

Project Manual
311 Service Requests Analysis
DS-GA-1007 Programming for Data Science

Pan Ding pd878

Liwen Tian tl1759

Maha Yaqub my1288

List of files required to run this program:

- 311 Complaint Dataset for the year 2014
- Zip_borough.csv
- tl_2013_us_zcta510 (Folder containing all the shape files needed for the maps)
- All the modules uploaded on the Github folder for the final project as listed below:
 - __init__.py
 - Main.py
 - start_function.py
 - prompt.py
 - ChartsPlot_main.py
 - load_clean_data.py
 - Plot_Charts.py
 - UserInput_Charts.py
 - TimeSeries_Main.py
 - UserInput_Time.py
 - TimeInterval.py
 - Plot_TimeSeries.py
 - AgencySeparation.py
 - BoroughSeparation.py
 - cleanup_MapData.py
 - mapModule.py
 - Plot_Maps.py

- UserInput_Maps.py

Instructions on how to obtain Data Files for the Project:

For the user's convenience, all the data files have been uploaded in a Dropbox. The code is all pushed on the Git repository. The following link provides access to the Dropbox from where you can download the data files:

https://www.dropbox.com/sh/n9hcwljxrd5yq0/AABbpQOh1CQ3IKTxgx_9HNGDa?dl=0

Alternate methods to obtain Data Files:

Alternatively, the 311 complaints dataset can be downloaded from the website using the following steps:

1. Go to
<https://nycopendata.socrata.com/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9>
2. Filter the data on unique key = 'Created Date' is between '1 January 2014' to '31 December 2014'.
3. Click on 'Export'. Download the data as a CSV file.
4. The dataset is almost 900 MB.

The Shapefiles can be downloaded by the following steps:

1. Go to
<https://www.census.gov/cgi-bin/geo/shapefiles2013/main>
2. Select 'ZIP Code Tabulation Areas' from the dropdown menu.
3. Download the zipped folder and extract the shapefiles and include them in the project folder.

The zip_borough.csv file can be downloaded using the following link:

http://vgc.poly.edu/projects/gx5003-fall2014/week9/lab/data/zip_borough.csv

Additional Package Requirements:

Also, in order to run the code, the Python package Bokeh needs to be installed. This package has been used to plot interactive maps for the user to do a spatial analysis of the 311 complaints all over the five boroughs. The following command can be used on the terminal to install Bokeh:

```
sudo pip install bokeh
```

Note: Make sure you download the latest version of Bokeh.

Also, the shapefile library also needs to be installed. This can be done using the following command:

```
sudo pip install pyshp
```

Functionalities of the Program:

The objective of our project was to provide a tool for the user to analyze NYC 311 complaints over the year 2014. The program enables the user to analyze the dataset in the following ways:

1. Bar Charts and Pie Chart Analysis
 - Display a bar chart for a selected borough which shows the volume of complaints for each agency in that respective borough.
 - The user can select an agency and the percentage of complaints made for that agency in each borough will be displayed as a pie chart.
2. Time Series Analysis
 - The user also has an option to give a time interval in the form of months for which the program will plot a time series showing the number of complaints for each borough separately over time.
 - There is also an option to view the time series of complaints over time for a selected agency.

3. Spatial Analysis using Maps

- The first function is that the map will show each zipcode in the five boroughs color coded according to the maximum number of complaints. There will be a hover over every zipcode which can enable the user to view the top agency.
- There is also an option to compare two agencies. The function will create an analogous map for NYC to compare the two agencies in terms of number of complaints for each zipcode. The names of the two agencies will be given as the input parameters and a ratio will be calculated for their respective number of complaints for each zipcode.
- The third function will divide NYC into a grid based on the centroid of each zipcode and will represent the volume of complaints based on the increasing size of circles which will be proportional to the number of complaints in each zipcode.

How to run the program?

The following steps should be followed in order to make sure the code runs properly:

1. Make sure all the modules are in the folder. This can simply be done by running the test.py module using the following command:

`python test.py`

If the result is 'OK' then this means that all the modules are present in the folder.

2. Then check if the following files are present in the folder:

- 311 Complaint Dataset for the year 2014
- zip_borough.csv
- All the files contained in the folder tl_2013_us_zcta510:
 - tl_2013_us_zcta510.dbf
 - tl_2013_us_zcta510.prj
 - tl_2013_us_zcta510.shp
 - tl_2013_us_zcta510.shx

3. Type the following command in your terminal to launch the program:

```
python Main.py [311 Complaint Dataset] zip_borough.csv tl_2013_us_zcta510.shp
```

This command launches the program and then it gives instructions on how to proceed.

Descriptions of all Modules:

- `__init__.py`

This is the constructor for the package.

- `Main.py`

This is the main file. It has the function which launches the entire program. This is the file being used in the command that is used to run the package.

- `start_function.py`

This module contains the interactive user input for the main menu of the program. It asks the user what method of analysis they want to select.

- `prompt.py`

This is the code which prompts the user to select whether they want to go back to the main menu or make another plot in the same selected method.

- `ChartsPlot_main.py`

This file loads the csv file into a dataframe and cleans the data.

Then it asks the user to input a Borough name, and plots the volume of complaints for each agency in this Borough. It stops until the user enters 'finish'.

Then it asks the user to input an agency name, and plots the volume of complaints for this agency in each Borough. It stops until the user enters 'finish'.

- `load_clean_data.py`

This module defines a class to load the csv file to a dataframe and eliminates the irrelevant columns. It also defines a function that cleans the dataframe.

- Plot_Charts.py

This module contains two functions. One draws a bar plot for a given Borough. The other draws a pie plot for a given agency.

- UserInput_Charts.py

This module defines two functions to get input from the user. The functions will continue to ask for input until it gets a valid one. "Ctrl+C" will exit the main program.

- TimeSeries_Main.py

This module is also an interactive one. It asks the user to select what analysis method they want to do either by borough or by agency. It also asks the user to define a time interval.

- UserInput_Time.py

This module includes the function which takes in the time interval from the user. It also checks whether the end month is always greater than the start month.

- TimeInterval.py

This module takes in the selected time interval and adjusts the dataframe accordingly.

- Plot_TimeSeries.py

This is the class for the plots that are to be generated in the time series method of analysis.

- AgencySeparation.py

This function takes in the agency names and groups the data according to that agency.

- BoroughSeparation.py

This function groups the data by created date and borough.

- `cleanup_MapData.py`

This module cleans up the map points and matches it with the other data files so that the complaint data can be merged accordingly.

- `mapModule.py`

This interactive module asks the user what method of spatial analysis they want to select.

- `Plot_Maps.py`

This module matches the zipcodes in the csv file with the ones in the shapefile so that they can be plotted according to the complaint dataset.

- `UserInput_Maps.py`

This takes in the user input as two agencies which are to be compared spatially.