Assessing the Livability of NYC Neighborhoods

Aydin Abiar aa9380@nyu.edu
Sohail Hodarkar sph8686@nyu.edu
Monica Thirunavukkarasu mt4705@nyu.edu
Thomas Wong tsw8626@nyu.edu

Abstract

Livability is a very subjective term, and the likelihood of an area being described as livable is heavily influenced by the expectations of the individual(s) evaluating it. In order to disambiguate this, our study begins by describing one potential definition of livability and subsequently proceeds to rank several neighborhoods of New York City based on this definition. We also analyze the contribution of different factors towards the overall livability of a particular locality. Besides serving as a tool for validating ones expectations from a specific neighborhood, our system can also be used by city-wide administrative institutions which are constantly looking to enhance certain aspects of NYC. Our findings, which are based on comprehensive analytics of large volumes of data, suggest that specific areas in Manhattan, and a few of them in Queens and Brooklyn, are amongst the most-livable in the city.

1 Introduction

As not only one of the most popular cities in the world, but also the most populated, New York City is home to approximately 8.8 million people. Spanning 302.6 mi^2, the city is comprised of 5 boroughs- The Bronx, Brooklyn, Manhattan, Queens, and Staten Island. With possibly more people moving to NYC as we write this report, finding a "livable" neighborhood for themselves would certainly not be trivial. Being such a diverse city, it is natural that the answer to "which is the most livable neighborhood" would fetch uncountable answers. However, our idea was to provide concrete data-informed answers to such a question. Having done it ourselves, we resonate with the fact that moving to a new city is overwhelming. Validating one's expectations from an identified neighborhood can make the process a lot more seamless, thereby better-preparing the individual.

Driven by that ideology, our motivation behind this project was to create a system capable of helping people make data-backed decisions when choosing a neighborhood to reside in. To this end, we cleaned, profiled, processed, and eventually analyzed a plethora of disparate datasets related to New York City. As far as relevance is concerned, we believe that this tool could be used by those looking to relocate to/within NYC, city-level Government Administrations looking to analyze and improve some aspects of the city. We believe that the largest class of beneficiaries of a system like ours, would be the current and potential residents of The "Big Apple".

The remainder of the report is self-contained, and is organized as follows: Section 2 defines the term "livability" in context of this study, Section 3 touches upon the technologies leveraged by our system, Section 4 gives a quick idea of the datasets used and the process of gathering, cleaning, profiling, and scoring them, Section 5 depicts our design diagram, Section 6 dives into the analytical insights and results of our experiments, and our conclusions are listed in Section 7.

2 Livability, As We See It

In order to proceed with a uniform definition of the term, we define livability to be an amalgamation of various factors, which shall now be described. It is paramount that a resident of neighborhood feels, and indeed is, safe at all given times. Therefore, safety was the factor on our list. The health of a neighborhood's residents is an extremely important factor, and is heavily influenced by the hygiene conditions around them. Needless to say, health & hygiene was next on the list. It is important that residents find their accommodation comfortable and ambient, as it is a central aspect of ones life. Thus, the quality of housing found its way into our consideration. A neighborhood should be well connected to other parts of the city. With that in mind, we decided to include the ease of transportation as a factor. Finally, residents would almost certainly want to socialize and could end up meeting over meals at different restaurants. It is for this reason, that we take the nearby restaurant ratings into account when conducting our study.

Having settled on these five key agents, it remains to be seen what data was used to gather insights into them, which is exactly what section 4 covers. At this point, it is also important to point to the fact that we identify neighborhoods by their zip-codes, and shall henceforth, we shall use the terms interchangeably.

3 Technologies

We now provide a brief overview of the distributed, and highly-scalable systems which serve as the backbone

of our system:

Hadoop Distributed File System. Abbreviated as HDFS, this is a distributed, highly-available, highly fault-tolerant file system designed to run on commodity hardware [Shv+10]. It is used to store large data sets in a distributed (by blocks) manner across multiple nodes in a cluster (NYU Dataproc, in our case). It is optimized for streaming reads of large files and provides high-throughput access to application data, and is designed to be extremely scalable thereby allowing efficient storage of large files. In addition, HDFS is also extremely capable of handling hardware failures due to its block-replication mechanism. It is commonly used in large enterprise environments to store large amounts of data for applications such as data analytics, machine learning, data warehousing, and distributed storage.

Given that the data used by our project was cumulatively around 15GB in size and needed to support distributed processing, it served as a perfect use case for HDFS.

