Customer-Service-Requests-Analysis

January 30, 2022

1 Customer-Service-Requests-Analysis

[1]:
```
### import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style
import seaborn as sns
%matplotlib inline
```

[2]:
```
service311 = pd.read_csv ('311_Service_Requests_from_2010_to_Present.csv')
```

```
/usr/local/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3063:
DtypeWarning: Columns (48,49) have mixed types.Specify dtype option on import or
set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

[3]:
```
service311.head()
```

[3]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip
0	32310363	12/31/2015 11:59:45 PM	01-01-16 0:55	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0
1	32309934	12/31/2015 11:59:44 PM	01-01-16 1:26	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0
2	32309159	12/31/2015 11:59:29 PM	01-01-16 4:51	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0
3	32305098	12/31/2015 11:57:46 PM	01-01-16 7:43	NYPD	New York City Police Department	Illegal Parking			
4	32306529	12/31/2015 11:56:58 PM	01-01-16 3:24	NYPD	New York City Police Department	Illegal Parking			

```
3  Commercial Overnight Parking  Street/Sidewalk       10461.0
4              Blocked Sidewalk  Street/Sidewalk       11373.0

        Incident Address  … Bridge Highway Name Bridge Highway Direction  \
0    71 VERMILYEA AVENUE  …                 NaN                       NaN
1        27-07 23 AVENUE  …                 NaN                       NaN
2  2897 VALENTINE AVENUE  …                 NaN                       NaN
3    2940 BAISLEY AVENUE  …                 NaN                       NaN
4          87-14 57 ROAD  …                 NaN                       NaN

  Road Ramp Bridge Highway Segment Garage Lot Name Ferry Direction  \
0       NaN                    NaN             NaN             NaN
1       NaN                    NaN             NaN             NaN
2       NaN                    NaN             NaN             NaN
3       NaN                    NaN             NaN             NaN
4       NaN                    NaN             NaN             NaN

  Ferry Terminal Name   Latitude  Longitude  \
0                 NaN  40.865682 -73.923501
1                 NaN  40.775945 -73.915094
2                 NaN  40.870325 -73.888525
3                 NaN  40.835994 -73.828379
4                 NaN  40.733060 -73.874170

                                  Location
0   (40.86568153633767, -73.92350095571744)
1  (40.775945312321085, -73.91509393898605)
2  (40.870324522111424, -73.88852464418646)
3   (40.83599404683083, -73.82837939584206)
4  (40.733059618956815, -73.87416975810375)

[5 rows x 53 columns]
```

[4]:
```
service311.shape
```

[4]:
```
(300698, 53)
```

[5]:
```
service311.columns
```

[5]:
```
Index(['Unique Key', 'Created Date', 'Closed Date', 'Agency', 'Agency Name',
       'Complaint Type', 'Descriptor', 'Location Type', 'Incident Zip',
       'Incident Address', 'Street Name', 'Cross Street 1', 'Cross Street 2',
       'Intersection Street 1', 'Intersection Street 2', 'Address Type',
       'City', 'Landmark', 'Facility Type', 'Status', 'Due Date',
       'Resolution Description', 'Resolution Action Updated Date',
       'Community Board', 'Borough', 'X Coordinate (State Plane)',
       'Y Coordinate (State Plane)', 'Park Facility Name', 'Park Borough',
```

