Breathe in Data: Visualization

Final Report

Tree Coverage, Air Quality and Air Quality Complaints

Janice Darling

CSC 59969 Visualization

18 May 2017

The main goal of any visualization is to present data in a lucid manner which allows for the effective execution of tasks such as recording, analysis and communication. This main goal ties in with the aim of this project which was to create a tool or dashboard which provided a visual representation of the multiple datasets. In particular, this dashboard will be a visualization that allows for the exploration the existence of a relationship between tree coverage and air quality complaints. Also, since the relationship between the two variables, tree coverage and air quality complaints are undoubtedly influenced by other factors, other datasets were introduced provide the viewer or user of the dashboard with the ability to draw deeper conclusions about the data that is being represented. The later introduced datasets include data concerning measures of various air pollutants as well as asthma discharge counts and rates. Each dataset will be later discussed in finer detail.

The main motivation to undertake tree coverage and air quality as the main topic for this project was due the ease of access of data concerning these subjects. The New York City Department of Parks and Recreation performs a census of street trees throughout the city every five years and makes this data publicly available on the New York City open data portal (https://opendata.cityofnewyork.us/). Also available on the portal are datasets for air quality complaints to 311. As the research and conversations with experts in the field for this project progressed, the importance of urban environmental health and its affecting factors became increasingly apparent. The specific datasets used for this project are as follows:

- 2015 Street Tree Census
- 2012 - 2014 Asthma Discharges
- 2012 - 2014 Fine Particulate Matter Measures

- 2012 - 2014 Black Carbon Measures
- 2012 - 2014 Nitrous Dioxide Measures

In each of the datasets above, a single data point is multivariate meaning that there are multiple attributes. Also, the first two datasets listed above are on the same level, ie. the zip code level, whereas the final three datasets are all on the borough community district level.

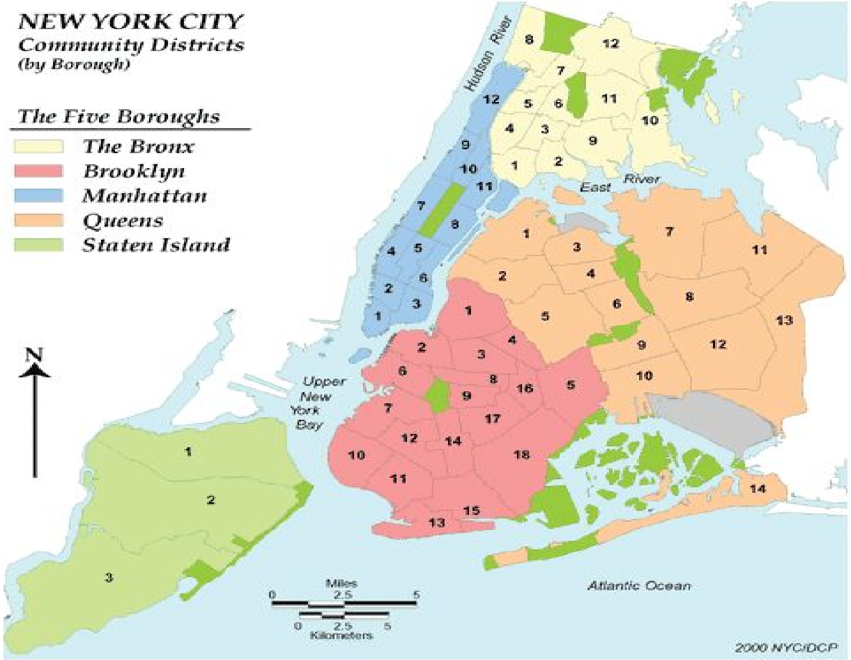


Fig. 1: Community Districts in each borough

Therefore, geoJSON files containing the outline of each community district and zip code were used in order to help visualize the data on the respective level.

As an exploratory data visualization, the end result of this project seeks to the give the user or viewer the ability to tell a story about the presence or absence of a relationship between tree coverage and air quality. Conventional wisdom would dictate that an area that has more trees will also have a lower number of air quality complaints and lower measures for the level of different air pollutants. This theory cannot be proven or rejected just by looking at the data which is in the comma separated values (csv) format. Therefore, it becomes necessary to present the data in an alternate manner by using different visualization concepts in order to tell the full story on the variables or at least what is represented in the datasets.