MapReduce. MapReduce is a programming model developed by Google to process large datasets in a distributed computing environment [DG04]. It is based on the idea of breaking down a large problem into smaller tasks that can be run in parallel on multiple servers. The Map Reduce model consists of two phases: Map and Reduce. In the Map phase, the data is split into small chunks and each chunk is processed independently of the others. In the Reduce phase, the results from the Map phase are combined and processed to get the desired output. It is well-suited for processing extremely large datasets that cannot be handled on a single machine, thereby perfectly fitting the Big Data prototype. It can be used for tasks such as sorting data, calculating statistics, or searching for specific data entries. Map Reduce is being used in many big data projects to process large volumes of data efficiently.

We used MapReduce for cleaning, profiling, and processing the datasets at hand.

Hive and Trino/Presto. Hive is a batch-oriented data warehousing system that runs atop HDFS and MapReduce, and enables the analysts to query the stored data using a SQL-like syntax. Trino, formerly known as Presto, is an open-source platform for performing distributed analytics using SQL queries. Given that it does not make use of the Hive runtime, which converts most user-queries into MapReduce jobs, it is designed to make data processing much more efficient and quicker [Set+] with response-times measuring a few milliseconds.

Since our analytics phase, which will be described later, involved running projection, ordering, and joining queries on voluminous datasets, we utilized Hive as a warehouse and carried out the analysis using Trino/Presto with a Hive connector.

Tableau. Tableau is an extremely popular data visualization software, that supports interactions with Hive tables. Given that we used Hive for warehousing our large datasets, and due to the fact that they provide free student licenses, the use of Tableau for visualizing our results was seamlessly easy.

4 Data: Gathering, Cleaning, Profiling, and Scoring

Before proceeding to describe the datasets used, and the stages of gathering, cleaning, profiling, and scoring them, it is important to recall the factors used to define livability in section 2. They are, safety, health & hygiene, quality of housing, ease of transportation, and nearby restaurant ratings. We now proceed to explain the data, and the steps taken to prepare it for the analytics phase, for each of the aforementioned factors.

Safety. In order to assess this aspect, we utilized two datasets- NYC Complaint Data and NYC Shooting Data.

NYC Complaint Data

Gathering: This dataset contained information about all valid felony, misdemeanor, and violation crimes reported to the New York Police Department (NYPD). It contained 7.83 million records, was publicly available with NYC Open Data, and occupied around 2.33 GB of storage.

Cleaning: The dataset contained a total of 35 columns, out of which the ones relevant for our study were complaint number, Date of reporting, Violation type, Borough, and Latitude-Longitude. They were retrieved using a MapReduce job where the mapper emitted (complaint number, relevant features) and the reducer helped in emitting a single record per complaint number. As evident, this dataset did not have a column for zip-codes. This was where the Subway Stations (which would be described later) dataset turned out to be useful. Each shooting incident was mapped to the zip-code of the "closest" subway station. This, too, was done using a MapReduce job that resembled a replicated-join and a top-k design pattern.

Profiling: Profiling the dataset implied gathering information about complaints of crime in every zip-code. In order to achieve this, a MapReduce job was run where the mapper emitted (zip code, 1) for every record, and reducer then emitted (zip code, sum). A combiner (same as the reducer) was also used.

Scoring: In order to be able to eventually rank neighborhoods, we had to engineer a score on the 0-100 scale for each neighborhood. This was achieved using a MapReduce job, that intuitively applied the following

formula for each zip-code:

$$\text{Score(Zip)} = 100 - ((\text{Count in Zip}/\text{Total Count}) * 100)$$

NYC Shooting Data

Gathering: This dataset consisted of information about shooting incidents reported across New York City. It was publicly available NYC Open Data, had 63.3 thousand records, and occupied 5.2 MB of storage.

Cleaning: It consisted of 19 columns, out of which the ones relevant for our study were Incident ID, Date of Occurrence, Borough, Murder (boolean flag), and Latitude-Longitude. They were retrieved using a MapReduce job where the mapper emitted (incident ID, relevant features) and the reducer helped in emitting a single record per incident ID. Similar to its predecessor, this dataset, too, did not have a column for zip-codes. This was where the Subway Stations dataset turned out to be useful, yet again as each shooting incident was mapped to the zip code of the "closest" subway station.

Profiling: Profiling the NYC Shooting Data followed the exact same steps as that of NYC Complaint Data.

Scoring: The process of scoring the NYC Shooting Data was identical to that of NYC Complaint Data.

With that, we now had two independent datasets which had to be correlated. Our approach to this, was to execute a reduce-side full-outer join on the zip-column, and average the two scores. Should a zip-code not appear in either of the dataset, it was treated to have the highest possible score in that. At the end of this step, we had one dataset that reflected the safety of each neighborhood as quantified using the average score.

Health & Hygiene. In order to assess these aspects, we referred to four main datasets - ER Visits & Admissions for Influenza-like/Pneumonia Cases, Daily Park Cleaning, Rodent Inspection, and Parks Inspection.

ER Visits & Admissions for Influenza-Like/Pneumonia Cases

Gathering: This dataset contained information of emergency room visits for influenza and/or pneumonia-like cases in the city, and was publicly available on NYC Open Data. The total size was 9.43 MB.

Cleaning: To clean this dataset, we needed to filter on the latest "extract date" column to remove duplicate rows. Including duplicate rows would cause incorrect calculation of scores described below.

Profiling: Once again, MapReduce code was written to gather statistics about each zip-code. We determine which zip-code has the highest number of cases, which zip-code has the lowest number of cases, etc.

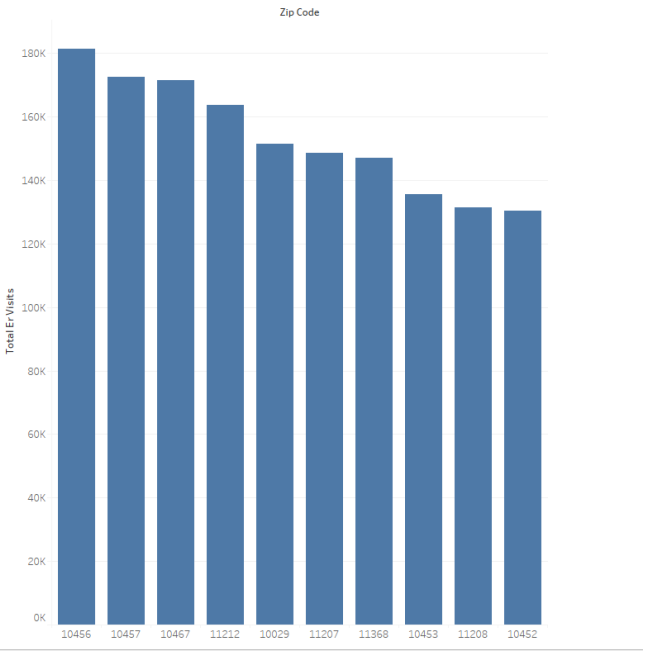

Figure 1: A bar graph snippet of the zip codes with the highest number of total cases

Scoring: For scoring the emergency room table, we ran query 1 (described later) to determine the zip-codes with the highest number of total cases (see Figure 1).

This is only a snippet of the entire result, but we only care about the zip code with the highest number of cases. From the query, zip code 10456 has the highest number of cases, 181422.

We then wrote MapReduce code to take the total number of visits each zip code has and divide it by 181422. Lastly, We assign the score for the zip code as "1.0 - quotient" (a zip code with a higher number of cases would get a lower score). Figure 2 is a snippet of the scores for this table.

Daily Parks Cleaning

Gathering: This dataset contained details of when, and how, a specific park was cleaned. The size was 748.03 MB, and it was publicly available with NYC Open Data.

Cleaning: To clean this dataset, we filtered out columns that we did not need. For example, the table initially had a column listing the fiscal quarter when the park was cleaned. This information isn't needed as there is another column that lists a timestamp value. This value suits our needs more.

To remove duplicates, we combined a few of the rows together (row ID, inspection date, start time, and end time) to form a unique key.

Profiling: In order to determine how zip codes fared against each other with regards to park cleaning (i.e. which zip code spent the most time cleaning vs the least),

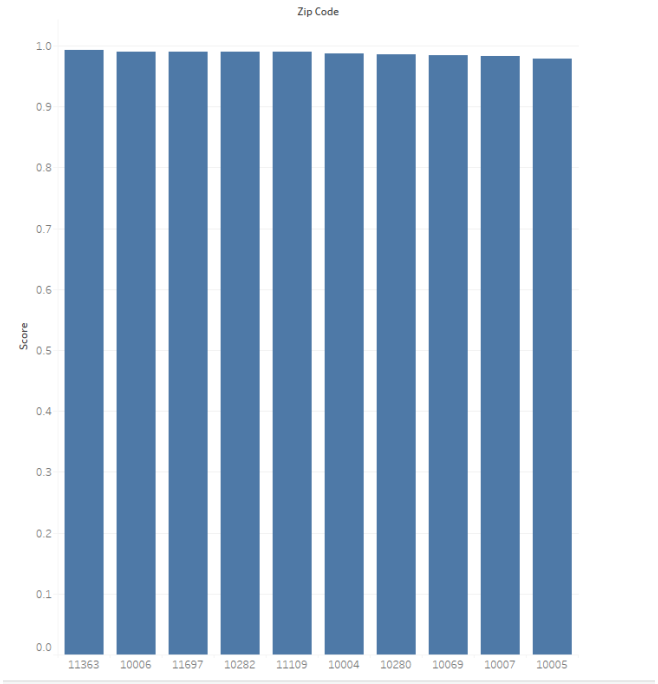

Figure 2: A bar graph snippet returning the entries of the ER score table from highest to lowest

we devised the scoring scheme described in the next section.

