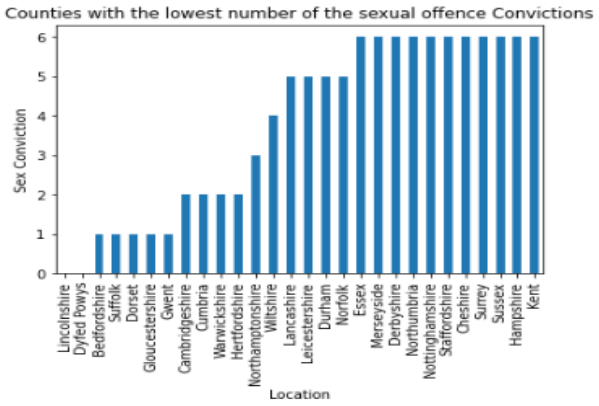


Figure 40

Figure 39 shows that of the selected 28 counties, our top 10 are Kent, Cheshire, Staffordshire, Notthinghamshire, Northumbria,Derbyshire, Norfolk, Merseyside,Leicestershire and Hampshire which have a

common value (18), are the counties where sexual offences conviction was at its peak. For find the the lowest, there are 2 counties (Lincomlshire and Dyfed Powys) with (0) number of sexual offences conviction while (Bedfordshire, suffolk, Dorset, Glouscestershire, Gwent) have only one (1) case of this offence, figure 40.

I decided do further investigation on Nottinghamshire, the county with the highest sum (297) and average(12.38) for the sexual Offence Conviction and compared its performance for the other variables- Drug and Robbery offences convictions.

As seen in Figure 44, it has an average of 47.43 which is a little lower than the group's average (48.38) and above the group's median threshold (46.0) meaning that this county could be dangerous as it has a relatively high number of both sexual offences convictions and drug offences conviction.

To further check for the robbery offences convictions in this county, Nottinghamshire has an average of (4.046537) which is above (3.56) the average for the group and above the group's median threshold (3.0), see figure 43.

This county has a summation of 297 sexual offences Conviction, 97 Robbery Offences Conviction and 1138 drug offences conviction, figure 44. In checking for the trend of these convictions in this county, figure 46 shows that the month of January 2014 had an extremely high number of 78 drug offences then maintained a consistent value of 46 from February 2015 through to October 2015, the increases to 48 in November 2015. The reason for the behaviour of the drug offences conviction shown is not provided for in this analysis, however, it has a total of 1138 of the group aggregate of 32513 which is 3% of the total number of drug offences conviction for the observation period.

Figure 46 shows that for the Sexual and robbery offences convictions, the Nottinghamshire County was not static like the drug offence conviction but had a number of fluctuations through the period of observation. It accounts for the total sum of 297 out of 5565 of the group's aggregate of sexual offences conviction that is 5% of the total number of this conviction for the period of observation. For robbery, Nottinghamshire had a summation of 97 out of (5565) the group aggregate for Robbery offences Conviction through the time period in focus, showing that this county accounts for 1.7% of the group's total sum for the robbery offences conviction, see figure 44.

```
#Plotting for the counties with the hightest number of Drug offence Convictions
NC_merged1.groupby('Location')['D_offences'].mean().sort_values(ascending=True).plot(kind='bar')
plt.ylabel('Average')
plt.xlabel('Location')
plt.title('Average of the Drug offence Convictions for Uk 28 Counties')
plt.show()
```

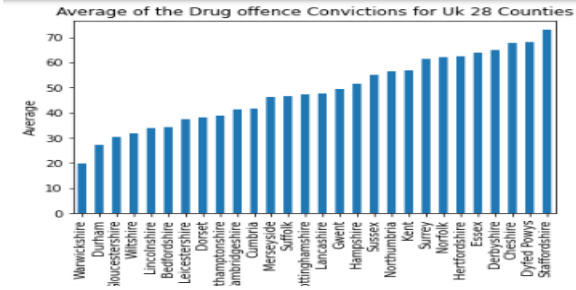


Figure 41

```
NC_merged1['D_offences'].median()
```

```
46.0
```

```
NC_merged1['D_offences'].mean()
```

```
48.38198757763975
```

