Reveal Hidden Relations in NYC Open Data

https://github.com/hb1500/DS-GA-1004

Hetian Bai
New York University
hb1500@nyu.edu

Jieyu Wang
New York University
jw4937@nyu.edu

Zhiming Guo
New York University
zg758@nyu.edu

ABSTRACT

To unfold the interesting hidden relationships among NYC urban data, it is vital to not only consider the relationship between any two features within a single dataset, but also the relationships cross many datasets. We utilized Spark Map-Reduce to unify time and spacial format as keys and implemented basic data cleaning, aggregation, and filtering non-necessary features out. To group, aggregate, and inner-join datasets by temporal or spatial keys, we took the advantage of Spark SQL to deal with large volumes of data. In this project, we chose mutual information as correlation metric, so we calculated the mutual information of all features in joined datasets, and ranked output mutual information values in order to detect correlations. As the final step of the framework, we visualized features with high mutual information in Python. Through our exportation, we found interesting correlations between temperature (from Weather Dataset) and many features from other datasets such as hourly total number of passengers from Taxi data, daily number of complaints from 311 Service datasets, daily total number of recorded injured cases from Collision datasets, etc.

1 INTRODUCTION

NYC Open Data makes the wealth of public data generated by various New York City agencies and other City organizations available for public use. As of December 2016, there are over 1,600 datasets available on the NYC Open Data Catalog. These datasets cover categories such as Business, City Government, Education, Environment and Health. With a well-designed program, researchers can easily discover relationships between any two features in NYC city datasets. These findings would possibly contribute to city management, disease control, traffic improvement, etc.

For this project, we proposed to use 8 datasets, which are Taxi data, Vehicle Collision, Weather, 311 Service Request, City Bike Trip Histories, NYPD Crime data, Property, Census data. We chose Mutual Information as metric to evaluate correlations. Analyzing these datasets could reveal hidden correlations between different features, and thus provide insights for lots of social challenges faced by New York Citizens such as transportation, resource consumption and public service quality.

2 PROBLEM FORMULATION

In order to compute hidden relationships between features cross multiple datasets, we aimed to compute correlations using mutual information between a pair of features in different datasets. In the first stage, data cleansing and formating will be applied to all datasets. Next, datasets were grouped and combined from two perspectives–spatial and temporal resolution. When grouping, we applied multiple aggregation functions in SQL like COUNT, SUM, AVG, etc. Following that, mutual information will be calculated for each pair of features. Once the program has obtained all the correlations, outputs were ranked by descending mutual information values which made it easy to identify high correlated features. To have further analysis, we proposes possible hypothesis based on the correctional results. To find evidence to support our hypothesis and better present our results, we visualized some those higher mutual information correlations to verify our hypothesis.

We will make sure the reproducibility of the project framework. The framework should contain data pre-processing pipeline, spatial and temporal aggregation, inner join of two datasets, correlation computing, and visualization of relationships between interested features. By the end of the project, all processing code will be posted to GitHub with clear content logic, and embedded with detailed descriptions.

2.1 Datasets Description

2.1.1 Yellow Taxi Data. This dataset includes fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts from 2009 to 2017.

2.1.2 Vehicle Collisions Data. This dataset contains a breakdown of every collision in NYC by location and injury from July 2012 to March 2018. Each record represents a collision in NYC by city, borough, precinct and cross street.

2.1.3 Weather Data. This dataset contains weather information from Year 2011 to Year 2018. It includes fields capturing wind speed, viability and temperature.

2.1.4 311 Service Requests Data. This dataset contains all 311 service requests from 2010 to April 15 2018. It contains fields capturing request date, agency, complaint type, location type, incident zip, incident address, street name, city, community board and borough.

2.1.5 Citi bike Data. This dataset contains Citi bike data from year 2013 to year 2018. It contains fields capturing trip duration, start time and date, stop time and date, start station name, end

Big Data Course Project, May 2018, NYU

station name, station id, station latitude and longitude, bike id, user type, gender and year of birth.