```
       'School Name', 'School Number', 'School Region', 'School Code',
       'School Phone Number', 'School Address', 'School City', 'School State',
       'School Zip', 'School Not Found', 'School or Citywide Complaint',
       'Vehicle Type', 'Taxi Company Borough', 'Taxi Pick Up Location',
       'Bridge Highway Name', 'Bridge Highway Direction', 'Road Ramp',
       'Bridge Highway Segment', 'Garage Lot Name', 'Ferry Direction',
       'Ferry Terminal Name', 'Latitude', 'Longitude', 'Location'],
      dtype='object')
```

[6]:
```
service311['Complaint Type'].unique()
```

[6]:
```
array(['Noise - Street/Sidewalk', 'Blocked Driveway', 'Illegal Parking',
       'Derelict Vehicle', 'Noise - Commercial',
       'Noise - House of Worship', 'Posting Advertisement',
       'Noise - Vehicle', 'Animal Abuse', 'Vending', 'Traffic',
       'Drinking', 'Bike/Roller/Skate Chronic', 'Panhandling',
       'Noise - Park', 'Homeless Encampment', 'Urinating in Public',
       'Graffiti', 'Disorderly Youth', 'Illegal Fireworks',
       'Ferry Complaint', 'Agency Issues', 'Squeegee', 'Animal in a Park'],
      dtype=object)
```

[7]:
```
service311['Descriptor'].unique()
```

[7]:
```
array(['Loud Music/Party', 'No Access', 'Commercial Overnight Parking',
       'Blocked Sidewalk', 'Posted Parking Sign Violation',
       'Blocked Hydrant', 'With License Plate', 'Partial Access',
       'Unauthorized Bus Layover', 'Double Parked Blocking Vehicle',
       'Double Parked Blocking Traffic', 'Vehicle', 'Loud Talking',
       'Banging/Pounding', 'Car/Truck Music', 'Tortured',
       'In Prohibited Area', 'Congestion/Gridlock', 'Neglected',
       'Car/Truck Horn', 'In Public', 'Other (complaint details)', nan,
       'No Shelter', 'Truck Route Violation', 'Unlicensed',
       'Overnight Commercial Storage', 'Engine Idling',
       'After Hours - Licensed Est', 'Detached Trailer',
       'Underage - Licensed Est', 'Chronic Stoplight Violation',
       'Loud Television', 'Chained', 'Building', 'In Car',
       'Police Report Requested', 'Chronic Speeding',
       'Playing in Unsuitable Place', 'Drag Racing',
       'Police Report Not Requested', 'Nuisance/Truant', 'Homeless Issue',
       'Language Access Complaint', 'Disruptive Passenger',
       'Animal Waste'], dtype=object)
```

[8]:
```
complaintTypecity = pd.DataFrame({'count':
                                  service311.groupby(['Complaint Type','City']).
 ↪size()}).reset_index()
complaintTypecity
```

[8]:
```
    Complaint Type           City  count
0     Animal Abuse        ARVERNE     38
1     Animal Abuse        ASTORIA    125
2     Animal Abuse        BAYSIDE     37
3     Animal Abuse      BELLEROSE      7
4     Animal Abuse   BREEZY POINT      2
..             …            …    …
759        Vending  STATEN ISLAND     25
760        Vending      SUNNYSIDE     15
761        Vending     WHITESTONE      1
762        Vending      WOODHAVEN      6
763        Vending       WOODSIDE     15