Figure 42

```
NC_merged1['R_offences'].mean()
```

```
3.558441558441558
```

```
NC_merged1['R_offences'].median()
```

```
3.0
```

Figure 43

```
#Checking for the average of the county of Nottinghamshire for the convictions
NC_merged1.groupby(['Location']).get_group('Nottinghamshire').mean()
```

```
S_offences    12.386711
R_offences     4.046537
D_offences    47.432583
dtype: float64
```

```
#Checking Nottinghamshire for total number of crimes.
NC_merged1.groupby(['Location']).get_group('Nottinghamshire').sum()
```

```
Location      NottinghamshireNottinghamshireNottinghamshireN...
S_offences                                          297.281056
R_offences                                           97.116883
D_offences                                         1138.381988
dtype: object
```

Figure 44

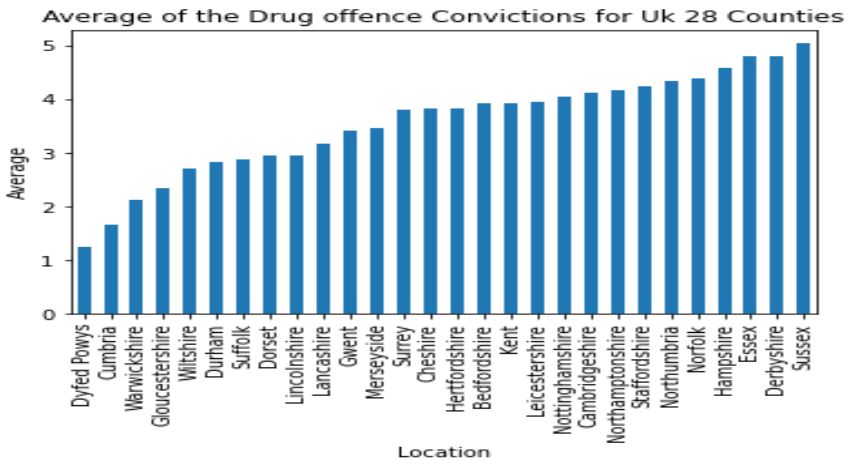


Figure 45

```
#Checking the monthly count of the convictions in Nottinghamshire:
NC_merged1.groupby(['Location']).get_group('Nottinghamshire').plot(kind='bar')
```

```
<AxesSubplot:>
```

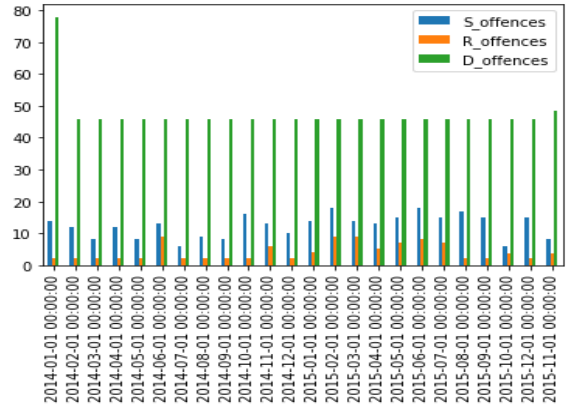


Figure 46

Correlation and Covariance:

The heat map, scattermatrics and the .corr () function were used to prove the correlation between sexual offences -the dependent variable and the explanatory variables- Drug and Robbery Conviction. Along the diagonal of the heatmap and the scatterplot (figure 47 and Figure 48), there is a strong indication of every variable strongly correlating against itself. The positive correlation between sex and robbery convictions is denoted by a strong red colour with a value of 0.126445 while the higher correlation existing between sex and drug convictions is denoted by a strong orange colour with a value of 0.201422. This means that an increase in drug offences or robbery offences could bring about an increase in sex offences.