2.1.6 NYPD Complaint Historic Data. This dataset includes all valid felony, misdemeanor, and violation crimes reported to the New York City Police Department (NYPD) from year 2006 to the end of year 2016. It contains fields capturing compliant number, complaint date, compliant time, offense description, borough etc.

2.1.7 Property data. This dataset contains property value in NYC.

2.1.8 Census Data. This dataset contains demographic data and income information.

2.2 Hypothesis

After taking a first glance at features in datasets, we were strongly interested in datasets interaction such as Weather vs Collision, Property vs NYPD Crime, and Weather vs Taxi. Our hypothesis include but not limited to:

Hypothesis 1: Property price in property dataset has negative correlation to crime rate in crime Dataset with respect to zip code. Generally speaking: area with higher property price has lower crime rate.

Hypothesis 2: Features in weather dataset can also impact the decision of passengers to take taxi: people under extreme temperature are more likely to take taxi.

Hypothesis 3: Temperature can also affect the number of complaints from people. Either lower or higher temperature can result arise of the number of complaints.

Hypothesis 4: Features (temperature, wind speed, and visibility) in weather dataset have strong correlations with features such as the total number of types of people (cyclist, pedestrian, motorist) injured in collision dataset. Extreme weather cases have higher potential to lead to collision injuries.

3 RELATED WORKS AND REFERENCES

The main reference for our project is "Data Polygamy: The Many-Many Relationships among Urban Spatio-Temporal Data Sets". (Chirigati, F., Doraiswamy, H., Damoulas, T., & Freire, J, 2015)[2] Data Polygamy, as proposed by this paper, is "a scalable topology-based framework that allows users to query statistically significant relationships between spatio-temporal data sets". Researchers have also performed an experimental evaluation using over 300 spatial-temporal urban data sets which shows that this framework is scalable and effective at identifying interesting relationships.

In the paper "Some data analyses using mutual information" [1], the author analyzed a number of datasets by making use of the concept of mutual information. A concise introduction of mutual information and its properties were introduced. The author states that the analysis is not complete when a large mutual information was found, further analysis and modeling is a necessary to determine causes of the correlations between features. The paper "A mutual information approach to calculating nonlinearity" [3] gives more detailed discussion of mutual information as a comprehensive metric to measure feature dependence or correlation. Mutual information is non-parametric and assumes no sort of underlying distribution or mathematical form of dependence. Therefore mutual information is a superior metric in taking into account both linear and nonlinear dependencies between random variables. However, it does have some limitation such as loss of information using binning transformation, and unstable with binning size.

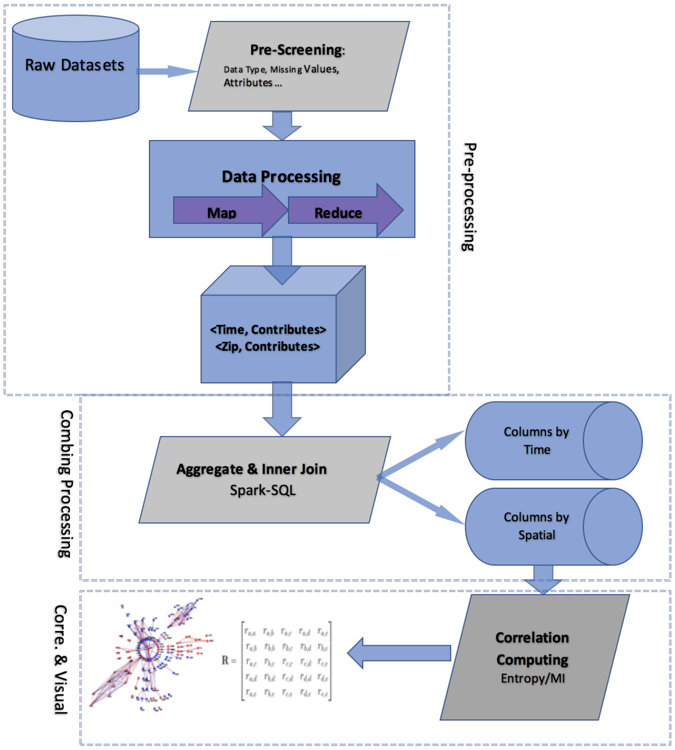


Figure 1: Architecture Streamline

4 METHODS, ARCHITECTURE AND DESIGN

4.1 Architecture and Design

This is the architecture we designed for this project. There are three general procedures which are Data Preprocessing, Datasets Combining Process, and Correlation Calculating & Visualizing. See Figure 1.

4.2 Methodology

4.2.1 ***Data Preprocessing.*** In this step, we use Spark MapReduce to preprocess each dataset. The preprocessing includes data cleaning and pre-aggregation. For data cleaning, we regulate that features will be ignored if there are over 80% of values in that feature is missing. In pre-aggregation, we first need to define our unified MapReduce <KEY> output format to compute correlations between datasets with all possible spatial and temporal resolutions. In this unified MapReduce output format, the key is temporal and spatial data, whereas the value is all the other attributes that contains usable information. Specifically, the unified MapReduce format is defined as a key-value pair as follows:

Mapper Output	Components
Key	YYYYMMDDHH/YYYY/Zip Code
Value	Attribute 1, Attribute 2,..., Attribute n

This is the Detailed information about Components in each Key:

Key Components	Example	Notes
YYYY	2017	year by 4 digits
MM	06	month by 2 digits
DD	16	date by 2 digits
HH	14	hour by 2 digits
DD	16	date by 2 digits
Zip	10004	zip code info.[2]

Notes:

(1) Time-Mark: The finest temporal resolution in this project is hour. This is a mark which represents the temporal resolution for this data entry. For example, if a date-time variable is '2017/08/01/12 : 45', even though the temporal resolution is minute, we still aggregate data by hours. The reason we limit the temporal resolution to hours is that majority of the dataset does not has precision to minutes.

(2) Zip Code: Missing values are replaced by '99999'

After finishing mappers, the output key-value pairs will go to the reducers. Then reducers will perform aggregation based on temporal and spatial units. Here are possible aggregation methods we applied to attributes:

(1) For continuous variable, the aggregation function were MINIMUM, AVERAGE, MEDIAN, MAXIMUM, COUNT, etc.

(2) For categorical features, we count the occurrence of each level. For example, for feature "B" (stands for Borough with levels 1-5) from Property data. We applied SQL function *SUM(IF(B = 2, 1, 0)) AS CNTB_2* to count the number of occurrence of each level.

4.2.2 ***Geographical Information Matching Zip-code - Grid Index Search.*** To match the geographical information to Zip-code, first, we need to convert the shape file of New York City to a series of polygons with the Python package *Shapely* and then conduct the point in polygon test for each pair of longitude and latitude. However, due to the high volume of point in polygon tests and the inefficiency of this test, the runtime for converting a medium amount of geographical data can be extremely time consuming.

In order to solve this issue, we proposed a Python version of grid-index search technique. With the whole list of these polygon information, we first computed the rectangle boundary of New York City. Then with in this large rectangle, we created 1000 × 1000 small rectangles. When checking which polygon a geographical coordinate point belongs to, we first select the small rectangle that this coordinate in. After this, we will conduct the point in polygon tests. However, we only consider the polygons that within the small rectangle. Using this grid-index searching method, we can effectively reduce the average runtime of conducting a single point in polygon test from 75ms to 1ms.

4.2.3 ***Correlation Metrics – Mutual Information.*** We utilized Spark-SQL to inner join any two of datasets by keys: Temporal Key (resolution by Hours or Year), and Spatial Key (Zip Code). Next, we calculated each joined dataset with Mutual Information Function we developed in Python.
Mutual information mathematical expressions are defined as below:

For discrete variable:

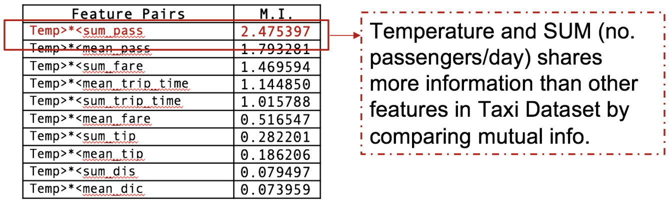
$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) log(\frac{p(x,y)}{p(x)p(y)})$$

For continuous variable:

$$I(X;Y) = \int_{y \in Y} \int_{x \in X} p(x,y) log(\frac{p(x,y)}{p(x)p(y)}) dxdy = H(X) - H(X|Y)$$

Notes: If the two columns we wish to compute are continuous variables, we need to use binning trick. For categorical variable, there is no need to do any transformation or trick.

In our mutual information calculating algorithms, we ranked mutual information values in order to implement feature selection. See Figure 2. Although the numeric value of mutual information cannot be directly interpret as relationship, we can fixed one feature and change the other feature one at a time, then ranked mutual information based on feature A (fixed) vs feature X (X can be B, C, D...). This mutual information rank provides us more insights about which mutual information is relatively high.



Feature Pairs	M.I.
Temp>*<sum_pass	2.475397
Temp>*<mean_pass	1.793281
Temp>*<sum_fare	1.469594
Temp>*<mean_trip_time	1.144850
Temp>*<sum_trip_time	1.015788
Temp>*<mean_fare	0.516547
Temp>*<sum_tip	0.282201
Temp>*<mean_tip	0.186206
Temp>*<sum_dis	0.079497
Temp>*<mean_dic	0.073959

Temperature and SUM (no. passengers/day) shares more information than other features in Taxi Dataset by comparing mutual info.

Figure 2: Feature Selection by MI

4.2.4 ***Data Visualization and Result Analysis.*** Having correlation coefficients, we selected columns with significant and interesting correlations to propose hypothesis that can be used to explain the existence of the high correlation. This was final step. We took advantage use of a data visualization tool called *Plotly* to draw interactive figure so that users can have an deeper insight of the correlations of the two features.

5 RESULTS AND ANALYSIS

After computing mutual information cross all datasets based on different temporal and spatial resolution, we selected some feature pairs with higher mutual information to visualize.

Hypothesis 1: Property Price & NYPD Crime Data Figure 3 indicates the property price level at each Zipcode area. The darker green means a higher house price; figure 4 denotes the total number of crimes at each Zipcode area. The darker indicates a higher number of crime happened. Generally, most area with higher property price suggests lower crime frequency. However, it is not the case for the area in Manhattan where higher price connects with higher crime frequency.

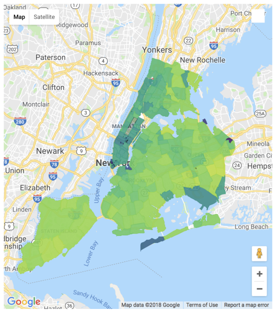


Figure 3: Property Map NYC.

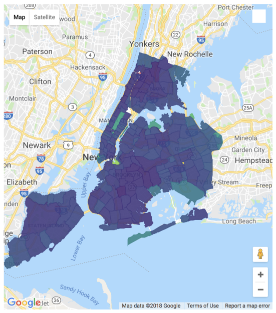


Figure 4: Crime Rate NYC.

Hypothesis 2: Temperature & Taxi Data In figure 5, we detected non-linear relationship between temperature and total number of complaints from people. To have a more clear relationship, temperature range was transformed into integer and distribution of total complaints at each binned temperature is aggregated by average in figure 6, then a stronger non-linearity occurs, suggesting that extreme temperature would cause more complaints. At extreme cold situation, complaints are maximized.

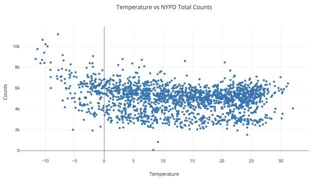


Figure 5: Temperature vs Total Complaints.

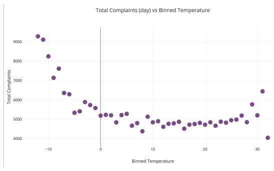


Figure 6: Binned Temperature vs Total Complaints

Hypothesis 3: Temperature & Total Complaints per Day Since the mutual information for temperature vs total number of passengers, and temperature vs mean trip time are relatively high, to check their relationship, in figure 7, three features (temperature,total passengers, and mean trip time) from 2011 to 2017 were plotted. The unit for these features was actually per hour but the figure was zoomed in month level. However, there is no significant pattern from this figure. Similarly, by binning the temperature and using box-plot in figure 8, only some outliers were detected, but no strong relationship could be observed.

Hypothesis 4: Temperature & Reported Injured Cases Figure 9 shows the relationship between temperature (blue curve) and other collision injured counts by hour. It strongly indicates that as temperature goes changes, the number of person injuries, motorist injuries, and cyclist injuries go up and down. One interested finding is that pedestrian injuries inversely changes along with temperature, this is possibly because when temperature is low, for example during snow reason, fewer pedestrian on the street, so the number of injured reported is low compare with warm seasons.

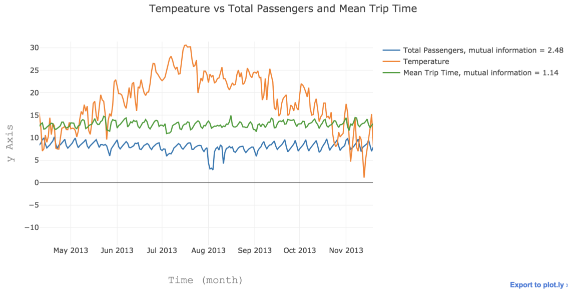


Figure 7: Temperature vs Taxi Information.

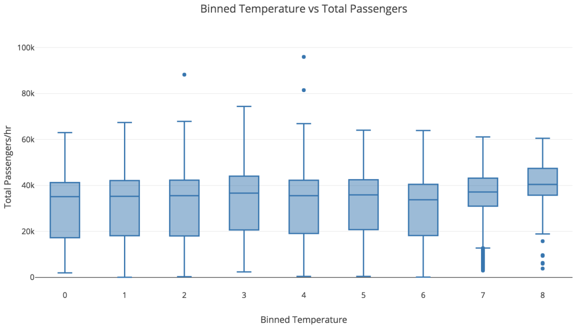


Figure 8: Binned Temperature vs Total Passengers

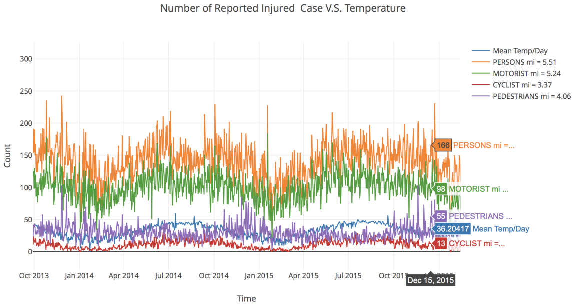


Figure 9: Temperature vs Collision

5. The table below summarizes some other interesting correlations with high mutual information. The mutual information between wind speed (Spd) in weather dataset and average distance in taxi dataset deserves to be explored. Also, the relationship between house value and total complaints by zip-code area might have a strong correlation.

Datasets A <feature A>	Datasets B <feature B>
311 <SUM(complaints)>	Taxi <SUM(passengers)>
311<COUNT(reports)>	NYPD<COUNT(reports)>
Weather<Spd>	TAXI <MEAN(distance)>
Weather<Visd>	TAXI <SUM(trip times)>
Property<MEAN(house value)>	311 <COUNT(complaints)

6 CONCLUSION

Results can be summarized into two main parts: temporal findings and spatial findings. Based on rank mutual information, we select to features to verify the hypothesis. In terms of temporal finding, we found that feature temperature from weather strongly interacting with many other factors including the number of complaints from 311 Service and the number of people injuries in collisions. However, high mutual information for temperature and the number of passengers from taxi data don't have strong pattern by visualization. In terms of spatial findings, we found that the property price is partially correlated with crime rate in NYC and this correlation can be different for different zipcode region.

Though calculating mutual information, we found some limitations of this metric to measure correlations between features. First, we should also keep in mind that a high mutual information coefficient does not guarantee a high correlation, and we cannot infer causality from high mutual information coefficient neither. Second, when involving binning continuous variables, as binning size changes, mutual information will also changes unstably. When both linearity and non-linearity relationship exist, mutual information can be a good choice. When only lienarity relationship exists, then Pearson correlation could be a better choice. Therefore, we suggest to use other mathematical metrics to evaluate correlations together with mutual information as well as using visualization methods.

7 TECHNICAL DEPTH AND INNOVATION

7.1 Technical Difficulties

In this project, we are focusing on using Spark to compute correlations between datasets with different temporal and spatial resolution. Our main technical difficulties are as follows:

(1) **Large Volume of Data**
We have eight large datasets and each of them is processed by different group member. In order to keep the consistency of our processed data, we need to formulate a uniform data processing pipeline that can help us to keep track of our work and thus, improve our productivity and efficiency.

(2) **Inconsistency in Taxi Dataset**
There are 108 (12 months × 9 years)subdatasets in Taxi dataset8. However, the features in these datasets are not exactly same (the number of features and the feature itself might be varied as dataset varies). It costs us some time to figure out the inconsistency and then design different data processing method for taxi datasets.

(3) **Different format of Location and Date**
In order to group our datasets temporally and spatially, we first need to design a uniform template to store date and location information. Since there are multiple datasets with different temporal and spatial resolutions and we need to keep the completeness of the original datasets. Therefore, it is important to get familiar with the structure of datasets and consider all the possible data issues before programming Spark programs.

(4) **Inefficiency of Converting Longitude and Latitude into Zip-code**
To calculate the mutual information coefficient by spatial key, we need to match a pair of longitude and latitude to a zip-code efficiently. If we match geographical information in a exhaustive algorithm, the run time will be too slow to run with large datasets.

7.2 Innovations

The innovation methods we used to solve the above difficulties are as follows:

(1) **Formulate a uniform data processing format**
We formulated a uniform data processing format that can help us to keep track of our work and to aggregate data by key more efficiently.

(2) **Design a Dataset template for cleaning purpose**
We designed a uniform format to store date and location information. For geographical information, we converted it to Zip-code. For date information, we converted it to "Year-MonthDayHour" format.

(3) **Efficiently transfer longitude and latitude into zip-code and neighbor index**
We developed a grid-search algorithm to accelerate the matching process of geographical information to Zip-code. This new algorithm can reduce the run time of point in polygon test from 75 ms to 1 ms.

8 FUTURE WORK

There are three main work worthwhile to perform in future. First, it is necessary to design a data pre-processing package that can automatically transform dirty data to clean data rather than design different data processing function for different dataset. Second, we could transplant grid index algorithm into Spark in order to process location data more efficiently. Last, we could explore more hidden correlations from pairs of features as we know higher mutual information does not necessarily lead us to the right direction.

REFERENCES

[1] David R Brillinger. 2004. Some data analyses using mutual information. *Brazilian Journal of Probability and Statistics* (2004), 163–182.

[2] Fernando Chirigati, Harish Doraiswamy, Theodoros Damoulas, and Juliana Freire. 2016. Data polygamy: the many-many relationships among urban spatio-temporal data sets. In *Proceedings of the 2016 International Conference on Management of Data*. ACM, 1011–1025.

[3] Reginald Smith. 2015. A mutual information approach to calculating nonlinearity. *Stat* 4, 1 (2015), 291–303.