[764 rows x 3 columns]
```

[9]:
```
service311.groupby(['Borough','Complaint Type','Descriptor']).size()
```

[9]:
```
Borough      Complaint Type         Descriptor
BRONX        Animal Abuse           Chained                      132
                                    In Car                        36
                                    Neglected                    673
                                    No Shelter                    71
                                    Other (complaint details)    311
                                                                 …
Unspecified  Noise - Vehicle        Engine Idling                 11
             Posting Advertisement  Vehicle                        1
             Traffic                Truck Route Violation          1
             Vending                In Prohibited Area             2
                                    Unlicensed                     5
Length: 288, dtype: int64
```

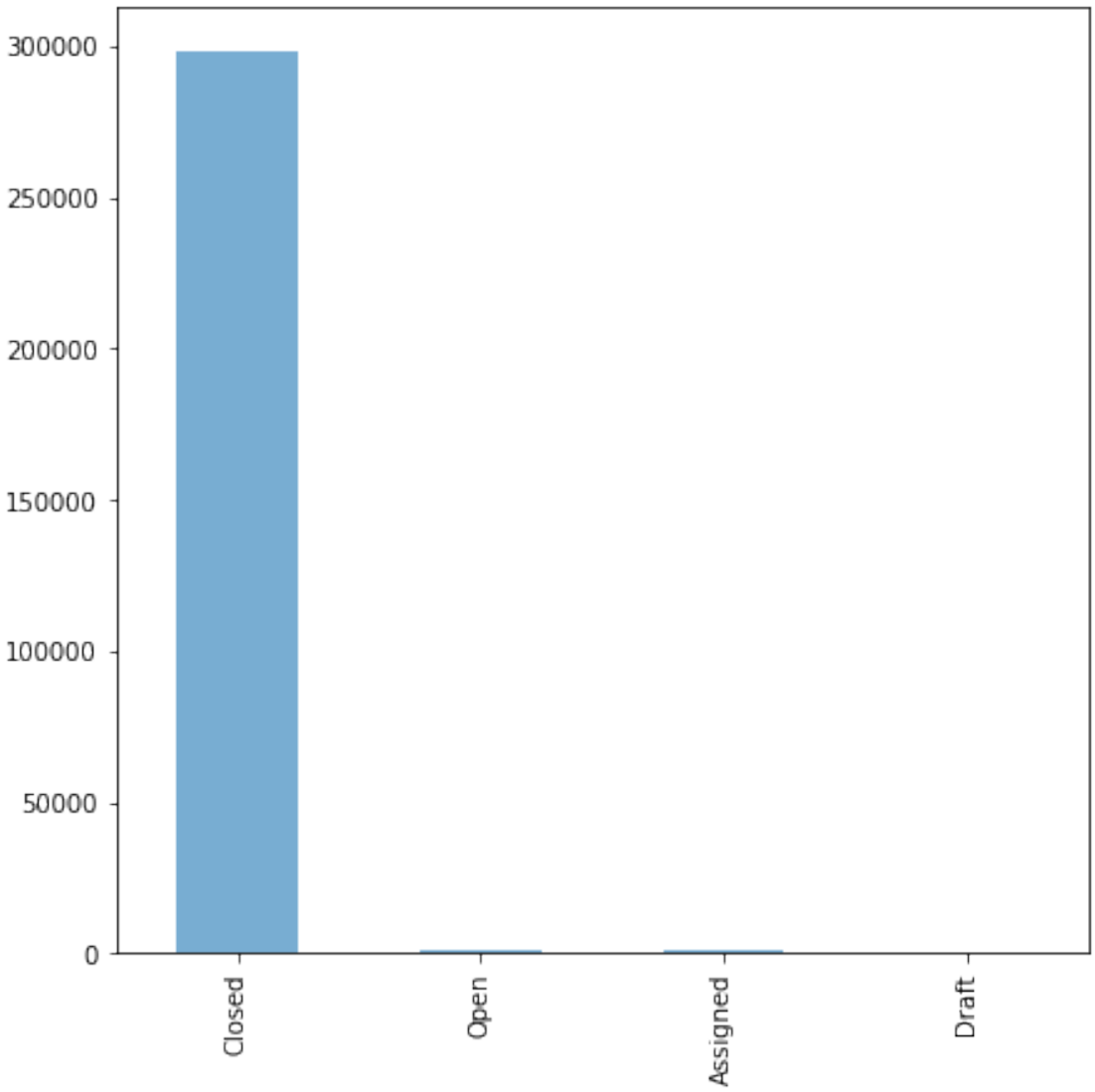
[10]:
```
import datetime
```

[11]:
```
df = pd.read_csv("311_Service_Requests_from_2010_to_Present.csv",␣
↪parse_dates=["Created Date", "Closed Date"])
```

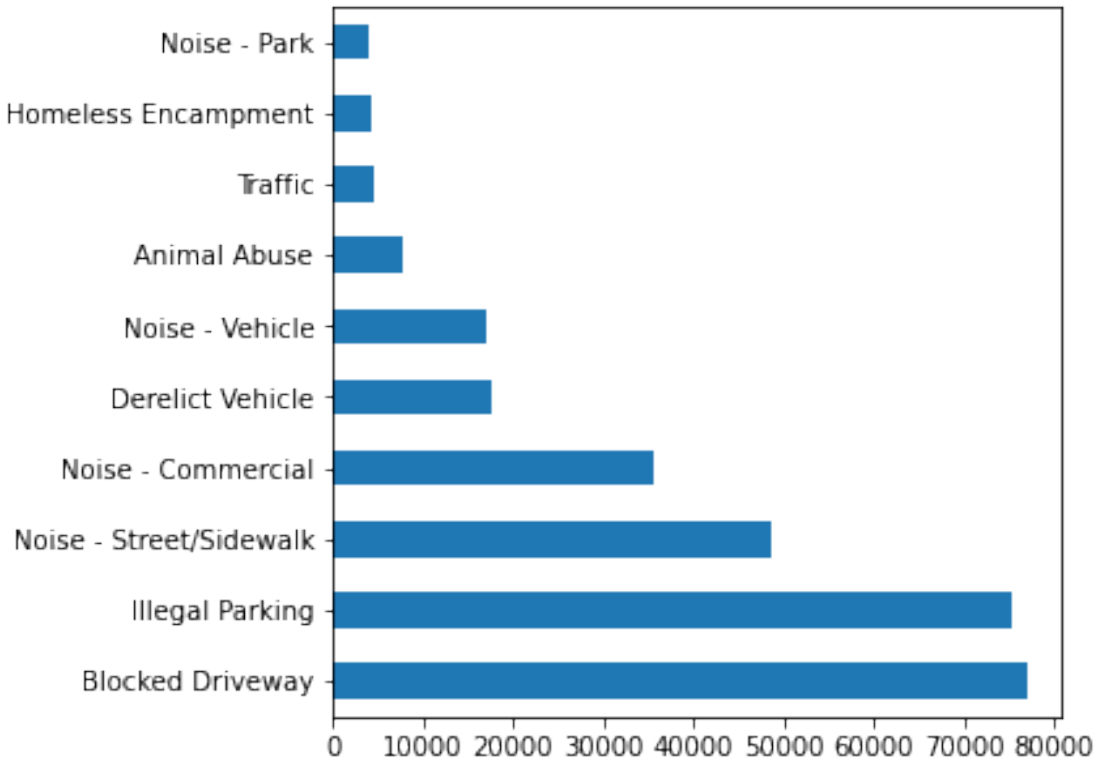
```
/usr/local/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3063:
DtypeWarning: Columns (48,49) have mixed types.Specify dtype option on import or
set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

