HPD Analysis (No MAP)-Copy1

September 1, 2024

1 Houston Police Department Elementary Data Analysis

1.0.1 Noah Choate

1.1 Data Information

1.1.1 About the Data

For the purpose of this assignment, the dataset that will be analyzed is the National Incident-Based Reporting System's (NIBRS) dataset in the Houston area. The NIBRS is responsible for capturing characteristics of incident based data, such as the date and time, type of incident, and location. The dataset provided details all criminal incidents in the Houston area in 2023.

The dataset consists of 16 columns and 249737 rows. In the analysis, I will begin by exploring the trends of when incidents occur by months / seasons of the year, and hours of the day in attempts to see if there are any trends. Following that by detailing the frequency of when the incidents happen in the top zip-codes, along with the top descriptions (Theft from motor vehicle, etc.). Concluding the analysis will be a spatial analysis utilizing k-clustering that will aim to find common areas of where incidents occur, beside a concluding statement to finish the report.

1.1.2 Loading the Libraries / Data

To begin, I first loaded libraries that will aid me in conducting my data analysis. Pandas and Numpy in order to help manipulate the dataframe to conduct a fair analysis as well as matplotlib and seaborn to produce visuals. I also had imported a Linear Regression function in order to help my analysis on the frequency of crimes throughout the year. In order to do spatial analysis, I had imported KMeans for clustering and folium to visualize the clusters on a map.

[10]:
```
# Import Necessary Libraries
import pandas as pd
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set(style = "darkgrid")
from sklearn.linear_model import LinearRegression
from sklearn.cluster import KMeans
import folium
```

```python
# Read dataframe in
df = pd.read_excel(r"C:
↪\Users\noahc\DSC3334-AlgorithmsandDataStructures\data\NIBRSPublicView2023.
↪xlsx")

# Preview of df
print(df.head())4
```

```
   Incident RMSOccurrenceDate  RMSOccurrenceHour NIBRSClass  \
0      2423        2023-01-01                  0        90D
1      2823        2023-01-01                  0        520
2      2823        2023-01-01                  0        35A
3      4623        2023-01-01                  0        220
4      6723        2023-01-01                  0        13B

                    NIBRSDescription  OffenseCount   Beat  \
0      Driving under the influence             1   5F20
1            Weapon law violations             1  10H50
2        Drug, narcotic violations             1  10H50
3  Burglary, Breaking and Entering             1   5F30
4                   Simple assault             1  14D20

                                  Premise StreetNo StreetName StreetType Suffix  \
0          Highway, Road, Street, Alley     1760   CAMPBELL         RD    NaN
1  Residence, Home (Includes Apartment)     3100      ANITA         ST    NaN
2  Residence, Home (Includes Apartment)     3100      ANITA         ST    NaN
3                  Bank, Savings & Loan     5253  HOLLISTER         ST    NaN
4  Residence, Home (Includes Apartment)     4119   BARBERRY         DR    NaN

      City ZIPCode  MapLongitude  MapLatitude
0  HOUSTON   77080    -95.516391    29.802506
1  HOUSTON   77004    -95.357674    29.732490
2  HOUSTON   77004    -95.357674    29.732490
3  HOUSTON   77040    -95.506306    29.842607
4  HOUSTON   77051    -95.367319    29.651655
```

1.1.3 Preparing / Cleaning the Data

Due to the nature of the raw dataframe given by the NIBRS, it will be necessary to perform some manipulation on the dataframe in order to improve the accuracy of my data analysis. One of the top priorities when cleaning data is to remove all of the missing data in the dataframe. Because our data is mainly categorical and independent events, it does not make sense to speculate/ predict data. One glaring issue is The "Suffix" column in the dataframe. This column was used to clarify what road an incident occured, (Ex: Shepherd Drive "N") however since many roads do not utilize a suffix, removing this column is necessary. If we do not remove this column prior to removing all missing data, over 75% of our data will be removed, and therefore the report will not be as accurate as it could be. We will then locate where missing values are, and drop them. Additionally it is vital to ensure all of our datatypes in our dataframe are the appropriate type in order to perform analysis with them.

[2]:
```python
# Remove Suffix col, will remove too much data if kept upon cleaning. Redundant
df.drop(columns = ['Suffix'], inplace = True)

# Confirm the correct data types
print(df.dtypes)

# locate this missing values
missing_values = df.isnull().sum()
print(missing_values)

# Because the dataset is categorical, remove all rows with missing values
df_dropNA = df.dropna()
print(df)
```

```
Incident                       int64
RMSOccurrenceDate     datetime64[ns]
RMSOccurrenceHour              int64
NIBRSClass                    object
NIBRSDescription              object
OffenseCount                   int64
Beat                          object
Premise                       object
StreetNo                      object
StreetName                    object
StreetType                    object
City                          object
ZIPCode                       object
MapLongitude                 float64
MapLatitude                  float64
dtype: object
Incident                  0
RMSOccurrenceDate         0
RMSOccurrenceHour         0
NIBRSClass                0
NIBRSDescription          0
OffenseCount              0
Beat                    193
Premise                   0
StreetNo                860
StreetName                0
StreetType            18276
City                      1
ZIPCode                3393
MapLongitude           3190
MapLatitude            3190
dtype: int64
```

```
        Incident RMSOccurrenceDate  RMSOccurrenceHour NIBRSClass  \
0           2423        2023-01-01                  0        90D
1           2823        2023-01-01                  0        520
2           2823        2023-01-01                  0        35A
3           4623        2023-01-01                  0        220
4           6723        2023-01-01                  0        13B
…              …               …                …          …
249733   1601024        2023-12-31                 23        240
249734 185927523        2023-12-31                 23        13A
249735 185940723        2023-12-31                 23        35A
249736 185940723        2023-12-31                 23        90D
249737 185962423        2023-12-31                 23        290

                     NIBRSDescription  OffenseCount   Beat  \
0          Driving under the influence             1   5F20
1                Weapon law violations             1  10H50
2            Drug, narcotic violations             1  10H50
3        Burglary, Breaking and Entering           1   5F30
4                       Simple assault             1  14D20
…                                   …            …     …
249733             Motor vehicle theft             0   5F20
249734              Aggravated Assault             1  11H50
249735       Drug, narcotic violations             1   1A10
249736     Driving under the influence             1   1A10
249737   Destruction, damage, vandalism            1  16E10

                                 Premise StreetNo         StreetName  \
0              Highway, Road, Street, Alley     1760          CAMPBELL
1       Residence, Home (Includes Apartment)     3100             ANITA
2       Residence, Home (Includes Apartment)     3100             ANITA
3                       Bank, Savings & Loan     5253         HOLLISTER
4       Residence, Home (Includes Apartment)     4119          BARBERRY
…                                         …        …                 …
249733                        Other, Unknown     7930        BLANKENSHIP
249734  Residence, Home (Includes Apartment)     4800          ALLENDALE
249735          Highway, Road, Street, Alley     1800              PEASE
249736          Highway, Road, Street, Alley     1800              PEASE
249737  Residence, Home (Includes Apartment)    13351  CITY PARK CENTRAL

       StreetType     City ZIPCode  MapLongitude  MapLatitude
0              RD  HOUSTON   77080    -95.516391    29.802506
1              ST  HOUSTON   77004    -95.357674    29.732490
2              ST  HOUSTON   77004    -95.357674    29.732490
3              ST  HOUSTON   77040    -95.506306    29.842607
4              DR  HOUSTON   77051    -95.367319    29.651655
…            …        …       …             …           …
249733         DR  HOUSTON   77055    -95.488949    29.819928
249734         RD  HOUSTON   77017    -95.250621    29.681868
```

```
249735          ST   HOUSTON   77003    -95.362190    29.747570
249736          ST   HOUSTON   77003    -95.362190    29.747570
249737          LN   HOUSTON   77047    -95.384408    29.614650

[249738 rows x 15 columns]
```

To ensure the analysis being performed is an accurate representation of the total dataframe, it is vital that we minimize the percentage of values dropped. A percentage of values dropped being below 10% in a dataframe containing roughly 250,000 entries will still provide an accurate analysis.