I also used the covariance to check for the behavior of the relationship between the dependent and the explanatory variables. The covariance between sex and robbery convictions is 1.187876 while the covariance between sex and drug conviction is 14.177139 (figure 49). The results are both positive showing that the dependent variable moves in the direction of both explanatory variables.

```
#Plotting the heatmap to further prove the strong correlation between sexual offences and Drug offences
sns.heatmap(NC_merged1.corr(), cmap="RdYlGn", linewidths=0.30)
plt.show()
```

Figure 47

```
NC_merged1.corr()
```

	S_offences	R_offences	D_offences
S_offences	1.000000	0.126445	0.201422
R_offences	0.126445	1.000000	0.163489
D_offences	0.201422	0.163489	1.000000

Figure 48

```
#Checking for the covarriance between sex conviction and the other variables
NC_merged1.cov()
```

	S_offences	R_offences	D_offences
S_offences	15.398106	1.187876	14.177139
R_offences	1.187876	5.731589	7.020613
D_offences	14.177139	7.020613	321.734771

Figure 49

PREDICTIVE DATA ANALYSIS

Multiple Linear Regression: I am using this to know if the drug and robbery convictions contribute to the number of sex conviction, the result is β0=5.754981691968988 and β1=0.04062817 for the drug coefficient and β2=0.1574854 for the robbery coefficient. This means that the positively influence sex convictions and sex conviction would be at 5.754981691968988 without them, see figure 50.

```
#Multiple linear regression
from sklearn.linear_model import LinearRegression

data = pd.read_csv('pres_merged1.csv', index_col=0)

feature_cols = ['D_offences','R_offences']

X = data[feature_cols]
Y = data.S_offences

lr_model = LinearRegression()

lr_model.fit(X, Y)

print(lr_model.intercept_)

print(lr_model.coef_)
```

```
5.754981691968988
[0.04062817 0.1574854 ]
```

Figure 50

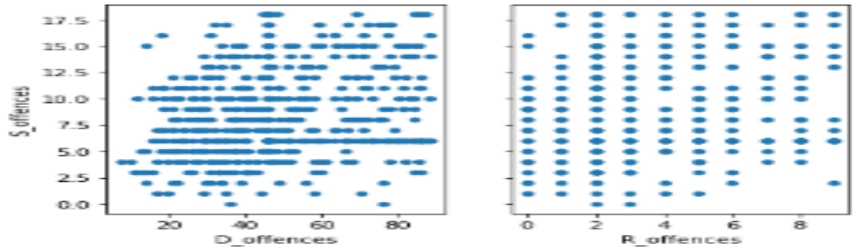
Model Testing and Prediction using Linear Regression:

I want to see how Drug and Robbery Convictions drive sex convictions; by generating the model, I created a linear relationship to know how much effect the variables have on sex convictions, the result is β0 (the intercept) =6.149119 and β1 (the slope) =0.044065 for the drug convictions. The β1 is positive, meaning that every rise in drug convictions by 6.149119 unit can bring about a rise in sex convictions at the rate of 0.044065. For the robbery offences conviction, also shows the result β0:7.538484 and β1: 0.207251. Since the β1 is positive, it means that a rise in robbery convictions could also cause a rise in sex convictions at the rate of 0.207251 with effect not as strong as the drug conviction, see figure 51.

To test the model, I created a new value by predicting an increase of 300 in drug offences conviction to see what the Sex Conviction would be; it resulted in sexual offences being at 21.57.