[12]:
```
df["Request_Closing_Time"] = df["Closed Date"] - df["Created Date"]
```

[13]:
```
#Have a look at the status of tickets
df['Status'].value_counts().plot(kind='bar',alpha=0.6,figsize=(7,7))
plt.show()
```



[14]:
```
#Complaint type Breakdown with bar plot to figure out majority of complaint␣
 ↪types and top 10 complaints
service311['Complaint Type'].value_counts().head(10).
 ↪plot(kind='barh',figsize=(5,5));
```



[15]:
```
service311.groupby(["Borough","Complaint Type","Descriptor"]).size()
```

[15]:
```
Borough       Complaint Type          Descriptor
BRONX         Animal Abuse            Chained                      132
                                      In Car                        36
                                      Neglected                    673
                                      No Shelter                    71
                                      Other (complaint details)    311
                                                                   ...
Unspecified   Noise - Vehicle         Engine Idling                 11
              Posting Advertisement   Vehicle                        1
              Traffic                 Truck Route Violation          1
              Vending                 In Prohibited Area             2
                                      Unlicensed                     5
Length: 288, dtype: int64
```

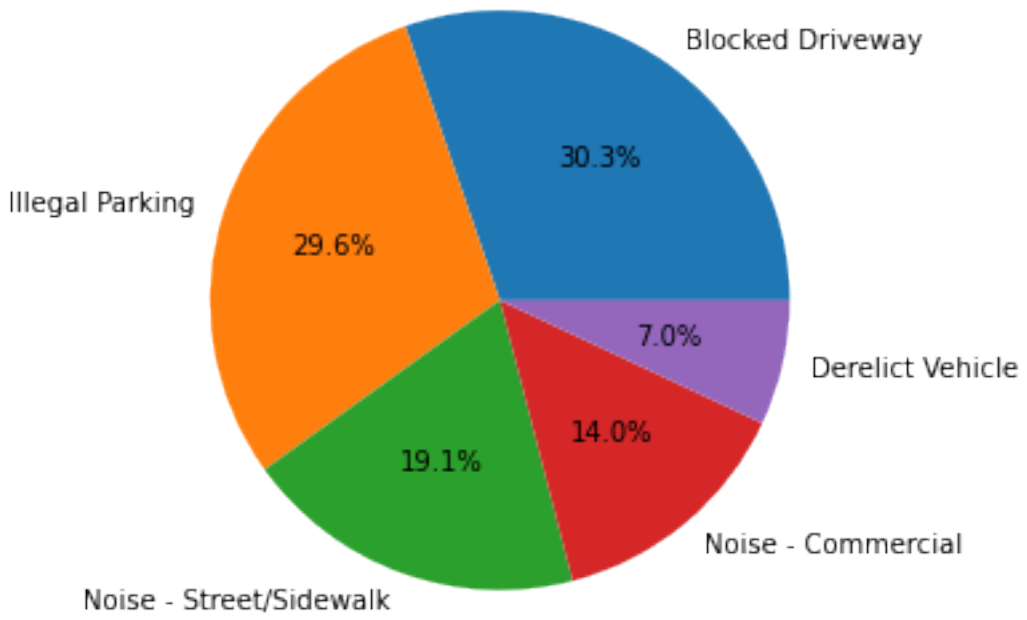
[16]:
```
majorcomplints=service311.dropna(subset=["Complaint Type"])
majorcomplints=service311.groupby("Complaint Type")

sortedComplaintType = majorcomplints.size().sort_values(ascending = False)
sortedComplaintType = sortedComplaintType.to_frame('count').reset_index()
```

```
sortedComplaintType
sortedComplaintType.head(10)
```

