Milestone report: Correlation analysis on spatiotemporal data

Heejong Kim (hk2451)
Cool name pending team
heejong.kim@nyu.edu

Jun Yuan (jy50)
Cool name pending team
junyuan@nyu.edu

Jiin Nam (jn1664)
Cool name pending team
jn1664@nyu.edu

KEYWORDS

Correlation analysis, Mutual information, Spatiotemporal data

1 INTRODUCTION

In this report, we introduce the pipeline of correlation analysis based on the spatiotemporal dataset. Our project follows the steps of data cleaning, ZIP code mapping, correlation calculation, and analysis.

Problem. Correlation analysis plays an important role in data analysis. The information gathered from big data engine provides insights that how we can make our lives better by studying the correlations between the features. For example, based on the correlations between spatio information and the search keywords, Google Flu Trends[1] helped to prevent the spread of flu. Also, you can make sense of the way to the career success by researching on the correlations between the objective/subjective success and a wide range group of working factors[7]. To understand the correlation, we can use correlation coefficients, such as Pearson correlation coefficient and its variants which summarize the association. The designated correlation measure for our team is mutual information. In addition, there are some helpful visualization tools to detect the correlations and outliers among data, like Table Lense[9]. However, with the development of data storage and recording, we are facing challenges of dealing with more and more complex data.

2 METHODS

In terms of dataset, *NYC Open Data* [10] is a good source. So we start our project with the dataset *311 Service Requests from 2010 to Present* [2]. The following parts of this section include the data quality issues we are facing and corresponding solution, as well as the correlation calculation algorithm.

2.1 Data Cleaning

In the 311 dataset, the data quality issues are below:

- *Incompleteness.* In the dataset, some columns have mostly empty properties, such as the column 'Landmark', 'Vehicle Type', and other 10 columns have over 90% cells are null. There are also some cells with values like "None" or "Unspecific". In some cases, like the 'Vehicle Type' we have mentioned, we cannot get the information from other columns, and since the information in the column is so incomplete that cannot contribute useful knowledge to the final correlation result, we could just drop it. However, in some cases, we can restore the value according to the values in other cells. For example, if the record has a missing ZIP code, we can still know the location by the latitude and longitude, and then map the ZIP code to the location. So for the first case, we just cleaned empty feature; while for the later one, we use ZIP code mapping functions to restore the information which is demonstrated in the following section.
- *Incorrectness.* With the increasing of the data volume, there can be some incorrect information in the table. For example, there can be some weird content like '999' for an address. And for the NYC data we used, the ZIP code listed in the column "Incident Zip" includes some numbers which are not valid ZIP code in New York City.
- *Inconsistency.* For the same content, there can be a different format of expression, like "YYYY-MM-DD" and "MM-DD-YYYY" for a date, or "xxxxx" and "xxxxx-xxxx" format for ZIP code. The inconsistency of format could cause wrong results in the end.

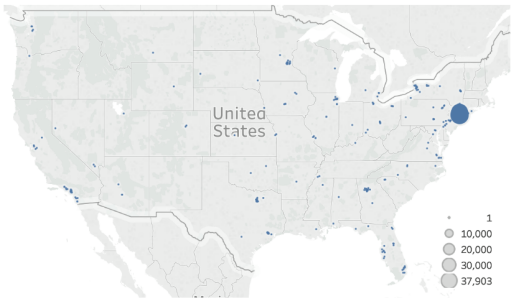

Figure 1: Zip code distribution in the U.S. For 311 dataset used for this report, most of the ZIP codes fall in the New York State. After explicitly deleting some obvious incorrectness like '99999', we still get several outlier ZIP codes which are valid but cannot contribute more to the correlation analysis as the NYC ZIP codes do.

To clean the incomplete data, we listed possible values indicating a missing value, including "None", "Unspecified", "N/A", or just null value. And for each value above, we count the times they appear in each column, if this column has over 90% missing data, we could say this column is just not useful for the correlation because no matter how other feature changes, it stays just empty. And to figure out the incomplete data we can restore later, we check whether all the related information is empty. If the record has empty values in all the ZIP codes, latitude, and longitude fields, it is useful; but either it has a ZIP code or the coordinate information, we can still process it.

To clean the incorrect information, in this stage, we just drop the obvious incorrect record, like the '999' address and "99999" ZIP code, because it cannot contribute to meaningful result. But for some incorrectness not so obvious, like the ZIP code for Washington DC appears for the New York state, we still keep it now. As shown in Fig. 1, we have ZIP codes across the country. For those ZIP codes not in New York States, they are valid and cannot be observed easily from the table because they are real ZIP codes in the U.S. But they

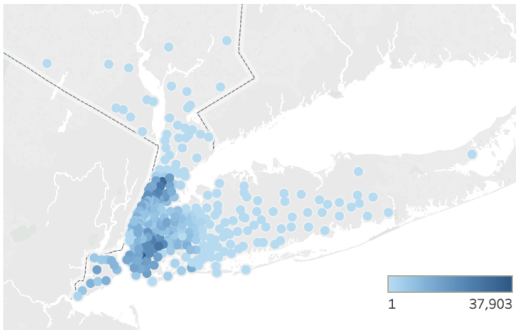

Figure 2: Zip code distribution in New York City. The ZIP codes in New York City have more complete properties in the datasets.

are invalid in a way, because we are dealing with the NYC data and the ZIP codes outside New York City (fig. 1, fig. 2) are just outliers in this case and may cause some incorrectness in the final result.

And when it comes to the inconsistency, we can use some format functions to regulate the format. For example, the date has corresponding datetime formatter function in spark, we just call it and change all the date into the form of "MM/DD/YYYY". And for the ZIP code in the form of "xxxxx-xxxx", we keep the first 5 numbers as the ZIP code.

2.2 ZIP code Mapping

Some of the datasets we are dealing with have only GPS information. We can apply the crossing number algorithm to convert GPS points to ZIP codes to support diverse space resolution for that cases such as NYC Taxi data. The algorithm checks whether if a point is in a polygon or not by counting the number of edges of the polygon meet with a line connecting the point and the other point places out of the polygon.

```
Point_in_Polygon(Point p, Polygon poly)
    //Create line btw p and outside point o_p
    line = createLine(p, o_p)
    //Intersection btw line and the polygon
    num = get_num_intersect(line, poly)
    if num is even
        return true
```

With the given polygon set for ZIP code, the algorithm can allow checking if GPS points are in any of polygons representing ZIP codes. If it is, it converts the points to the mapped ZIP codes. Since the 311 data already has ZIP code attribute, we applied our mapping to NYC taxi data for later use. Table. 1 shows mapping instances from the NYC taxi data.

2.3 Correlation Calculation

Our team's designated correlation measure is "Mutual information". The formal definition of the mutual information is:

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) log(\frac{p(x,y)}{p(x)p(y)}), \quad (1)$$

where $p(x, y)$ is the joint probability function of X and Y and $p(x)$ and $p(y)$ are the marginal probability from X and Y each. X and Y represent each attribute in our analysis. The marginal probability is count number of rows having the same category divided by the number of rows in the attributes. The joint probability is obtained by counting the number of rows having two attribute's categories together.

GPS to ZIP code mapping		
Longitude	Latitude	ZIP code
-73.916661142	40.828848333	10456
-73.784556739	40.697338138	11433
-73.740890627	40.714806463	11429
-73.975517666	40.724369992	10009
-73.873141669	40.753770398	11372
...	...	...

Table 1: Results converting GPS to Zip code using Crossing number algorithm

We used 9 attributes from the 311 data for the correlation calculation in this report. The attributes are 'Agency', 'Borough', 'City', 'Community Board', 'Complaint type', 'Created Date (Aggregated"', 'Incident Zip', 'Park Borough' and 'Status'. We attached the partial code for calculating mutual information. In this example, the script calculates the correlation between "Agency" and "Community Board". We calculated only upper triangle of mutual information correlation matrix for efficiency.

```
attrIn1 = dt.select('Agency').distinct()\
    .rdd.flatMap(lambda x: x).collect()
attrIn2 = dt.select('Community_Board').\
    distinct().rdd.flatMap(lambda x: x).collect()
totalRow = dt.count()
pXY = np.zeros((len(attrIn1), len(attrIn2)))
pX = np.zeros((len(attrIn1),1))
pY = np.zeros((len(attrIn2), 1))
for i in range(0,len(attrIn1)):
        for j in range(i,len(attrIn2)):
        pXY[i][j] = \
        (dt.filter(dt["Agency"]\
                == attrIn1[i]).filter\
             (dt["Community_Board"] \
              == attrIn2[j]).count())/totalRow
for i in range(0,len(attrIn1)):
    pX[i] = (dt.filter(dt["Agency"] == \
                       attrIn1[i]).count())/totalRow
for i in range(0,len(attrIn2)):
    pY[i] = (dt.filter(dt["Community_Board"] == \
                       attrIn2[i]).count())/totalRow
pXpY = pX.dot(pY.transpose())
mask = pXY != 0
MI = np.sum(np.multiply\
               (pXY[mask], np.log(\
                   np.divide(pXY[mask], pXpY[mask]))))
```

3 PRELIMINARY RESULTS

We only analyzed the 311 data collected in 2011 for this report. Fig. 3 shows correlation matrix constructed by calculating the mutual information between attributes. Top 3 attribute pairs with high correlation are: "Agency - Complaint Type", "Borough - Community

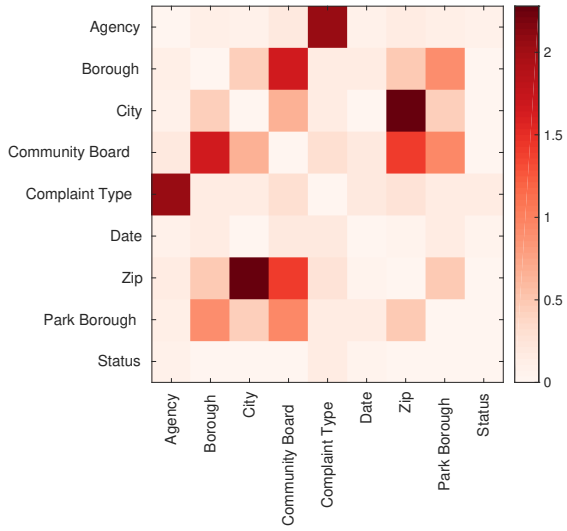

Figure 3: Correlation result between attributes from the 311 dataset in 2011. Each row and column represents attributes.

Board", "City-Incident Zip". We checked who are the categories which make high mutual information value in "Agency - Complaint Type" pair. In Fig. 4, there were some correlation values between categories. The top three pairs are "DOT - Street Condition", "HPD - Heating", "NYPD - Residential Noise". Department of Transportation (DOT) provides the service related to the transportation infrastructure in the City of New York. Rehabilitate and maintain the infrastructure is one of their missions. The relationship between the DOT and street condition complaints makes sense. Department of Housing Preservation and Development (HPD) promotes quality and affordability of housing which makes sense that it has the relationship with heating complaints. New York City Police Department (NYPD) performs a wide variety of public safety. And it corresponds to the relationship with residential noise complaints. The results are plausible in that the complaints shown as pairs are related to the agencies.

Git repository. **Cool name pending** or go to the next url: https://github.com/DS-GA1004/Cool_name_pending.git

4 FUTURE WORK

4.1 Data cleaning aggregation

Spatio information. Even with the cleaning steps that we applied to the 311 data, we still need more transformation. For example, the ZIP code attribute has values like "77092-2016" and "NY". In common sense, a 5-digit integer value such as "11201" is valid. So, we will keep the first 5-digit integer which can repair some invalid ZIP code.

Temporal information. We only considered the date in the format of "DD/MM/YY" in this report. The format for datetime attribute is "MM/DD/YY HH:MM:SS AM/PM" in EST time. We will convert and aggregate it to multiple time resolutions: hour, day, week, month and year.

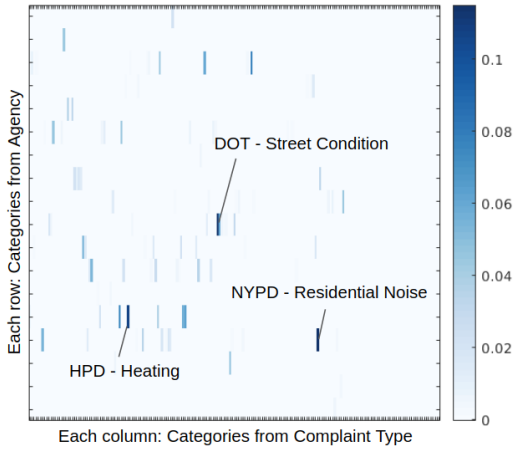

Figure 4: Mutual information matrix between categories of "Agency" attribute and "Complaint type" attribute from the 311 dataset in 2011. Each row and column represents categories.

4.2 Correlation analysis

We will explore more correlation pair between attributes from datasets: NYPD Motor Vehicle Collisions[6], NYC taxi[8], Weather, 311 complaints[2], Citi bike trip[3], Crime[5], Property price, Census data[4]. For further analysis, we expect that there would be relationships between weather and many attributes including collisions, bike trips, and taxi trip data. We also expect a relationship between crimes and property price. And, there might be temporal relationships between attributes in days, time and year.

REFERENCES

[1] Samantha Cook, Corrie Conrad, Ashley L Fowlkes, and Matthew H Mohebbi. 2011. Assessing Google flu trends performance in the United States during the 2009 influenza virus A (H1N1) pandemic. *PloS one* 6, 8 (2011), e23610.

[2] NYC Open Data. 2018. 311 Service Requests from 2010 to Present | NYC Open Data. (2018). https://nycopendata.socrata.com/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9

[3] NYC Open Data. 2018. Citi Bike NYC. (2018). https://www.citibikenyc.com/system-data

[4] :NYC Open Data. 2018. NYC Censes. (2018). http://www1.nyc.gov/site/planning/data-maps/nyc-population/census-2010.page

[5] :NYC Open Data. 2018. NYPD Complaint Data Historic | NYC Open Data. (2018). https://www.citibikenyc.com/system-data

[6] NYC Open Data. 2018. NYPD Motor Vehicle Collisions | NYC Open Data. (2018). https://data.cityofnewyork.us/Public-Safety/NYPD-Motor-Vehicle-Collisions/h9gi-nx95

[7] Thomas WH Ng, Lillian T Eby, Kelly L Sorensen, and Daniel C Feldman. 2005. Predictors of objective and subjective career success: A meta-analysis. *Personnel psychology* 58, 2 (2005), 367–408.

[8] The City of New York. 2018. NYC Taxi Limousine Commission - Trip Record Data. (2018). http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

[9] Ramana Rao and Stuart K Card. 1994. The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 318–322.

[10] NYC Open Data The City of New York. 2017. NYC Open Data. (2017). https://opendata.cityofnewyork.us/