Evaluating the relationship between Sex and Drug convictions, I want to determine if it is coincidental or not by creating a hypothesis:

H0: β1=0
Ha: β1≠ 0
α = 5% or 0.05 - with 95% interval
the result is a Pvalue: 2.543147e-07 which is much lower than 0.05 and proves that there is really a relationship between Sex conviction and Drug Conviction.




```
Drug B0 and B1:
Intercept      6.149119
D_offences     0.044065
dtype: float64
Robbery B0 and B1:
Intercept      7.538484
R_offences     0.207251
dtype: float64
   D_offences
0         350
Drug test:  0    21.571756
dtype: float64
   R_offences
0         100
Rob test:  0    28.263561
dtype: float64
D_offences PValue:
Intercept      2.565354e-39
D_offences     2.543147e-07
dtype: float64
R_offences PValue:
Intercept      2.123149e-106
R_offences      1.318920e-03
dtype: float64
```

Figure 51

HYPOTHESIS TESTING:

Sex cases is always of interest to me because I'm female and a mother. After visualizing the dataset, testing by predicting the model with multi-linear regression showing that sexual offences conviction is obviously more strongly correlated with drug (0.201422) than robbery offences(0.126445), I decided to prove that the relationship between Sexual Offences Convictions which is our dependent variable and the explanatory Variables –Drug offences Conviction using the following hypothesis:

Hypothesis 1: (H0) and (Ha) are as follows:
H0: There is no relationship between Sexual Offences Conviction and Drug Offences Conviction (i.e., β1 is zero)
Ha: Sexual Offences Conviction is driven by Drug Offences Conviction (i.e., β1 is not zero)

Hypothesis 2:
The explorative data analysis showed the median of the drug cases was 46.0 for both 2014 and 2015, while the calculated mean was 48.363095 for 2014 and 48.400880 for 2015, this makes me believe that the mean for drug offence could possibly maintain the value(48.38) of the previous year 2014 just like the media and 2015 should not be 48.400880 which is the calculated mean. I would be testing this hypothesis with the Z-test.

Hypothesis 2: (H0) and (Ha) are as follows as follows:
H0= 48.38
H0≠48.38

Testing Hypothesis 1 with the Pearson's Correlation Coefficient:
In order to evaluate the relationship between the variables, I used the Pearson's correlation coefficient because my data has numerical variables and this is best suited for evaluating relationships between the numerical variables.

H0= β1= 0
Ha= β1≠ 0,
α = 5% or 0.05 - with 95% interval, I am making this claim
Decision rule: we reject HO if the p-value of the result is less than 0.05

```
#Using the Pearson's Coefficient to test for the strenght of
from scipy.stats import pearsonr
data1 = NC_merged1['S_offences']
data2 = NC_merged1['D_offences']
stat, p = pearsonr(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
    print('We are accepting null hypothesis')
else:
    print('We are rejecting null hypothesis')
```

```
stat=0.201, p=0.000
We are rejecting null hypothesis
```

Figure 52

The result (Figure 52) showed the p-value: 0.00 and a correlation of 0.201, drawing an inference from the p-value for Drug Offence conviction with 95% confidence level interval, I can say again that there is a relationship between the Sexual offence conviction and the Drug offence conviction ($p<0.05$), hence I can reject the null hypothesis and accept the alternative hypothesis with the proof of the Pearson correlation test.

Testing Hypothesis 2 with Z-Test:

I am hypothesizing that mean value of drug conviction is 48.363 which is the average for 2014, given that the median is static.

H0= 48.400880
Ha ≠ 48.400880
α = 5% or 0.05 - with 95% interval, I am making this claim
Decision rule: we reject HO if the p-value of the result is less than 0.05
The result is p-value of 0.978107731887984 (Figure 53), the null hypothesis would be accepted because the p-value is higher than 0.05 at 95% confidence interval, proving that the mean is of the drug convictions is the calculated mean and not the hypothesized mean.

```
#Using the Z-Test to test hypothesis 2
from scipy.stats import ttest_1samp
from statsmodels.stats import weightstats as stests
import numpy as np

#reading file
pred = np.genfromtxt("pred.csv", delimiter=',')

#calculating the drug offence mean
D_offences_mean = np.mean(pred[:,3])
print(D_offences_mean)

#hypothesis testing
#drug mean is 50(Ha)
#Formula Z-test

#ztest
ztest,pval = stests.ztest(pred[:,3], value=48.363)
print("p-values",pval)

#95% confidence
if pval<0.05:
    print("Z-test => we are rejecting null hypothesis")
else:
    print("Z-test => we are accepting null hypothesis")
```

```
48.38198757763975
p-values 0.978107731887984
Z-test => we are accepting null hypothesis
```