[16]:

	Complaint Type	count
0	Blocked Driveway	77044
1	Illegal Parking	75361
2	Noise - Street/Sidewalk	48612
3	Noise - Commercial	35577
4	Derelict Vehicle	17718
5	Noise - Vehicle	17083
6	Animal Abuse	7778
7	Traffic	4498
8	Homeless Encampment	4416
9	Noise - Park	4042

[17]:
```
sortedComplaintType = sortedComplaintType.head()
plt.figure(figsize=(5,5))
plt.pie(sortedComplaintType['count'],labels=sortedComplaintType["Complaint␣
↪Type"], autopct="%1.1f%%")
plt.show()
```



[18]:
```
#Group dataset by complaint type to display plot against city
groupedby_complainttype = df.groupby('Complaint Type')
```

[19]:
```
grp_data = groupedby_complainttype.get_group('Blocked Driveway')
grp_data.shape
```

[19]:
```
(77044, 54)
```

[20]:
```
#To get nan values in the entire dataset
df.isnull().sum()
```

[20]:
```
Unique Key                          0
Created Date                        0
Closed Date                      2164
Agency                              0
Agency Name                         0
Complaint Type                      0
Descriptor                       5914
Location Type                     131
Incident Zip                     2615
Incident Address                44410
Street Name                     44410
Cross Street 1                  49279
Cross Street 2                  49779
Intersection Street 1          256840
Intersection Street 2          257336
Address Type                     2815
City                             2614
Landmark                       300349
Facility Type                    2171
Status                              0
Due Date                            3
Resolution Description              0
Resolution Action Updated Date   2187
Community Board                     0
Borough                             0
X Coordinate (State Plane)       3540
Y Coordinate (State Plane)       3540
Park Facility Name                  0
Park Borough                        0
School Name                         0
School Number                       0
School Region                       1
School Code                         1
School Phone Number                 0
School Address                      0
School City                         0
School State                        0
School Zip                          1
School Not Found                    0
```

```
School or Citywide Complaint      300698
Vehicle Type                      300698
Taxi Company Borough              300698
Taxi Pick Up Location             300698
Bridge Highway Name               300455
Bridge Highway Direction          300455
Road Ramp                         300485
Bridge Highway Segment            300485
Garage Lot Name                   300698
Ferry Direction                   300697
Ferry Terminal Name               300696
Latitude                            3540
Longitude                           3540
Location                            3540
Request_Closing_Time                2164
dtype: int64
```

[21]:
```
#fix blank values in City column
df['City'].dropna(inplace=True)
```

[22]:
```
#Shape after dropping nan values
df['City'].shape
```

[22]:
```
(300698,)
```

[23]:
```
#count of null values in grouped city column data
grp_data['City'].isnull().sum()
```