[3]:
```
# Clarify how much data was removed
before_length = len(df)
print(f'Length before dropping NA values: {len(df)}')

after_length = len(df_dropNA)
print(f'Length after dropping NA values: {len(df_dropNA)}')

percent_decrease = (abs(before_length - after_length) / before_length) * 100
print(f'Percentage of values dropped: {percent_decrease}%')
```

```
Length before dropping NA values: 249738
Length after dropping NA values: 226988
Percentage of values dropped: 9.109546805051693%
```

Furthermore, it would also be noteworthy to understand how many unique instances there are for the columns in the dataframe I will be doing an analysis on. I chose ZIP-code over City as there are more zip-codes listed, and therefore will be a more accurate display of percentages given by the analysis.

[4]:
```
# Zipcode
unique_zips = df['ZIPCode'].nunique()
print(f'Number of Unique Zip-Codes Listed: {unique_zips}')

# Description
unique_descriptions = df['NIBRSDescription'].nunique()
print(f'Number of Unique Descriptions Listed: {unique_descriptions}')

# NIBRSClass
unique_class = df['NIBRSClass'].nunique()
print(f'Number of Unique NIBRS Classes Listed: {unique_class}')

# Date
unique_date = df['RMSOccurrenceDate'].nunique()
print(f'Number of Unique Dates Listed: {unique_date}')

# Lat/Long
unique_lat = df['MapLatitude'].nunique()
print(f'Number of Unique Latitudes / Longitudes Listed: {unique_lat}')
```

```
Number of Unique Zip-Codes Listed: 212
```

```
Number of Unique Cities Listed: 141
Number of Unique Descriptions Listed: 61
Number of Unique Beats Listed: 129
Number of Unique NIBRS Classes Listed: 61
Number of Unique Dates Listed: 365
Number of Unique Latitudes / Longitudes Listed: 69580
```

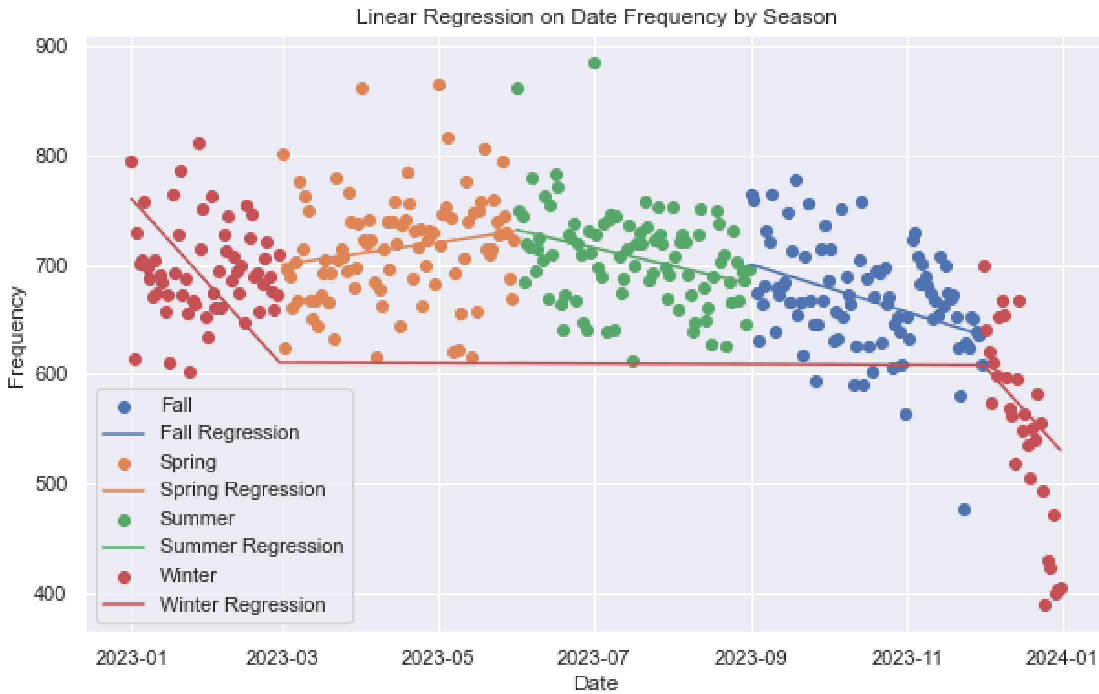
1.2 Descriptive Statistics and Visualization

To begin the analysis we will look a time-series analysis regarding seasonal and trends of crime, followed by a look into the distribution of incidents by the hour of the day. Following the time analysis, will be an examination of the frequency of the top NIBRS Descriptions of incidents, as well as the ZIPCodes. Finalizing the report will be a spatial analysis to identify hotspots in the Houston area of crime using k-clustering.

1.2.1 Linear Regression Time-Series Analysis

[5]:
```python
# Prepare Data for Linear Regression on Dates / Seasons

# Season Work
df['Month'] = df['RMSOccurrenceDate'].dt.month

seasons = {
    'Winter': [12, 1, 2],
    'Spring': [3, 4, 5],
    'Summer': [6, 7, 8],
    'Fall': [9, 10, 11]
}

# Map months to seasons
df['Season'] = df['Month'].apply(lambda x: next(season for season, months in␣
 ↪seasons.items() if x in months))

season_frequency = df.groupby(['Season', 'RMSOccurrenceDate']).size().
 ↪reset_index(name='frequency')

# Plot linear regression for each season
plt.figure(figsize=(10, 6))
for season, data in season_frequency.groupby('Season'):
    X = data.index.values.reshape(-1, 1)
    y = data['frequency']
    model = LinearRegression()
    model.fit(X, y)
    y_pred = model.predict(X)
    plt.scatter(data['RMSOccurrenceDate'], data['frequency'], label=season)
    plt.plot(data['RMSOccurrenceDate'], y_pred, label=f'{season} Regression')

plt.xlabel('Date')
```

```
plt.ylabel('Frequency')
plt.title('Linear Regression on Date Frequency by Season')
plt.legend()
plt.show()
```

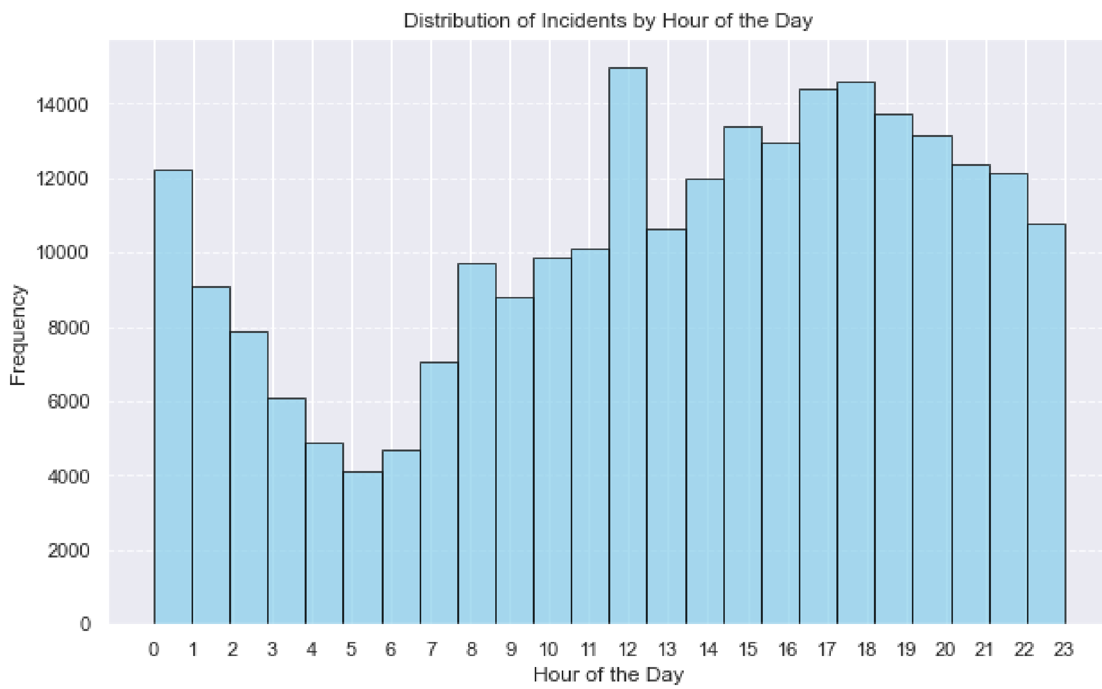


The plot above is displaying the seasonal trends of the frequency of dates incidents occur in. According to the data, it is common for the Houston area to begin the year slowly creeping from around the 700 incidents per day to roughly 725 incidents during the peak around May. Slowly towards the end of the year we can expect incidents to drop in October from roughly 700 towards 650, only to drop 625 in early December. Oddly, in late December there is a sharp decrease that finishes the year around 550 incidents. If we speculate this graph to repeat, then we can expect a sharp rise during the new year. Utilizing previous years' data would help this prediction, however for the purpose of this elementary data analysis that will not be necessary. This information is important as it could be a reasoning to re-tool / re-manage our police departments. For example, during December when incidents reach a low, maybe shuffle around the police department to have less people patrolling during December. Conversely, slighly increase the number of police on patrol around May, when incidents are at a yearly-high.

1.2.2 Distribution Time-Series Analysis

[6]:
```
# Prepare data for visualization of distribution of incdents per hour of the day
df['RMSOccurrenceHour'] = pd.to_datetime(df['RMSOccurrenceHour'], format='%H')

# Extract hour
```

```
df['Hour'] = df['RMSOccurrenceHour'].dt.hour

# Histogram to show the distribution of incidents by hour
plt.figure(figsize=(10, 6))
plt.hist(df['Hour'], bins=24, color='skyblue', edgecolor='black', alpha=0.7)
plt.title('Distribution of Incidents by Hour of the Day')
plt.xlabel('Hour of the Day')
plt.ylabel('Frequency')
plt.xticks(range(24))  # Set ticks for each hour
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```



The histogram above display the distribution of incidents and the hour of the day they occur. From the data, we can observe that most incidents occur noon. Throughout the day, you can typically expect most incidents are likely to occur around 4-6 PM. This information can be useful as it can show when our police department can be utilizied most efficiently on the street. Because of the analysis above, it is proven that obviously it would be more beneficial to have the majority of our officers patrolling from noon to 6 PM rather than midnight to 6 AM.

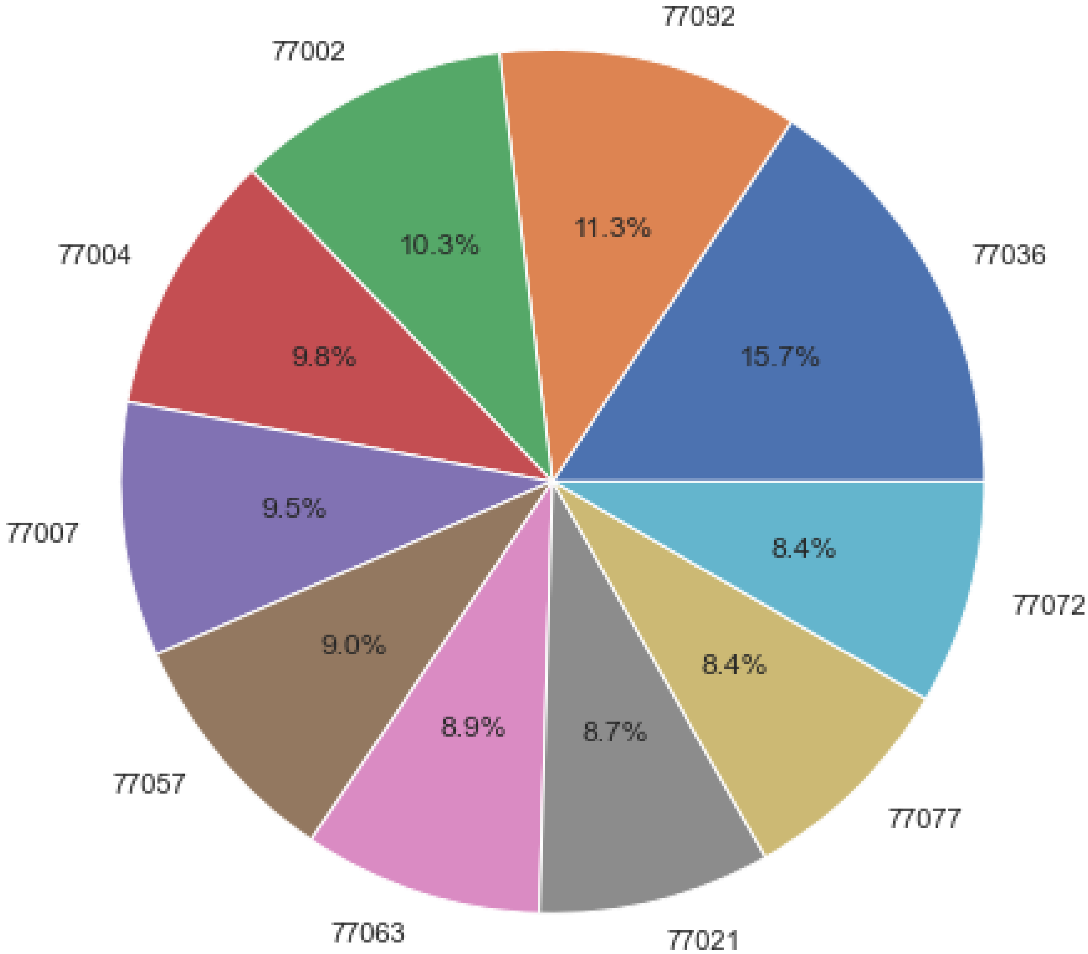
1.2.3 Data Frequency Analysis

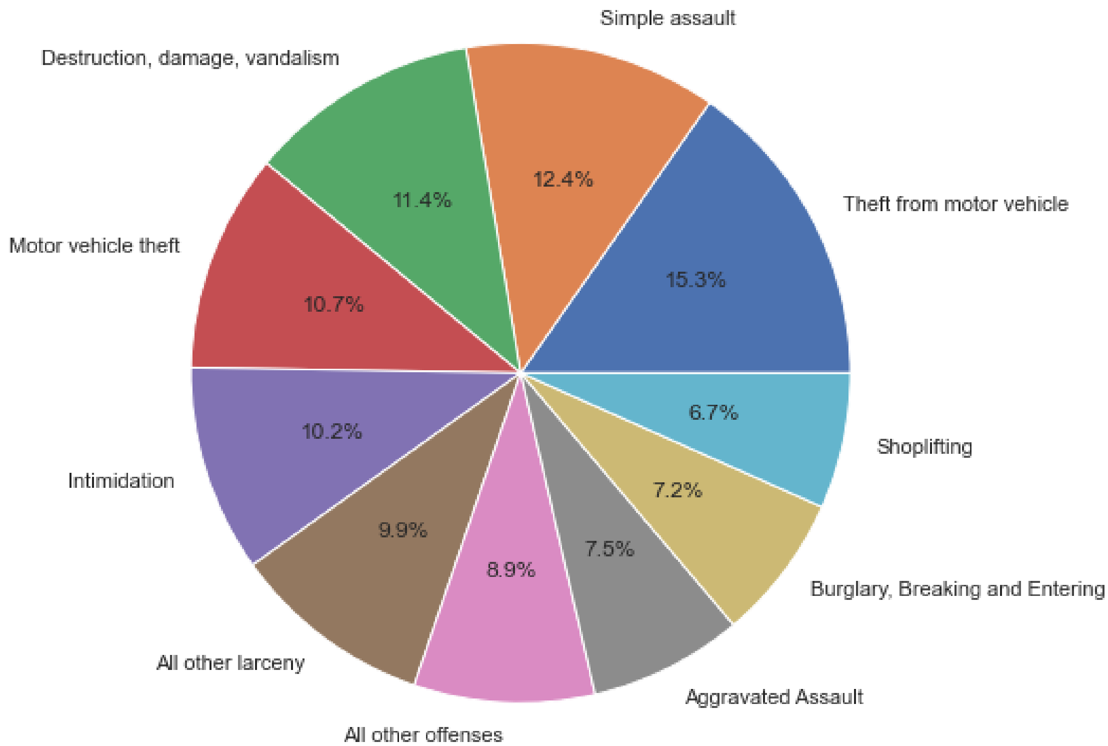
Pie Chart

[7]:
```
# Top 10 most common ZIP-Code occurrances
common_zip = df['ZIPCode'].value_counts()
ten_common_zips = common_zip.head(10)
```

```python
plt.figure(figsize=(8,8))
plt.pie(ten_common_zips, labels = ten_common_zips.index, autopct = '%1.1f%%')

# Top 10 most common Offenses
common_offenses = df['NIBRSDescription'].value_counts()
ten_common_offenses = common_offenses.head(10)
plt.figure(figsize=(8,8))
plt.pie(ten_common_offenses, labels = ten_common_offenses.index, autopct = '%1.
 ↪1f%%')
```

[7]:
```
([<matplotlib.patches.Wedge at 0x27f4fe2c850>,
  <matplotlib.patches.Wedge at 0x27f4fe2cfa0>,
  <matplotlib.patches.Wedge at 0x27f50214700>,
  <matplotlib.patches.Wedge at 0x27f50214e20>,
  <matplotlib.patches.Wedge at 0x27f50218580>,
  <matplotlib.patches.Wedge at 0x27f50218ca0>,
  <matplotlib.patches.Wedge at 0x27f50212400>,
  <matplotlib.patches.Wedge at 0x27f50212b20>,
  <matplotlib.patches.Wedge at 0x27f50205280>,
  <matplotlib.patches.Wedge at 0x27f502059a0>],
 [Text(0.9755569626079338, 0.5082210274154176, 'Theft from motor vehicle'),
  Text(0.2414622532201113, 1.0731709930248148, 'Simple assault'),
  Text(-0.5507505908462679, 0.9521941958878384, 'Destruction, damage,
vandalism'),
  Text(-1.0320964793133083, 0.38049554187805396, 'Motor vehicle theft'),
  Text(-1.0500947576358097, -0.3275683134642152, 'Intimidation'),
  Text(-0.6562450771013095, -0.8828037147522072, 'All other larceny'),
  Text(-0.05572681264412226, -1.0985875123778384, 'All other offenses'),
  Text(0.49323525991956685, -0.9832186828839643, 'Aggravated Assault'),
  Text(0.8799881555257706, -0.6600157923370867, 'Burglary, Breaking and
Entering'),
  Text(1.076022474741318, -0.2284198630845616, 'Shoplifting')],
 [Text(0.5321219796043275, 0.2772114694993187, '15.3%'),
  Text(0.13170668357460616, 0.5853659961953535, '12.4%'),
  Text(-0.3004094131888733, 0.5193786523024573, '11.4%'),
  Text(-0.5629617159890772, 0.20754302284257486, '10.7%'),
  Text(-0.5727789587104416, -0.17867362552593558, '10.2%'),
  Text(-0.3579518602370779, -0.4815292989557493, '9.9%'),
  Text(-0.03039644326043032, -0.5992295522060935, '8.9%'),
  Text(0.26903741450158186, -0.5363010997548896, '7.5%'),
  Text(0.479993539377693, -0.3600086140020473, '7.2%'),
  Text(0.5869213498589007, -0.12459265259157905, '6.7%')])
```


77092
77002
10.3%
11.3%
77004
77036
9.8%
15.7%
77007
9.5%
8.4%
77072
9.0%
8.4%
8.9%
8.7%
77057
77077
77063
77021



Table

[8]:

```
# Calculate frequency and percentage of ZIP codes
zip_count = df['ZIPCode'].value_counts()
total_times = zip_count.sum()
zip_percentage = (zip_count / total_times) * 100

# Create df for ZIP codes
zip_table = pd.DataFrame({'ZIP-Code': zip_count.index,
                          'Frequency': zip_count.values,
                          'Percentage': zip_percentage.values})

print("ZIP Code Table:")
print(zip_table)

# Calculate frequency and percentage of NIBRS descriptions
desc_count = df['NIBRSDescription'].value_counts()
total_descs = desc_count.sum()
desc_percentage = (desc_count / total_descs) * 100

# Create df for NIBRS descriptions
```

```python
desc_table = pd.DataFrame({'Description Type': desc_count.index,
                           'Frequency': desc_count.values,
                           'Percentage': desc_percentage.values})

print("\nNIBRS Description Table:")
print(desc_table)
```

```
ZIP Code Table:
       ZIP-Code  Frequency  Percentage
0         77036       9127    3.704967
1         77092       6576    2.669427
2         77002       5976    2.425866
3         77004       5718    2.321135
4         77007       5532    2.245631
..          …          …          …
207       75050          1    0.000406
208  77045-0000          1    0.000406
209       91406          1    0.000406
210       77001          1    0.000406
211  77060-1668          1    0.000406

[212 rows x 3 columns]

NIBRS Description Table:
                         Description Type  Frequency  Percentage
0                 Theft from motor vehicle      28234   11.305448
1                           Simple assault      22865    9.155595
2             Destruction, damage, vandalism      20973    8.398001
3                      Motor vehicle theft      19779    7.919900
4                             Intimidation      18761    7.512273
..                                      …          …          …
56                                  Incest          5    0.002002
57  Human Trafficking/Involuntary Servitude          2    0.000801
58                    Negligent manslaughter          2    0.000801
59                                 Runaway          2    0.000801
60                               Peeping tom          1    0.000400

[61 rows x 3 columns]
```

With categorical data, it is important to note the frequency and relative distribution of each element. In this scenario, it is important to analyze where these incidents are likely to occur and what incidents occur. To begin, a pie-chart visualizing the the zip-codes where most incidents occur, followed by a in-depth table. One thing to note is that the pie-chart display percentages in proportion to the top-10 zip-codes where incidents occur, while the table displays percentages in regards to all zip-codes. Because all of these zip-codes are in inner-Houston and Harris County, it was important to narrow the location search down to zip-code (for now). This is important as again, knowing what zipcodes most incidents occur in gives justification to toughen up patrol out in those areas in hopes to prevent more incidents from occuring. For example, we can see that not only the most- but double the incidents occur in 77036 than in 77077. With this knowledge, there is concrete evidence that more patrolling needs to be done in those areas than say 77001 where only 1 incident had occured last year.

Our second pie-chart and table combination details what NIBRS description type each incident had recieved. Again, the pie-chart shows the percentages relative to the top-10, while the tables give the percentages regarding all description types. This visualization is important as it shows what category Houston needs help with the most in regards to illegal incidents. From the table, we can tell that roughly 11% of all incidents are Theft from a motor vehicle. With this knowledge, we can train new police officers how to deal with the most common situation they will likely face on the job, as well as measure to prevent these from occuring. If we know theft from a motor vehicle is common, then we can encourage local businesses to install cameras in their parking lots to prevent customer theft, tell citizens to keep valuables out of sight. If we know simple assault adds up to roughly 9% of all incidents in the Houston area, we can inform citizens of the risk and advise them to carry modes of self-defense whether that be pepper-spray or a handgun.

1.2.4 Data Spatial Analysis with K-Clustering

[1]:
```
# Code included in .html / .ipynb file. Not available on PDFs
```

Spatial analysis utilizing K-Clustering is helpful to find hotspots of incidents occuring. In this scenario, a K-Value of 13 is chosen for the 13 divisions in Houston. While each division is not labeled on the map, the 13 clusters are intended to show a possible re-mapping of the divisons in terms of efficency regarding incidents rather geographic location. This is important as in potential districts converging, diverging, dissolves, or remapping, this can be an outline for a beneficial way to remap our districts to protect the community. This could also help districts relocate and remanage current positions of where officers patrol or how many officers patrol a designated area. Alternatively, this analysis helps positon our police force to be prepared for the harm that comes in our communities' way.

1.3 Conclusion

Data Analysis can be highly beneficial when it comes to protecting our communities. With analysis' such as regression to check on patterns in yearly/ daily trends, K-Clustering to see what areas are highly dense and even having the frequency of incidents occuring can provide helpful stepping stones in organizing our police force. Through the use of the data analysis, we have discovered that most incidents peak during the summertime and dramatically drop during December to quickly pick up a roughly 700 incidents a month until the cycle repeats. We have indentified the most dangerous zip-codes and descriptions of where incidents occurs to help locate and inform our officers on where to go and what challenges they will likely face performing their job. Lastly, we have performed K-Clustering utilizing K = 13 (The number of divisions in Houston) to help strategically place our officiers in position for the best possible performance. While this analysis only consists of data in 2023, the more and more data we insert into this analysis the more beneficial it would provide. It would prove worthwhile to revisit this dataset and add new and old entries to help us in the future, as the more data we have the more accurate our numbers become.