As previously mentioned, the datasets used were primarily sourced from the NYC open data portal. The tree and air quality complaints data sets are multivariate and have similar schema for the location attributes. Each tree and complaint has location data which is represented not only by the coordinates but also by street address and zip code. The third data set sourced is the asthma data which has the number of discharges for asthma and the rate per 10,000 individuals. This data was provided by the New York State Department of Health. Finally, the datasets for the air quality measures for Black Carbon, PM 2.5 and Nitrous Dioxide for various periods between 2012 and 2014 were obtained from the Environment and Health Data Portal.

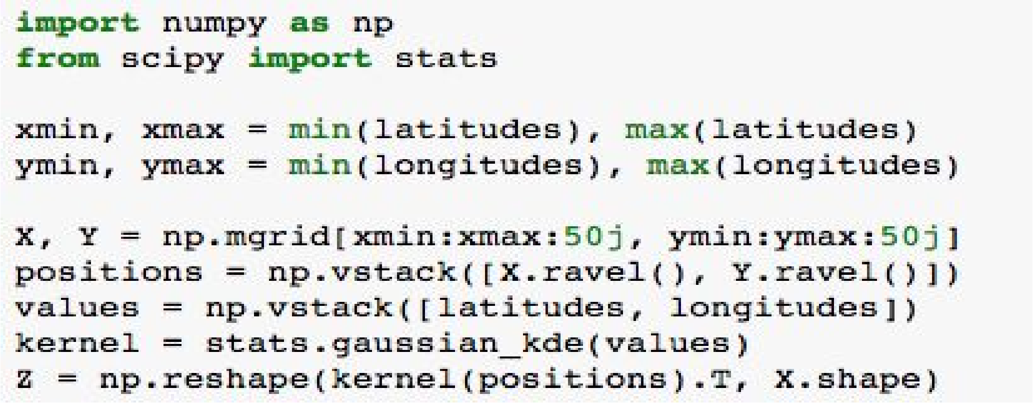
Speaking with an expert in the field of urban environment, geography and sickness provided a concept for what individuals would like to see in an interactive dashboard comparing datasets concerning the above subjects. It also provided the opportunity to apply different concepts visualization in an effort to attain the goals of any visualization, as well as to keep in line with main task of the project which was to present the data in such as way as to allow the user of the dashboard to draw a conclusion about the relationship between air quality, air quality complaints and tree coverage.

The tools used in the creation of the visualization are as follows:

- Python
- HTML, CSS, Javascript

- Jupyter Notebooks

Python was used extensively for processing the data and breaking it down into a form that could be easily read and utilized later on. In order to normalize the tree coverage and air quality complaints datasets, the scipy python package was used to perform a Kernel Density Estimate. Kernel Density Estimation (KDE) is a non-parametric method of estimating the probability density function of a random variable(scipy.stats.gaussian_kde para 1). The works for both univariate and multivariate data. The image below shows how the KDE values were obtained for each the points in the tree data. The code was also applied to the air quality complaints data since the schema are very similar.



```
import numpy as np
from scipy import stats

xmin, xmax = min(latitudes), max(latitudes)
ymin, ymax = min(longitudes), max(longitudes)

X, Y = np.mgrid[xmin:xmax:50j, ymin:ymax:50j]
positions = np.vstack([X.ravel(), Y.ravel()])
values = np.vstack([latitudes, longitudes])
kernel = stats.gaussian_kde(values)
Z = np.reshape(kernel(positions).T, X.shape)
```

Fig. 2: KDE for Tree Coordinates

The Python package matplotlib was then used to create a kde plot of the of the points obtained. The result is below:

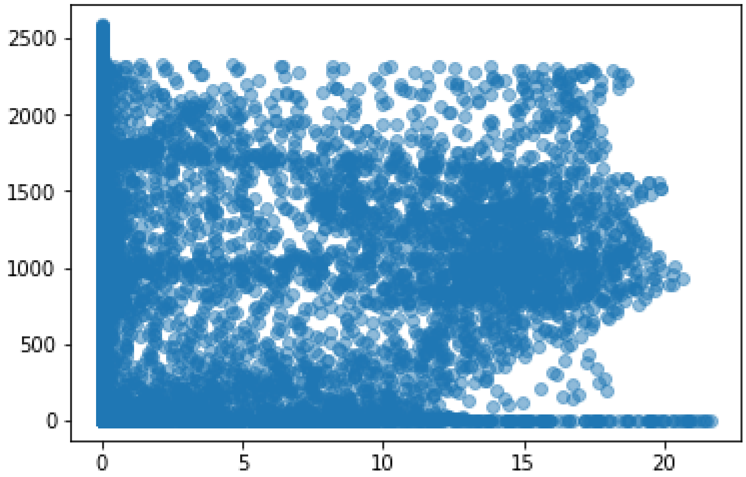
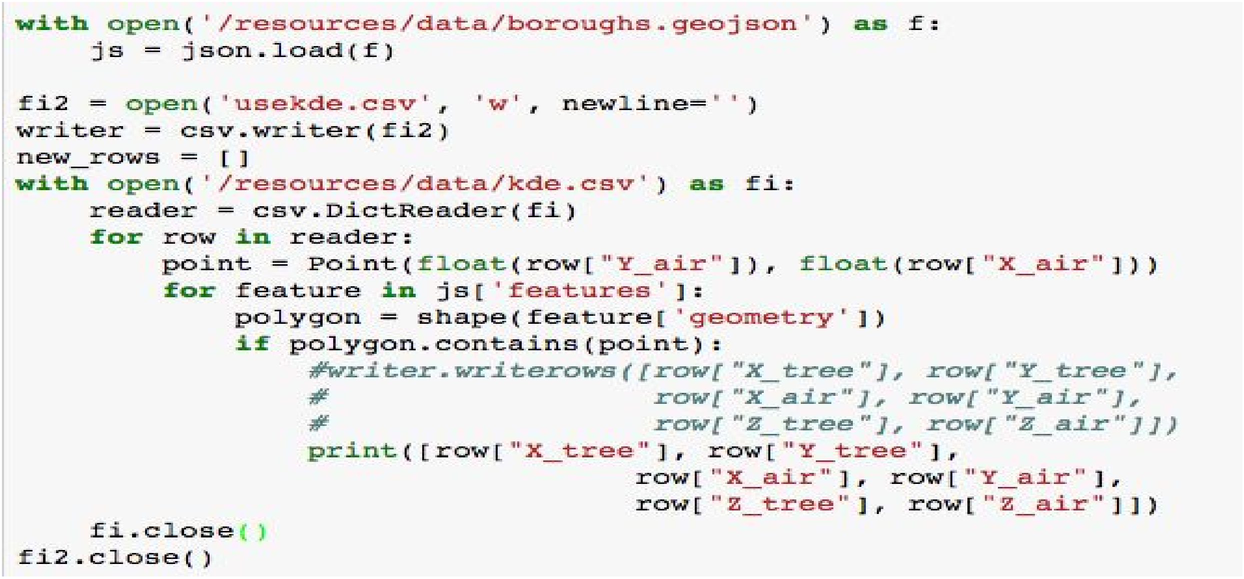


Fig 3: Plot of Air Quality Complaints vs Tree Coverage

The plot exposed the problem of the inclusion of points that did not have an occurrence of trees or air quality complaints. This was remedied by using the shapely and csv packages. The image below show the code used to remove such points.



```
with open('/resources/data/boroughs.geojson') as f:
    js = json.load(f)

fi2 = open('usekde.csv', 'w', newline='')
writer = csv.writer(fi2)
new_rows = []
with open('/resources/data/kde.csv') as fi:
    reader = csv.DictReader(fi)
    for row in reader:
        point = Point(float(row["Y_air"]), float(row["X_air"]))
        for feature in js['features']:
            polygon = shape(feature['geometry'])
            if polygon.contains(point):
                #writer.writerows([row["X_tree"], row["Y_tree"],
                #                  row["X_air"], row["Y_air"],
                #                  row["Z_tree"], row["Z_air"]])
                print([row["X_tree"], row["Y_tree"],
                                  row["X_air"], row["Y_air"],
                                  row["Z_tree"], row["Z_air"]])
    fi.close()
fi2.close()
```

Fig 4: Code used to remove erroneous KDE points

In the above code, the shapely package is used to create a point for each of the coordinates returned after laying the meshgrid from Fig 3. Using a geoJSON file containing the polygon or outline of each borough in New York City, a check was made to determine if the point fell within the polygon. If this was true, the point was then written to a csv file for later usage in creating the dashboard.

The final dashboard was then created through the extensive use of different Javascript plugins. The main map was created using Leaflet.js and Mapbox.js which are packages for the creation of interactive maps.

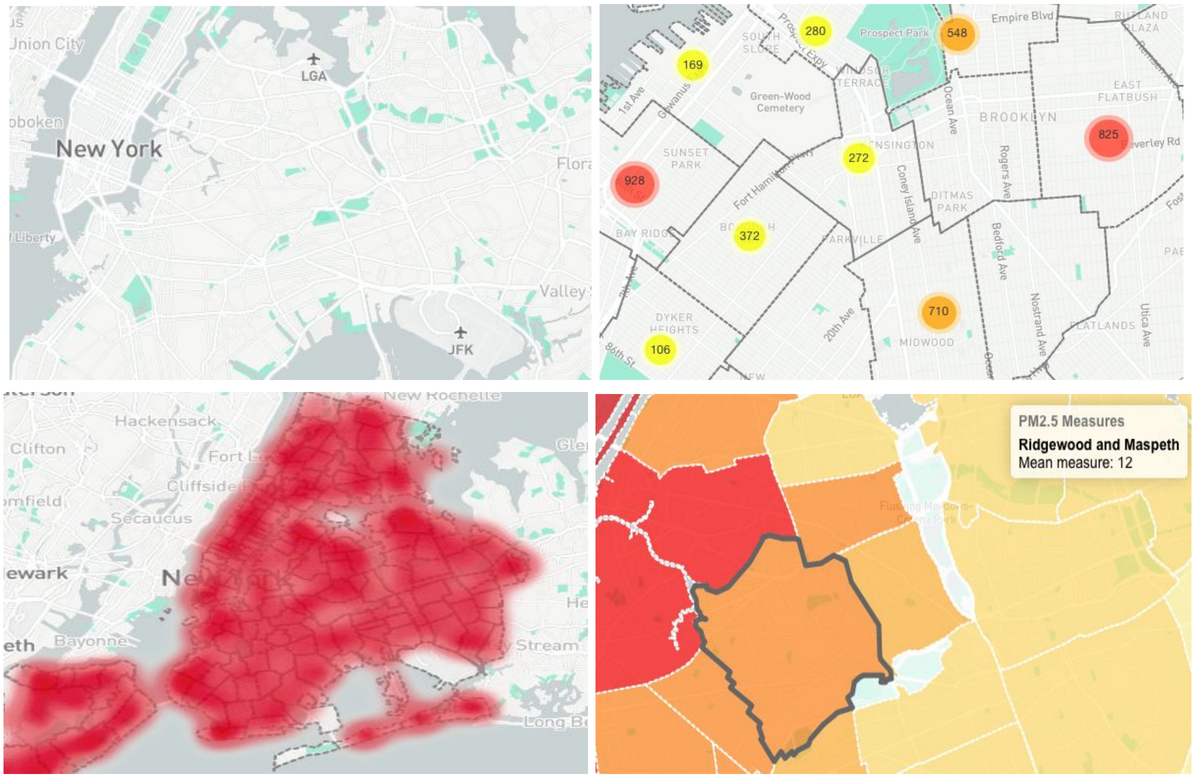


Fig. 5: Clockwise : Blank Leaflet.js map with Mapbox.js tiles; Clustering points on map; Heatmap created from tree coordinates; Choropleth map showing PM 2.5 measures

In order to allow the user to be able to draw a conclusion between the tree coverage and the air quality complaints, a scatterplot matrix will be used. A scatterplot matrix is a allows for the rough determination of whether or not a linear correlation occurs between multiple variables (Scatterplot Matrices para 1). The image below provides and example:

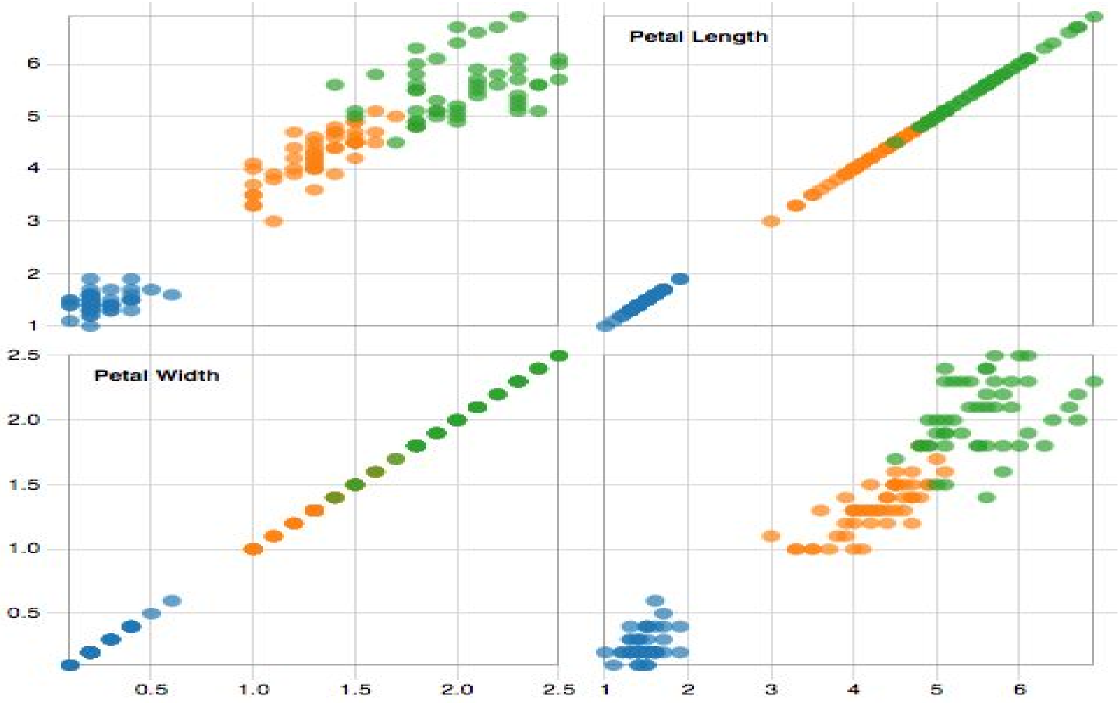


Fig 6. Scatterplot Matrix Showing the linear correlation between two variables

In Fig. 6, the two variables Petal Width and Petal Length are being investigated to determine if a correlation exists. In the top left hand plot, the x-axis represents Petal Width and the scale can be found at the bottom of the lower left hand plot. The y-axis is represents the Petal Length which denotes the graph represents a plot of Petal Width versus Petal Length. The other plots in the matrix can be read in a similar fashion. Through the rough inspection of each of the plots above, one can draw the conclusion that there is a linear correlation between the two variables being considered. The intent was to extend the concept to the tree coverage and air quality complaints using d3 to create an interactive plot. After the scatterplot was created, the user, in addition to being able to roughly tell whether or not a linear correlation exists between the two variables, could then hover over a point in the scatterplot matrix. The section on the map by zip code level represented by that point colored by a raw count of the trees in the area would then be highlighted. This would serve the purpose of allowing the user of the dashboard to take note of where in the density plot the area falls in comparison to the rest of the data.

In conclusion, the creation of the exploratory visualization revealed the importance of taking into consideration not only visual concepts such as color but also the concepts needed to draw a proper conclusion about the data. Since the datasets are expansive in terms of the number of points and attributes represented, trends or correlations cannot be easily derived just through the inspection of the csv files. Therefore methods must be used to normalize the data before attempts are made to create a visual representation.

Works Cited

New York City Community District Map. New York City Department of City Planning, n.d. Web. 20 May 2017. <http://www.fcny.org/cmgp/streets/pages/nyc_cdmap4.htm>.

Bostock, Mike. "Scatterplot Matrix." *Popular Blocks*. N.p., 2 June 2016. Web. 20 May 2017. <https://bl.ocks.org/mbostock/3213173>.

Moonheadsing. "Scatterplot Matrices." *Learning Omics*. N.p., 01 Feb. 2013. Web. 20 May 2017. <https://learningomics.wordpress.com/2013/01/31/scatterplot-matrices/>.

"Scipy.stats.gaussian_kde." *Scipy.stats.gaussian_kde — SciPy v0.17.0 Reference Guide*. N.p., 20 Feb. 2016. Web. 20 May 2017. <https://docs.scipy.org/doc/scipy-0.17.0/reference/generated/scipy.stats.gaussian_kde.html#scipy-stats-gaussian-kde>.