Figure 53

Clustering

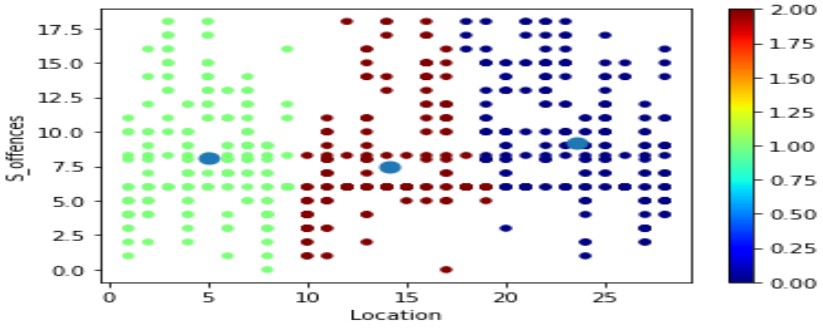


Figure 54

The clustering technique I used is the k-means and I chose to cluster based on sexual offences convictions and location. I initialized at random with a maximum iteration of 100 for all columns. The visualization (fig54) is showing that there are 3 groups that are different from each other: Cluster 1 - blue, cluster 2 – green and cluster3 - red. Location is on the X-axis and we have 25 of them; the green category are the first 10 locations which have low to medium count of sexual convictions, the red category are the locations between 10 and 20 which have a mix of the medium and maximum number of sexual convictions then the third category which is the blue shows the peak zone that have highest count of sexual convictions.

I chose to use the k means for the clustering technique because it is the right fit for my model as it is simple as it is efficient and works with only the variables needed for the clustering, even though it has a downside of being susceptible to outliers because it works with the mean and it assumes everything is spherical while my model isn't spherical.

Classification:

The classification technique I am using is the KNN. My k-neighbor is 3, I am predicting sex conviction and the result is predicted class: 9.333, which means that sex conviction would be 9.33 at location2 (Cambridgeshire) given that robbery is 7 and drug is 31(Figure 55)

```
#Impoting the necessary librarires
import pandas as pd
import numpy as np
from sklearn import metrics
from sklearn.neighbors import KNeighborsClassifier

from sklearn.neighbors import KNeighborsRegressor
from sklearn.neighbors import KNeighborsRegressor
neigh = KNeighborsRegressor(n_neighbors=3)

pred = pd.read_csv('pred.csv')
Y = df.iloc[:,1]
X = df.iloc[:,0:3]

neigh.fit(X, Y)
KNeighborsRegressor(...)
print(neigh.predict([[2,10,8]]))
predicted_class = neigh.predict(np.reshape(prediction_test,[1,-1]))
print("predicted class: ", predicted_class)
```

```
[9.33333333]
predicted class:  [9.33333333]
```

Figure 55

Analysis tools and techniques:

Tools: Python programming language and Jupyter notebook is the tool used for my analysis, it is user friendly and flexible while Atom, Spyder and Pycharm are not as commonly used as python.

Techniques: For the analysis of my dataset, it went through the following stages:

- **Data preprocessing and Cleaning**: - loading and arranging my data set, identified and treated anomalies, removed improper formats to make it error free. Each months dataset had 51 columns and 43 rows, 47 columns were dropped the irrelevant columns, converted the data types from object to integers. For the outlier detection, I trimmed the dataset by removing the aggregate rows of national and region, then removed the mega city of greater machester, I used the flooring and capping technique for the counties, while 75% quantile was my ceiling to pick out higher values which were removed.
  For outlier replacement, I tried the median but still had some extreme values, didn't use the mean because of it is susceptible to outliers, so I worked with the mode and which cleaned my dataset optimally.
  For the missing month, I considered filling the equivalent month from the previous year which is November 2014, but I did not trust its accuracy because it factor in trends and I decided to use the mean to fill up the missing values since dataset is now clean and no outlier to influence it.
- **Visualization**: I used the barplot to show trends, scatterplot to show the relationship between my variables, histogram to show how my data is distributed, heatmap to show the correlation.
- **Data Visualization:** I used the box plot to detect the anomalies in my dataset, the histogram to check for the distribution of my variables, scatterplot ,scattermatrics and heatmap showed the relationship between my variables , the bar charts helped with showing the trends in my data.
- **Hypothesis testing:** I have two hypotheses, firstly, used Pearson's Coefficient correlation to test for the relationship between Sexual conviction and Drug conviction because it is the best for numerical data. Secondly, used the Z-Test to test the mean of the Drug conviction variable because it's best for greater-than 30 sample sized data.
- **Regression:** I used the multiple linear regressions for prediction and testing my model since I'm working with multiple variables, it is best for evaluating the relationship between numerical variables.
- **Clustering:** Kmeans was used to create groups within my dataset because it simple despite its weakness of assumption of spherical clusters.
- **Classification**: I used the KNN because it makes no assumption and works well with any kind of data even though it uses a lot of memory. Didn't use the Naïve Bayes because work a large amount of data, the decision tree because it's predictable.

Visualization Tools:

For the visualization of my analysis, I used the following tools:

- Matplotlib: As a beginner, I found it to be and flexible and sometimes complex as it supports barchart, histogram, scatterplot
- Seaborn: I find it unique as it produces beautifully colourful visualization with just a single line of code, swarm plot, catplot, etc. It integrates well with pandas, works with and based on matplotlib.

REFERENCES

Kanoki , 2001: *Data Science, Pandas, Python Dataframe Groupby data and time,* Kanoki Blog[Online]Viewed 1 April 2021, <https://kanoki.org/2020/05/26/dataframe-groupby-date-and-time/>.

Rashida, N. S.,2021: *Exploratory Data Analysis, visualization and Prediction model in Python using pandas*, Matplotlib,seaborn and scikit_learn Libraries in Python.(Towards Data science blog), viewed 3 April,2021, <https://towardsdatascience.com/exploratory-data-analysis-visualization-and-prediction-model-in-python-241b954e1731>.

Chris, M., 2015: *Overview of Python Visualization tools*. (KD nuggets Blog), viewed 3 April, 2021, <https://www.kdnuggets.com/2015/11/overview-python-visualization-tools.html>.

Prasad, P., 2018: *What is exploratory Data Analysis?* (Towards Data science blog), viewed 3 April, 2021, < https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15>.

Alamin, M.M., 2021: *Identifying, Cleaning and Replacing outliers Titanic dataset* (Analytics Vidhya Blog), viewed 10 April 2021, <https://medium.com/analytics-vidhya/identifying-cleaning-and-replacing-outliers-titanic-dataset-20182a062893>.

Ajitesh, K., 2020: *Python- Replacing missing values with Mean, Median & Mode* (Vitalflux blog), viewed 15 March 2021, < https://vitalflux.com/pandas-impute-missing-values-mean-median-mode/>.

Jason, B., 2020: *How to calculate correlation between variable in Python* (Machine learning Mastery Blog), viewed 10 April, 2021, <https://machinelearningmastery.com/how-to-use-correlation-to-understand-the-relationship-between-variables/>.

Scott, R., 2018: *K-Nearest Neighbours Algorithm in Python and Scikit-Learn* (Stack Abuse Blog), viewed 15 April 2021, < https://stackabuse.com/k-nearest-neighbors-algorithm-in-python-and-scikit-learn/>.

Jason, B., 2019: 17 Statistical Hypothesis Tests in Python (Cheat Sheet) (Machine Learning Mastery Blog), viewed 15 April 2021, <https://machinelearningmastery.com/statistical-hypothesis-tests-in-python-cheat-sheet/>.

Scikit-learn Developers, 2007-2020: Sklearn.neighbours KNeighborsRegressor (Scikitlearn Blog), viewed 17 April 2021, < https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html>.