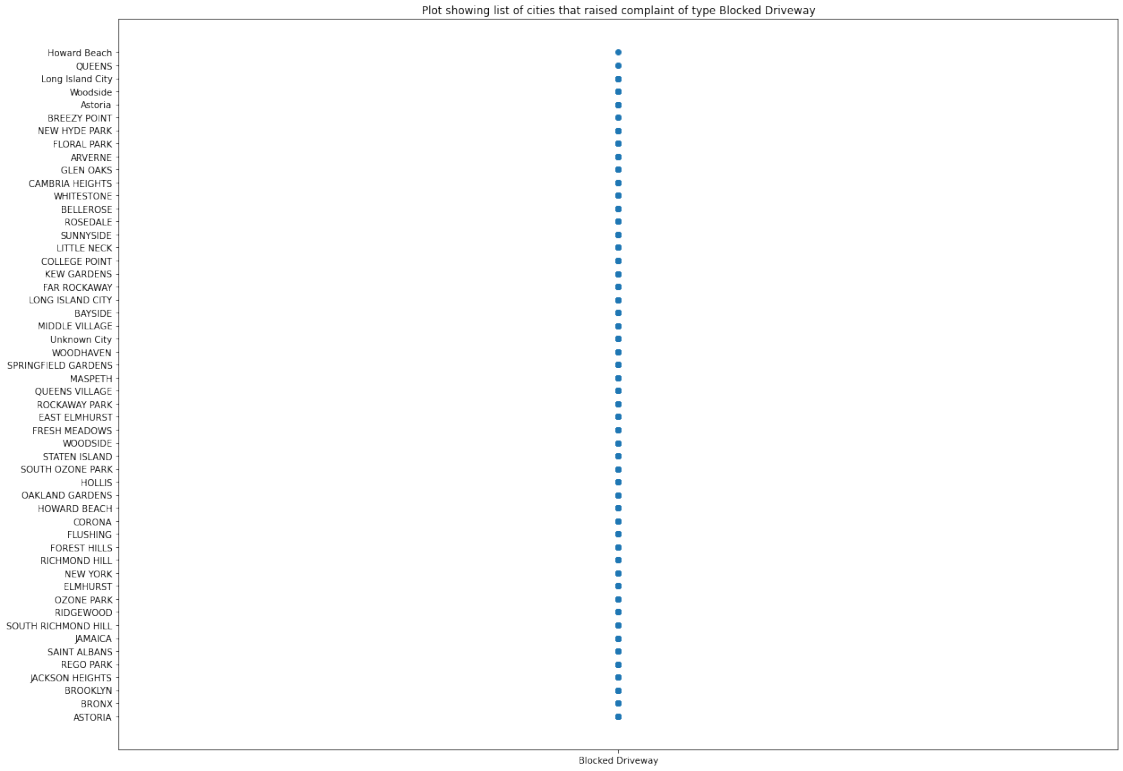
[23]:
```
283
```

[24]:
```
#fix those NAN with "unknown city" value instead
grp_data['City'].fillna('Unknown City', inplace =True)
```

```
/usr/local/lib/python3.7/site-packages/pandas/core/series.py:4536:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  downcast=downcast,
```

[25]:
```
#Scatter plot displaying all the cities that raised complaint of type 'Blocked␣
 ↪Driveway'
plt.figure(figsize=(20, 15))
plt.scatter(grp_data['Complaint Type'],grp_data['City'])
plt.title('Plot showing list of cities that raised complaint of type Blocked␣
 ↪Driveway')
```

```
plt.show()
```


Plot showing list of cities that raised complaint of type Blocked Driveway
Howard Beach
QUEENS
Long Island City
Woodside
Astoria
BREEZY POINT
NEW HYDE PARK
FLORAL PARK
ARVERNE
GLEN OAKS
CAMBRIA HEIGHTS
WHITESTONE
BELLEROSE
ROSEDALE
SUNNYSIDE
LITTLE NECK
COLLEGE POINT
KEW GARDENS
FAR ROCKAWAY
LONG ISLAND CITY
BAYSIDE
MIDDLE VILLAGE
Unknown City
WOODHAVEN
SPRINGFIELD GARDENS
MASPETH
QUEENS VILLAGE
ROCKAWAY PARK
EAST ELMHURST
FRESH MEADOWS
WOODSIDE
STATEN ISLAND
SOUTH OZONE PARK
HOLLIS
OAKLAND GARDENS
HOWARD BEACH
CORONA
FLUSHING
FOREST HILLS
RICHMOND HILL
NEW YORK
ELMHURST
OZONE PARK
RIDGEWOOD
SOUTH RICHMOND HILL
JAMAICA
SAINT ALBANS
REGO PARK
JACKSON HEIGHTS
BROOKLYN
BRONX
ASTORIA
Blocked Driveway

[ ]: