

Captstone 2 Milestone Report 1

(The final product can be found at WhereToRestaurant.pythonanywhere.com)

Problem Statement

Location is one of the most important factors that determines whether a restaurant will succeed or fail. Restaurateurs must carefully select locations to open new restaurants in order to give their business the best chance to succeed, but how do they go about making that decision? Several factors should be considered including: How many people live in the area? What kind of income do they have? How many and what kind of other restaurants are already in the area? The plan for this project is to provide a tool that will simplify this decision making process by displaying the relevant information on an interactive map, including a single metric that will rank zip codes based on the likely opportunity for a successful new restaurant. With this tool, restaurateurs could select the type of cuisine for which they would like to open a restaurant and immediately see a map showing where the best opportunity is. Additionally they would see a list of the top 10 zip codes with more information about each zip code including the population, median income, and number of restaurants of their chosen cuisine type.

Gathering the Data

The data for this project will come from three sources. First, the U.S. Census provides [population, income, and age data](#) at the zip code level. Second, the city of New York provides [publicly available health inspection data](#) that includes both the zip code and cuisine type for all restaurants in New York City that have had a health inspection. Finally, jsspina.carto.com provides a geojson file containing the coordinates needed to map the boundaries of every zip code in New York City.

The population, income, and age data were acquired from factfinder.census.gov. On this website you are able to search for and extract data in CSV format. I start by selecting the

geography type of 5-digit zip code tabulation area, and then select all of New York. Then I select topics > people > basic count/estimate > population total. I choose the 2017 5-year estimate as it is the most recently available data. I then modify the table by transposing rows and columns so that each row is a zip code and I extract the data in CSV format. I then repeat these steps to get the income and age data so that I have three separate CSV files.

For the NY health inspection data, I visited data.cityofnewyork.us and chose to export the data in CSV format. Finally the zip code coordinates were found at jsspina.carto.com and were available for download in geojson format. I saved all of these files to the data folder in my project directory.

Cleaning the Data

I used Jupyter Notebook to load each of the data files and do data cleaning. For the population, income, and age data I simply dropped all unnecessary columns and then merged the three dataframes on the zip code key. For the age data I reduced the number of bins (columns) from 18 to 4. Those columns corresponded to the proportion of the population in each zip code that were between the ages of 0-14, 15-29, 30-54, and 55 and above.

In the health inspection data each row was a single inspection, which meant that I had duplicate restaurants because most restaurants are inspected more than once. I used Pandas' ``drop_duplicates()`` method to remove rows that had the same name, building, street, zip code, phone, etc. This reduced the number of rows from 401,000 to about 27,000. I then used a ``groupby`` statement to calculate the proportion of each cuisine type in each zip code. I now had a data frame where each row is a zip code and each column is a cuisine type, with the value being the proportion. This data contained 227 unique zip codes.

