**Ticket Issuers Patterns in NYC**

Timothy Meyers Christopher Fan, Emilie De Longueau, Thibault Duchemin

Contents

[Introduction: 1](#_Toc387853891)

[Data 2](#_Toc387853892)

[Data Collection 2](#_Toc387853893)

[Data Cleansing 3](#_Toc387853894)

[Data Filtering 3](#_Toc387853895)

[Preparing Cluster Analysis 4](#_Toc387853896)

[Feature Selection 6](#_Toc387853897)

[Determining Number of Groups: Hierarchical Analysis 7](#_Toc387853898)

[Generating Clustering Groups: K-Means 10](#_Toc387853899)

[Conclusion of Clustering: Real World Interpretation 12](#_Toc387853900)

[Investigating Geographic Patterns: 14](#_Toc387853901)

[Additional Research: 17](#_Toc387853902)

# Introduction:

The City of New York launched its initiative NYC Open Data, making various datasets available to the public. One of them groups the parking violation tickets issued. We were interested in adopting a new angle to this information: not based on trying to predict if I will get a ticket on a specific area (which has extensively been researched), but to analyze the problem focusing on the parking officers. Responsible of public safety, at the same level as, let’s say firefighters or policemen, very little is actually known about how they do their job. IMDB database has about hundred times more title about police (9693) or firefighters (1200) compared to parking officers (70).

Our analysis of the dataset identified an issuer that issued 7948 tickets in around 5 months, so 79 tickets per day. The highest performance was even above XX tickets for a day. When using an average estimation of $65 for a ticket (conservative), we estimate that the parking officer generated more than half a million dollar to the city. And this is for one person. How do they do that?

We wanted to understand whether of not the officers employed specific strategies to have such a high number of tickets, especially in environment when one would think the ticket distribution is random. Did they have information about specific area with high violation rate? Did they focus on a certain kind of violation? When did they, and what made they perform so well?

The patterns we uncovered allowed us to distinguish among parking officers some top performers, who presented characteristic behaviors. We present here a description of our approach to identify the top performers, the key specificities that separate them from the “average” parking officer, and the clustering analysis we conducted to infer the officers’ patterns. Finally, we conclude in summarizing some of our key findings and providing a user-friendly way to explore on the NYC data the

# Data

## Data Collection

Our analysis is based off the Parking Violations Issued dataset found on the NYC Open Data website. NYC Open Data provides over one thousand data sets generated by New York City agencies and other City organizations available for public use. For our project, we focused on the Parking Violations Issued dataset. The dataset is comprised of over 6.5 million rows spanning 43 columns, where each row is a parking violation issued by a parking officer primarily between July 2013 and today, May 2014. The dataset can be found here:

<https://data.cityofnewyork.us/City-Government/Parking-Violations-Issued/jt7v-77mi>

In preparation of our analysis, we decided to import the data, originally in a CSV file, into a PostgreSQL database. This would enable us to easily rollup or pivot the data based on the needs of our analysis, while achieving great performance despite the large size of the dataset.

## Data Cleansing

After creating a PSQL database table and importing our CSV data into it, our next task was to generate descriptive statistics. This required us to ensure that the columns most pertinent to our analysis didn’t contain “junk values”. For example, the column describing the date of the issued violation had dates beginning as early as 1970, and as far into the future as 2069. However, a quick scan of the data showed that the vast majority of tickets were distributed between 2013 and 2014. To simplify our analysis, we discarded rows outside of that date range.

Similarly, the column describing the time of the issued violation had invalid entries. All values were in the format HHMMA or HHMMP, where HH is the hour of day between 01 and 12 including, MM is the minute of the day between 00 and 59 including, and the final digit is either A or P, depending on whether the ticket was issued in the morning or afternoon. However, we found many entries with invalid times, such as 3012A, or 0290P. We deleted all rows from the database with a time value outside of the above specifications.

## Data Filtering

After cleansing the data, we still found problems that prompted us to perform a round of filtering. More than half a million rows were missing unique identifiers for parking officers, i.e. were equal to 0, which prevented us from generating statistics about their performance issuing tickets. We deleted these rows to keep only violations that could be attributed to an identifiable officer.

# Preparing Cluster Analysis

After this first overview on top performers characteristics through descriptive statistics, we wanted to go further by doing a clustering analysis on the 174 top performers selected. By conducting this analysis, we intended to determine specific patterns and distinct behaviors among best meter maids.

The first step was defining the feature vector to use. Selecting relevant features to run a clustering algorithm is a crucial step: it’s one of the most fundamental problems in the field of machine learning. And it is especially hard in unsupervised learning as we are not given class labels, so we don’t know in advance which ones are relevant, redundant, and if some can even misguide clustering results. Selecting too many features can also make clustering results hard to interpret.

In our case we made the assumptions that the **day type**, the **time of the day** and the **ticket violation type** would be relevant features to distinguish different clusters (personas) among best meter maids. Indeed, most features in the dataset had non-understandable values, or were badly informed, which considerably reduced our choice for feature selection.

After a deep cleaning of the data detailed in the previous section, we ran SQL queries (generate\_issuer\_aggregates.sql) on performers to prepare the selected feature vectors. The format of the csv file obtained had five columns: issuer\_code, type\_day, day\_period, violation\_code, ticket\_count. One raw represented the number of tickets issued for one best issuer, and for a specific value combination of the 3 selected features (e.g. Monday/6-7am/violation#3). Hence for each best issuer\_code, we had the same number of raws equal to the number of possible value combinations of the day type, time of the day and ticket violation.

For our clustering analysis, we defined to use value combinations of these 3 features as final cluster features. To obtain the final files with one raw per best issuer, we used a Python script than converted the initial file into final csv file to run in SPSS software.

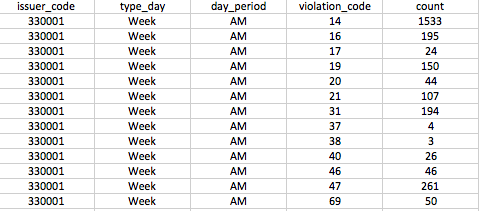


Figure 1: Initial file obtained from SQL queries

**Python script (ClusterDictionaryGeneral.py)**

**2x2x15 = 60 combined features**

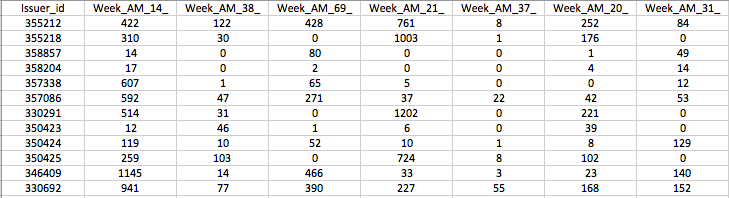


Figure 2: Final file with combined features for clustering analysis

The final file contains a\*b\*c combined features, with a, b, c respectively the total number of values for type\_day, day\_period and violation\_code.

## Feature Selection

We tested different feature combinations until we found the most insightful clusters. Details about each step are given below:

* Feature set 1: **hour\_of\_day/violation\_category**

The final features were a combination of a one-hour slot (between 6 am and 6 pm, hence 12 possible slots) and a violation\_category (among the 6 categories defined previously). The clustering was run with these 12\*6 = 72 combined features.

But cluster results were not satisfying with this selection:

* The violation categories were too broad and didn’t lead to discover insightful patterns among clusters.
* There was too much granularity using one-hour slots. We detected many redundancies in the clusters and figured out that using a lower level of detail would bring less complexity while giving more insightful results.
* We had no information on the type of day and thought it would be useful to integrate it in a second feature set.
* Feature set 2: **day\_period/day\_of\_week/violation\_code**

The final features were a combination of the day period (AM or PM, hence 2 possible choices), a violation\_code (among the only 15 most represented in the dataset), and the day of the week (from Monday to Sunday, hence 7 choices). The clustering was run with these 2\*15\*7 = 210 combined features.

We got better results and more insights by adding the day of week in our analysis. But:

* For the patterns detected per cluster group, the day of week showed many repetitions
* Having a large number of features (210) made the patterns more complex to detect
* Feature set 3: **day\_period/period\_of\_week/violation\_code**

The final features were a combination of the day period (AM or PM, hence 2 possible choices), a violation\_code (among the only 15 most represented in the dataset), and the period of the week (Week or Weekend, hence 2 choices). The clustering was run with these 2\*15\*2 = 60 combined features.

This feature selection was the best among all tested in terms of pattern discovery. It gave the most insights on different types of personas among top performers, and we used this one for all the conclusions in this report.

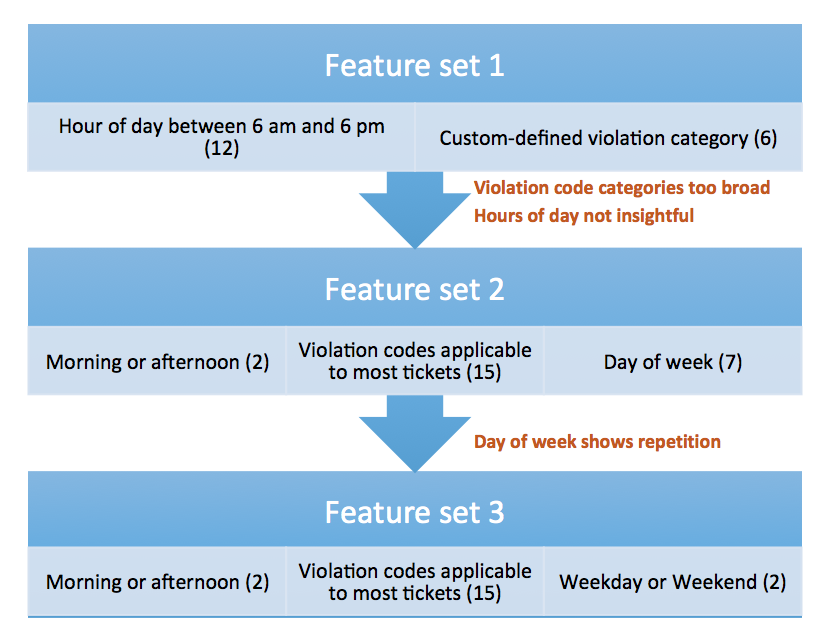


Figure 3: Steps followed for clustering feature selection

## Determining Number of Groups: Hierarchical Analysis

We used SPSS v22 to conduct the Cluster Analysis.

|  |  |
| --- | --- |
| **Step 1: Import Data** | * Using SPSS, import the csv file “Cluster\_Feature\_Vectors\_top.csv” * File Composition:   + Rows: 173 Rows of unique Issuer IDs   + Columns:     - Issuer\_ID: Unique ID Label to distinguish rows     - 60 Feature Columns: All scalar numerals   + Value: Each number in the feature vector represents the aggregated total number of tickets issued during the given calendar time period for the given feature |
| **Step 2: Define Hierarchical Analysis** | * **Agglomerative:** We determined that an agglomerative method would best allow us to determine an appropriate number of cluster groups for the top performers via a dendrogram. * **Non-Normalize:** We chose not to normalize the data because all of the feature variables have the same similar scale on the dimensions. We wanted to preserve the relative size of more popular features versus smaller ones to emphasize dis-similarity. * **Linkage Rules:** We chose a Ward’s Method as a criterion method to determine clustering because all of the feature variables are scalar numbers and do not have binary numbers involved. * **Distance:** We chose a squared Euclidian distance formula to give more progressively greater weight on clusters that are further apart |
| **Step 3: Output** | * SPSS Generates the following output: Cluster\_Analysis\_Kmeans\_Agglomerative.xlsx   + Proximity Matrix: Calculates the distance between each feature/row entry   + Agglomeration Schedule: Determines the ordering of which rows are added to each other to create the cluster. It lists the nodes combined and the distance coefficients   + Dendrogram: Visual representation of the clustering layered on a scaled distance |
| **Step 4:**  By plotting the agglomeration schedule one can see that there is a noticeable elbow curve within the last clusters. | Elbow |
| **Interpreting Dendrogram:**  Using the dendrogram we spotted 5 potential clusters.    3  5  4  2  1 | |

## Generating Clustering Groups: K-Means

We used SPSS v22 to conduct the Cluster Analysis.

|  |  |
| --- | --- |
| **Step 1: Import Data** | * Using SPSS, import the csv file “Cluster\_Feature\_Vectors\_top.csv” * File Composition:   + Rows: 173 Rows of unique Issuer IDs   + Columns:     - Issuer\_ID: Unique ID Label to distinguish rows     - 60 Feature Columns: All scalar numerals   + Value: Each number in the feature vector represents the aggregated total number of tickets issued during the given calendar time period for the given feature |
| **Step 2: Define K Means** | * **Data Selection:** We choose all 60 features and used “issuer\_id” as the label * **Number of Cluster:** Using the insight from the Hierarchical clustering we chose a cluster count of 5 * **Defined Iteration Count:** 40 |
| **Step 3: Output** | SPSS Generates the following output:   * Cluster Membership: The assignment of each of the 173 issuer\_ids to one of the 5 clusters. It also has the node distance from the centroid. * Iteration History: Shows each of the K-means convergence steps. Running the k-means a couple of times, the five clusters consistently converged within less than 10 iterations. * Anova Table: The analysis for each of the 60 feature contribution to the overall variance. * Final Centroid Totals (“kmeans5\_3Feature\_Final”): For each of the 5 clusters, outputs the feature total |
| **Step 4: Interpretation** | Running the k-means a couple of times, the five clusters consistently converged within less than 10 iterations.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Iteration | Change in Cluster Centers | | | | | | 1 | 2 | 3 | 4 | 5 | | 1 | 1121.972 | 1128.145 | 1304.358 | 1220.947 | 1122.462 | | 2 | 144.968 | 155.421 | 95.306 | 43.900 | 33.288 | | 3 | 37.132 | 133.567 | 48.715 | 100.735 | 0.000 | | 4 | 41.536 | 50.871 | 31.960 | 18.242 | 0.000 | | 5 | 56.208 | 50.386 | 19.555 | 43.208 | 62.191 | | 6 | 0.000 | 24.783 | 18.367 | 24.477 | 0.000 | | 7 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | |
| **Step 4: Interpretation** | The Centroid output reveals the aggregate total tickets for each cluster. To improve comparability, we converted each centroid total into the percentage of the Centroid aggregate total. We color coded the entries to reveal which features contributed the most to the overall number of tickets. This generated the corresponding color table which clear distinguishes differences between the five clusters.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **Cluster Centroid % of Total** | |  |  |  | |  | **1** | **2** | **3** | **4** | **5** | | Week\_AM\_14 | 8.89% | 1.90% | 10.64% | 2.71% | 37.59% | | Week\_AM\_38 | 2.56% | 2.28% | 5.76% | 0.78% | 0.31% | | Week\_AM\_69 | 0.17% | 0.50% | 5.94% | 3.16% | 7.38% | | Week\_AM\_21 | 39.41% | 1.57% | 15.74% | 1.02% | 1.13% | | Week\_AM\_37 | 0.26% | 1.63% | 2.95% | 0.37% | 0.11% | | Week\_AM\_20 | 6.88% | 0.68% | 5.09% | 0.13% | 1.22% | | Week\_AM\_31 | 0.07% | 0.18% | 1.78% | 0.81% | 8.96% | | Week\_AM\_16 | 2.39% | 1.54% | 7.51% | 0.10% | 2.52% | | Week\_AM\_46 | 2.65% | 0.50% | 2.77% | 0.21% | 3.17% | | Week\_AM\_40 | 6.48% | 0.39% | 2.95% | 0.21% | 2.04% | | Week\_AM\_47 | 0.00% | 0.03% | 4.19% | 1.20% | 6.16% | | Week\_AM\_19 | 2.72% | 0.27% | 2.11% | 0.29% | 2.63% | | Week\_AM\_42 | 0.02% | 0.15% | 0.54% | 0.60% | 0.62% | | Week\_AM\_71 | 2.25% | 0.39% | 1.75% | 0.29% | 0.57% | | Week\_AM\_17 | 0.65% | 0.30% | 1.39% | 0.23% | 0.99% | | Week\_PM\_14 | 0.76% | 10.23% | 1.54% | 19.26% | 4.01% | | Week\_PM\_38 | 0.69% | 11.65% | 2.11% | 3.50% | 0.06% | | Week\_PM\_69 | 0.02% | 2.25% | 2.02% | 11.59% | 0.90% | | Week\_PM\_21 | 0.79% | 0.24% | 0.21% | 0.08% | 0.03% | | Week\_PM\_37 | 0.29% | 9.75% | 2.20% | 3.24% | 0.06% | | Week\_PM\_20 | 0.43% | 8.42% | 1.42% | 1.36% | 0.48% | | Week\_PM\_31 | 0.02% | 2.31% | 0.93% | 16.52% | 1.38% | | Week\_PM\_16 | 0.41% | 7.20% | 1.06% | 1.33% | 0.96% | | Week\_PM\_46 | 0.91% | 4.80% | 0.42% | 2.11% | 0.99% | | Week\_PM\_40 | 0.48% | 4.36% | 0.54% | 0.99% | 0.54% | | Week\_PM\_47 | 0.00% | 0.15% | 0.69% | 4.41% | 0.45% | | Week\_PM\_19 | 0.53% | 2.16% | 0.42% | 1.36% | 0.40% | | Week\_PM\_42 | 0.00% | 1.04% | 0.87% | 6.60% | 0.59% |   The greater the green color, the larger contribution the specific row feature had to the total aggregate tickets in the cluster. |
| **Interpretation** | The highlighted green features were also reinforced within the ANOVA table as contributing the highest to the overall variations.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Feature** | **Mean Square** | **df** | **Mean Square2** | **df3** | **F** | **Sig.** | | Week\_AM\_21 | 11703701.23 | 4 | 62317.68 | 168 | 187.807 | 0 | | Week\_AM\_14 | 6149960.361 | 4 | 48930.274 | 168 | 125.688 | 0 | | Week\_PM\_14 | 3201941.114 | 4 | 53693.625 | 168 | 59.634 | 0 | | Week\_PM\_31 | 2765263.061 | 4 | 35097.749 | 168 | 78.787 | 0 | | Week\_PM\_69 | 1226829.305 | 4 | 27064.219 | 168 | 45.33 | 0 | | Week\_PM\_38 | 950459.981 | 4 | 36947.14 | 168 | 25.725 | 0 | | Week\_PM\_37 | 658201.412 | 4 | 38422.366 | 168 | 17.131 | 0 | | Week\_PM\_20 | 527691.37 | 4 | 17377.61 | 168 | 30.366 | 0 | |

# Conclusion of Clustering: Real World Interpretation

Using the significant features from the K-Means clustering, we were able to create personas for each of the 5 clusters. You can reference the “ViolationCodes\_match” tab in the excel file.

|  |  |  |
| --- | --- | --- |
| **Persona Name** | **Description of Behavior** | **Feature Importance** |
| **Butch ‘The Cleaner’**  **the-cleaner-01.jpg** | Street-cleaning: Meter people who issue mainly street cleaning tickets  (12.14% of issuers)  -> Targets street cleaning: 68% of them  -> Favorite time: the morning | Cluster 1:  “Week\_am\_21” = 40% of centroid totals.  Violation 21 = Street Cleaning: No parking where parking is not allowed by sign, street marking or traffic control device. |
| **Curbside Terror’ Terry**  **ticket.jpg** | Parking officers who issue expired-meter, general no parking, and general curbside parking tickets/ Top weekend performers.  (27.75% of issuers)  -> Knows where expired meters are  -> Performs best the weekends  -> Afternoon & the weekends | Cluster 2:  Both “Week\_PM\_38” , “Week\_PM\_37” features were the most prominent of all the clusters.  Violation Code 37: Parking in excess of the allowed time  Violation Code 38: Failing to show a receipt or tag in the windshield |
| **‘Early-Birdie’ Rose**  **187640_1.jpg** | Morning generalists: Street-cleaning, general no standing or parking, time  violation in zone, fail to show receipt, parked in truck unloading zone, etc.  during weekdays (24.86% of issuers)  -> Gets all her stuff done the morning  -> Multi-ticketer: load, receipts, time, parking  -> Work in majority weekdays | Cluster 3:  This was the most unremarkable cluster with broad distributions of tickets across code in the morning. |
| **Shawn ‘The Shield’**  **1327339.jpg** | Afternoon Commercial enforcers: No parking or standing, truck unloading  zone, and commercial metered zone tickets during weekdays.  (24.28% of issuers)  -> Focus mainly on commercial zones  -> Enforce only the afternoon  -> Works on the weekdays. (he’s busy on the weekends) | Cluster 4:  Highest contribution in “Week\_PM\_31” and “Week\_PM\_69”.  Code 31: Standing of a non-commercial vehicle in a commercial metered zone.  Code 69: Failing to show a muni-meter receipt, commercial meter zone. |
| **‘Mafioso’ Benetto**  **Jennifer+Garner+Getting+Parking+Ticket+Brentwood+p_U5ntVrcgEl.jpg** | No standing in weekday mornings: Meter people who issue “General No  Standing” tickets during weekday mornings.  (10.98% of issuers).  -> Operates in high-traffic areas  -> Location in center of the city mainly to easy the morning deliveries  -> Protects commercial zones from unwanted parking | Cluster 5:  “Week\_AM\_14”: Had the most significant feature contribution.  Code 14: General No Standing: Standing or parking where standing is not allowed by sign, street marking or; traffic control device |

# Investigating Geographic Patterns:

Using the K-Means analysis, the top issuers can be assigned to a cluster and ranked by distance from the centroid.

The next step is to map issuers that are closest to the cluster centroids to see explain geographic details that can explain the behavior patterns observed in the clustering analysis.

Methods:

* The issued tickets for select issuers was geographically mapped from 9/1/2013 to 9/30/2013.
* [Google Fusion Tables](https://sites.google.com/site/fusiontablestalks/stories) were used for geo-coding services
* Google Maps were used to render maps
* Code: Folder: Finalsubmissioncode/Ticket Map/showmap.html
* Website Implementation: <http://people.ischool.berkeley.edu/~chrisfan/Maps/maps_fusion_table.html>

|  |  |
| --- | --- |
| **Cluster 1: Butch ‘The Cleaner’**   * “Cleaner” and “Cleaner2” * One of the reasons why “cleaners” are so effective is that they can issue a high number of tickets in a relatively concentrated geographic density | Figure : "Cleaner" Issuer 345221  Figure 5 Cleaner2 Issuer "350433" |
| **Cluster 2: Curbside Terror’ Terry**   * “Curbside”: Issuer 354084 * These issuers target streets with characteristics such as having famous tourist zones such as Broadway | Tourist Zones: Broadway |
| **Cluster 4: Shawn ‘The Shield’**   * **“Shield”: Issuer 358570** has a distinctive pattern targeting specific streets * Shields target strategic streets with known business commercial activity * Notable Streets:   + W14th   + W 48th: Near Rockefeller Center   + 3rd: |  |
| **Cluster 5: Benneto:**   * **Issuer: 346330** * The high number of “standing” type violations can be attributed to the strategic locations between major connection choke points connecting the island of Manhattan. * For example, Hells Kitchen is both near the Lincoln Tunnel which connects Manhattan to New Jersey * Also the area has some key transportation characteristics like the Port Authority Bus Terminal and the New York Passenger Ship Terminal |  |

# Additional Research:

The nature of the exploratory research was to determine whether there are observable heuristics in which the ticket issuers use to determine their routes. Additional research would be to connect the nature of the routes to other characteristics.

* **Ticket Revenue Maximization:** The analysis only looked at the aggregate number of tickets issued per issuer. Another analysis would be to connect ticket prices to issued tickets to determine if there could be ways to be a top performer with regards to revenue generated
* **Traffic Flow Correlation:** The analysis did not factor into account the effect of the traffic flow to aggregate tickets issued.
* **Ticket Coincidence Probability:** Using the issued ticket street location and time, analysis could be conducted to determine the percentage likelihood of geographic coverage for a given day to determine the probability of getting issued a ticket for a given location.