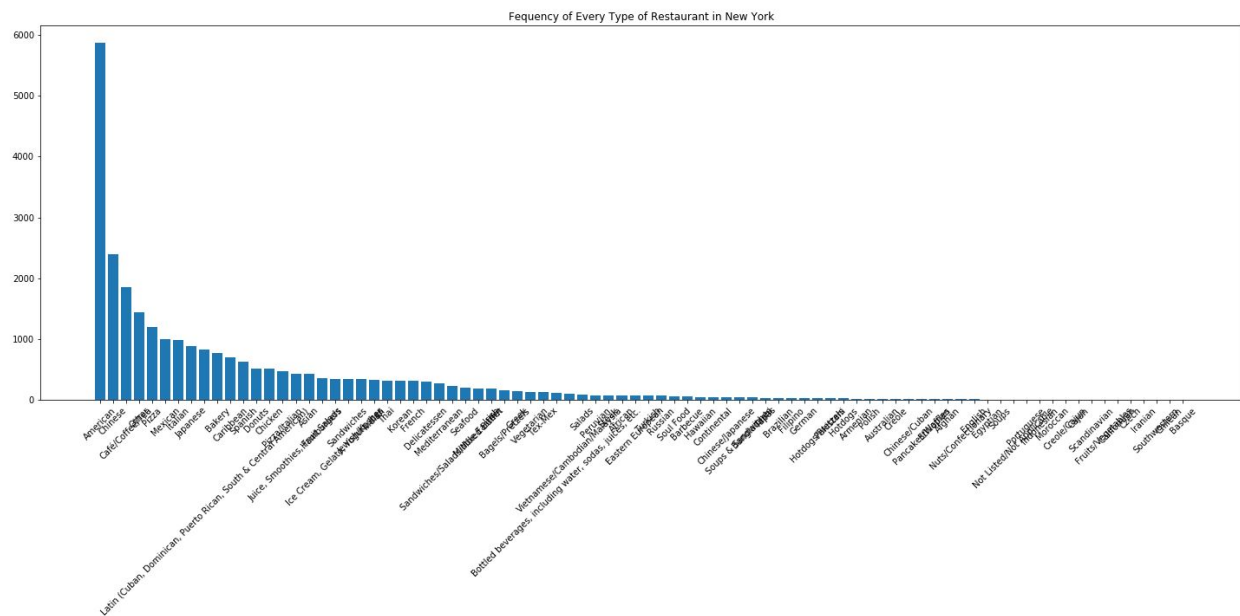
I then checked the coordinates data to see how many zip codes it had, and found that it had some duplicates which I removed using dictionary comprehension. The coordinates data

had some zip codes that were not in the health inspection data and vice versa, so I removed zip codes from each until I had 220 unique zip codes that were in both dataframes.

I then did a left merge of the census data onto the restaurant data. Because the census data contained all zip codes in the state of New York, the left merge excluded all zip codes which were not in New York City. I also included a column for the total number of restaurants in each zip code as well as number of restaurants per capita. I then found 41 rows for which the median income column was null. I examined these rows and found that they were very small zip codes with very few restaurants (many having less than 5) so I removed these rows as well, leaving a final count of 179 zip codes to be mapped.

Data Exploration

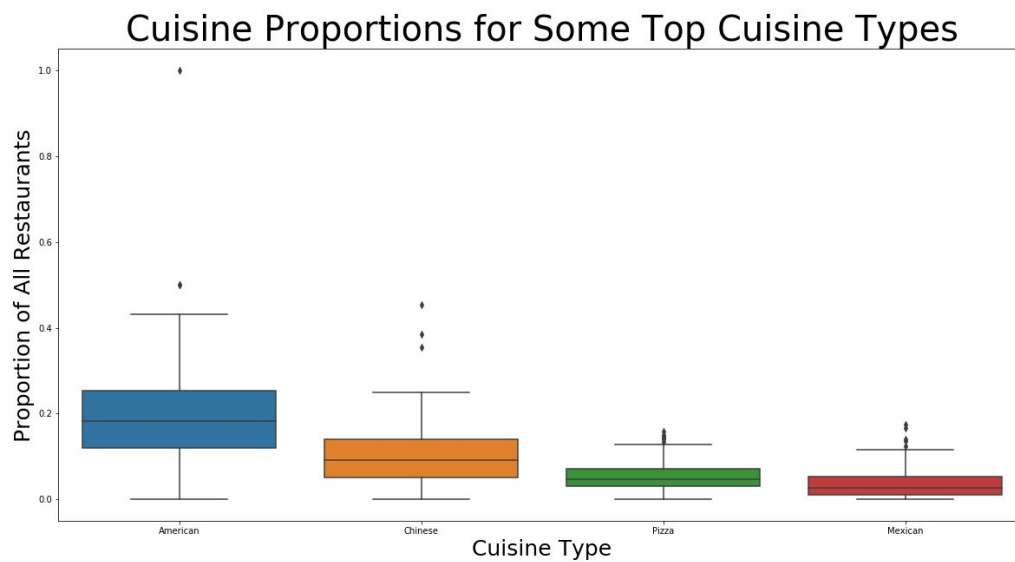
I plotted which cuisine types occurred most often and found the data to be very top heavy.



I found that American restaurants make up over 20% of all restaurants in NYC, and the top 25% of cuisine types make up over 80% of all restaurants in NYC. I believe based on this data that

the tool I am designing will be best suited for those top cuisine types as there will be sufficient data to make proper comparisons between zip codes.

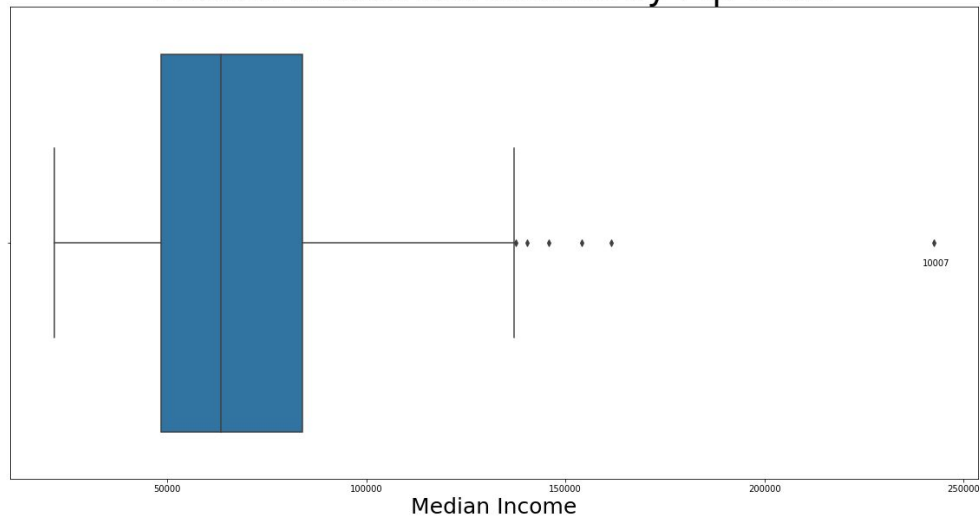
I also wanted to see how the proportions of some top cuisine types varied across zip codes, so I made a barplot.



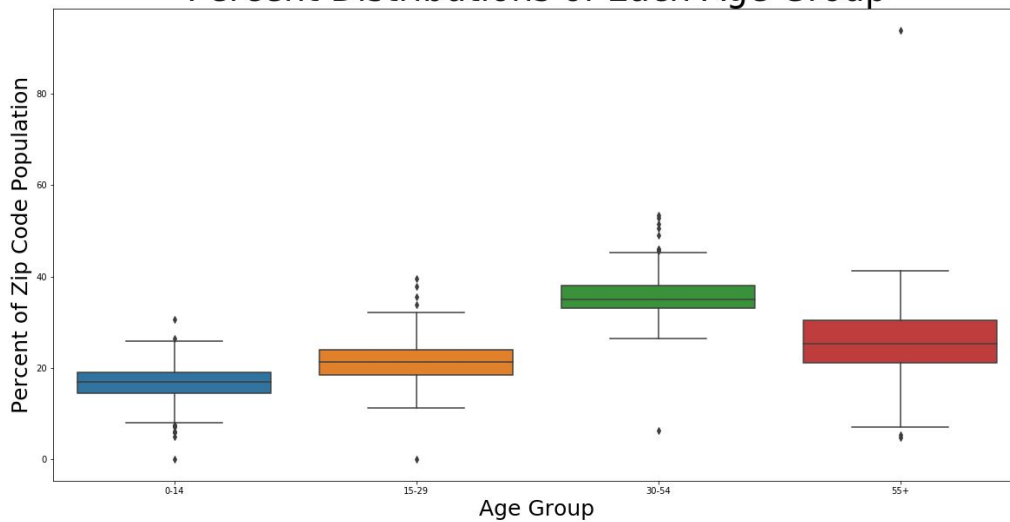
Even for some of the top cuisine types (Pizza, Mexican) the variation in proportion from zip code to zip code is quite low. Because of this, I think it will be best to MinMax scale these values to get a better comparison between zip codes.

I also looked at the distributions of Median Income and the 4 age groups.

Median Household Income by Zip Code



Percent Distributions of Each Age Group



Based on the way the data looks, I decided the best way to move forward with this tool was to create a single metric that can be used to rank the zip codes based on the opportunity for success for opening a new restaurant. Because the various data are on different scales and I am most interested in ranking the zip codes, it is appropriate to scale all of the relevant columns

with a MinMax scaler so that they vary between 0 and 1. I can then simply add up all the columns of interest (5 in total) and multiply by 20 to get a final “opportunity metric” that ranges from 0-100.

Making the First Map

I started by generating a single map, to understand how the code worked. I calculated a single opportunity metric for Italian cuisine and the 30-54 age group and added it to my data frame. To generate the choropleth maps I used the folium python library. The first step is to generate a plain map by defining the starting coordinates and default zoom level. I set these so that it was centered on New York City and showed the whole city. I then needed to add a choropleth layer onto the map that colored each zip code according to its opportunity metric. I started by using the `Choropleth` class described in the [folium documentation](#). After getting a map to generate I realized that I wanted to display the value of the opportunity metric when the user hovered their mouse over any zip code region. I searched stack overflow and found [this thread](#) which suggested using the `GeoJson` class instead and gave a link to [this example nbviewer page](#). I was able to rewrite my code to now display a tooltip when hovering over a zip code. To do this, though, I needed to insert the metric into the GeoJson properties, because this new class can only color the map from values in the GeoJson, not the data frame.

I knew I would need to generate many maps so I wrote a function that takes the cuisine type and target age group as inputs, and generates a map for those selections. It starts by calculating the correct opportunity metric and inserting it into the GeoJson object. It then generates the base map and adds the GeoJson layer. The GeoJson class has some parameters that allow me to make some style decisions like the thickness of the outlines around zip codes, the opacity of the colors, and a `highlight_function` that boldens the outline of a zip code when the mouse is hovered over it. The function returns a map, and draws it to the screen.

Creating the Website

I wanted this tool to be user friendly, so I decided that a website would be best since users would not need to download anything to their computer. I used Corey Schafer's excellent [flask tutorial series](#) to design the website with flask. Flask is an easy to use web development library in python that allows you to write python code within an HTML file. I used the CSS stylesheet provided by Corey in the tutorial with a few minor changes. I created a simple home page with two input fields, one for cuisine and one for age group, and a "Generate Map!" button. Below this button is an iframe which displays another html file stored in a separate folder. This iframe contains a default map that has not been colored.

WhereToRestaurant

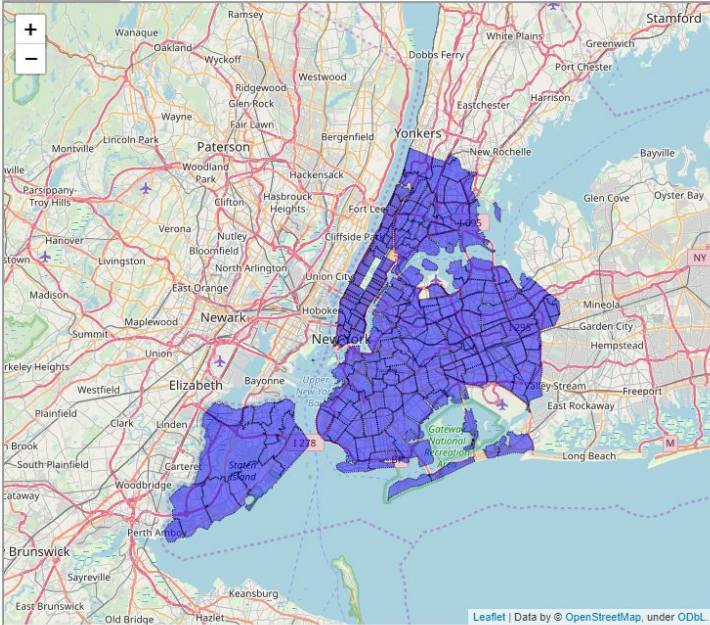
Home About

Make selections

Select cuisine American

Select age group 0-14

Generate Map!



About the Author

Hi my name is Thomas DeGreve
Feel free to contact me!

email: ted091020@gmail.com

linkedin:
<https://www.linkedin.com/in/thomas-degreve-310109164/>

Github:
<https://github.com/ted091020>

When the user presses the “Generate Map!” button it reloads the page while also calling the function that I wrote previously which saves the generated map as an html file to a specified folder. When the page reloads, the iframe is redirected to the new file where the generated map is found.

WhereToRestaurant

HomeAbout

Make selections

Select cuisineAmerican

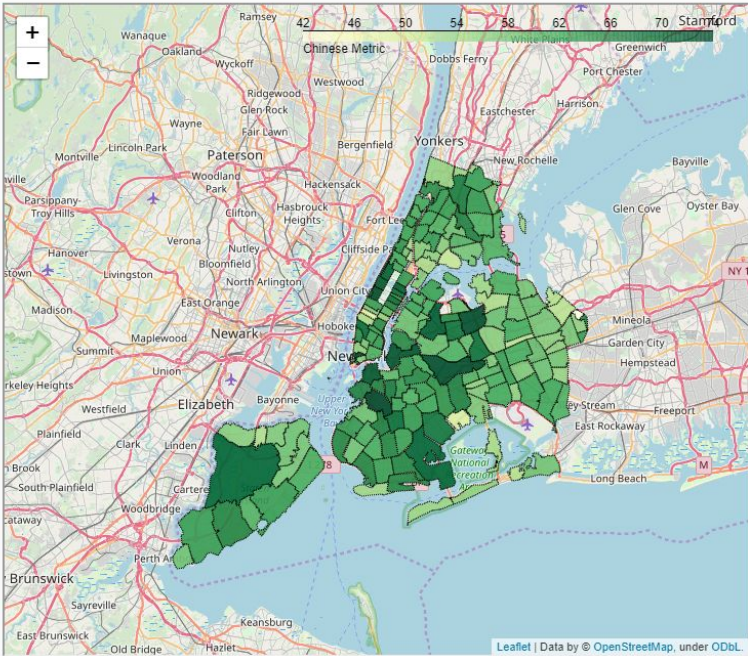
Select age group0-14

Generate Map!

Selections

Cuisine: Chinese

Age group: 30-54



About the Author

Hi my name is Thomas DeGreve
Feel free to contact me!

email: ted091020@gmail.com

linkedin:
<https://www.linkedin.com/in/thomas-degreve-310109164/>


Github:
<https://github.com/ted091020>

Top 10 zip codes

Once a map has been generated, the website informs you which selections were used, and also generates a list of the top 10 zip codes along with corresponding information about each zip code.

WhereToRestaurant

HomeAbout



Leaflet | Data by © OpenStreetMap, under ODbL.

Top 10 zip codes

Rank: 1

Zip Code: 11219

Population: 93979

Percent age 0-14: 30.6

Total Restaurants: 118

Restaurants per Capita: 0.00126

Number of American Restaurants: 3

Median Income: \$37665

Rank: 2

Zip Code: 11368

Population: 114647

Percent age 0-14: 23.1

Total Restaurants: 254

Restaurants per Capita: 0.00222

Number of American Restaurants: 22

Median Income: \$47862

Rank: 3

Zip Code: 11208

Finally, I found the website pythonanywhere.com which does free web hosting and has special functionality that makes it easier for python users. The free version has limited storage size, but because my website is small it works perfect. After changing a bit of the code so that it works with the file directory system of the website, I was able to upload all the files and get the website up and running. I have made several iterations since then such as adding the 'About' page, adding my contact information to the sidebar, and making the website more mobile friendly.

Final Thoughts

I believe this tool can be a useful first step for restaurateurs who wish to narrow down their location search and quickly gain useful information about the areas they are interested in. Importantly, given a dataset equivalent to New York's health inspection data which contains zip codes and cuisine type of all restaurants, this tool could easily be extended to other cities, entire states, or even the entire country. The only limitation is the availability of data. There are companies who mine such data and offer it to customers for a fee, so it is possible to extend this tool dramatically.

Another potential area for improvement would be to speak with actual restaurateurs to better understand the decisions that go into choosing a new location. The opportunity metric could then be improved to give greater insights.