Scoring: The Daily Parks scores were calculated in a similar fashion as the one above. We first determined the zip codes with the highest number of work cleaning activities (see Figure 3):

And here is a snippet of the scores table for daily parks (Figure 4):

Rodent Inspection

Gathering: This dataset contained information about rat inspections around the city, was publicly available with NYC Open Data, and was 502 MB in size.

Cleaning: To clean this dataset, we filtered out columns that we did not need. For example, the table initially had columns for the latitude and longitude coordinates of the inspection. This information isn't needed as there is another column that lists a zip-code of where the inspection took place.

Profiling: In order to determine how zip-codes fared against each other with regards to rodent inspection (i.e. which zip-code spent the most time proactively dealing with rodents vs the least, which zip-code had the highest number of violations, etc.), we devised the scoring scheme described in the next section.

Scoring: Scoring the rodents inspection table was a challenge. Each row may represent a positive or negative result, so we could not simply rely on counting rows and

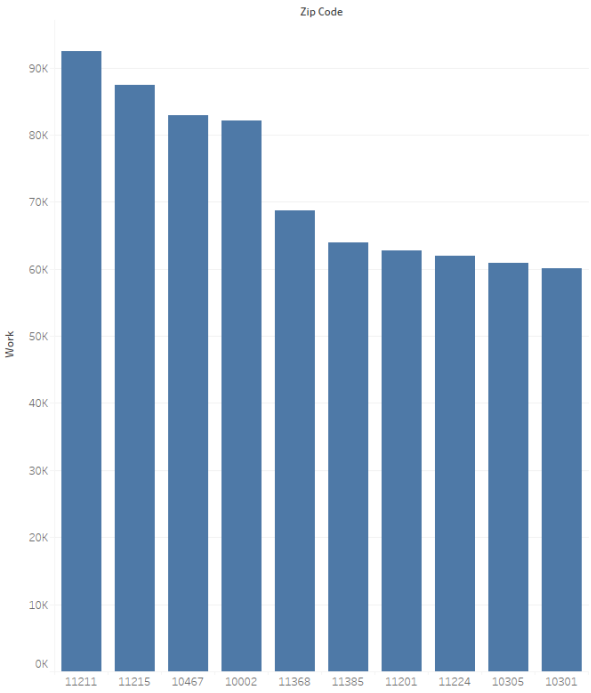

Figure 3: This snippet lists the zip codes with the highest number of work-related activities from highest to lowest

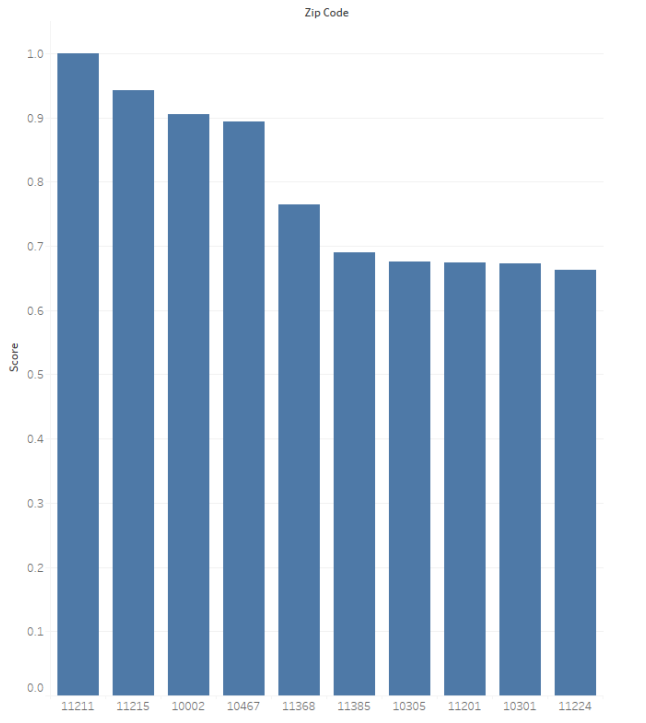

Figure 4: A brief snippet of the daily parks score table

dividing it by something. The inspection types are made up of:

1. Initial
2. Compliance
3. Bait
4. Clean Up
5. Stoppage

A zip-code is awarded a high number of points if it passes an initial inspection, and it loses some points if it fails the initial inspection.

A compliance inspection only occurs if a zip-code fails its initial inspection. If a zip-code passes a compliance inspection, it'll gain points (but not as many as it would have if it passed the initial inspection). If a zip-code fails a compliance inspection, it will lose a lot of points as this is the second time a violation has occurred.

A zip-code will be awarded points for bait, clean up, and stoppage activities. These are all activities that will prevent rodents from infesting areas. Out of the three, bait is awarded the least amount of points (as it's quick to do). Stoppage is awarded the highest number of points out of the three as it takes the most amount of work; according to the website, in a stoppage activity, holes and cracks are sealed up to prevent the free movement of pests.

After the points are tallied up, each zip-code is assigned a final score based on the number of points it earned. Zip-codes with more points will earn a higher score than zip-codes with fewer points would. Figure 5 shows a snippet of the rodents score table.

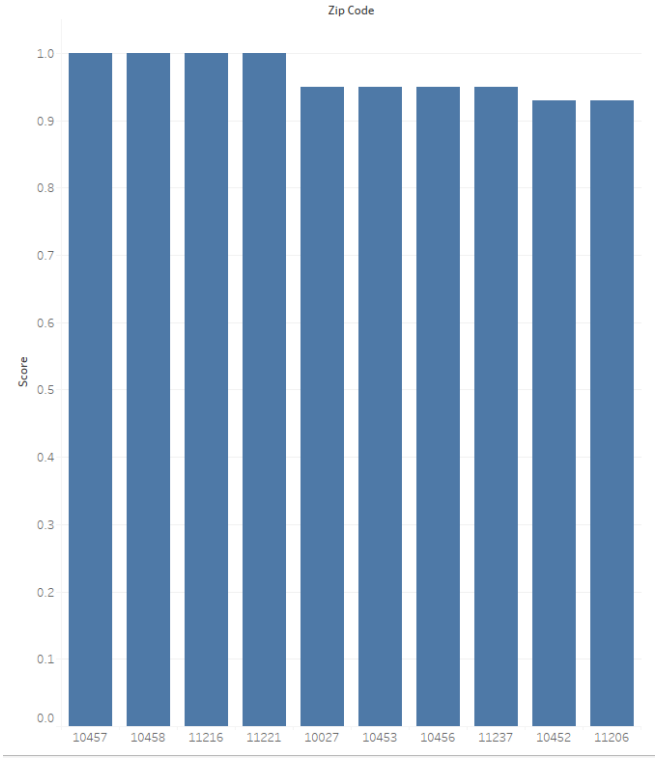

Figure 5: A brief snippet of the rodents score table

Parks Inspection

Gathering: This dataset contained inspection results of parks around the city, was publicly available with NYC Open Data, and its size was 39.04 MB.

Cleaning: Again, we removed unneeded columns such as the district. There wasn't too much to clean in this dataset.

Profiling: Similar to above, MapReduce code was written to determine how zip codes fared against each other with regards to parks inspections. Each zip code would be awarded with/penalized by a number of points depending on the inspection results. The zip-code with the highest score would rank first, etc. These details are described in the next section.

Scoring: The parks inspection table was scored in a similar manner. As mentioned previously, in each inspection, a park is assigned an overall rating and a separate rating for cleanliness.

There are three possible scores; A for acceptable, U for unacceptable, and U/S for extremely unacceptable. A park will receive the highest number of points if it receives an "A" for both its overall and cleanliness rankings. Similarly, a park will lose many points if it receives an "U/S" for both overall and cleanliness.

Different combinations of "acceptable/unacceptable/extremely unacceptable" between the overall and cleanliness ratings will receive different scores. However, as this category is catered towards health, scores will be biased towards the cleanliness rating. For example, a park with an overall rating of unacceptable and a cleanliness rating of acceptable will earn more points than a park with an acceptable overall rating but unacceptable cleanliness rating would.

After the scores were tallied up, we ran the following query to determine the zip-code with the highest score (see Figure 6).

Figure 6 is only a snippet of the entire result, but we only care about the zip code with the highest score. From the query, zip code 10024 has the highest number of points, 140902.

We then wrote MapReduce code to take the total number of points each zip code has and divide it by 140902. Lastly, we assign the score for the zip code as "total number of points / 140902". Figure 7 has a snippet of the scores for this table.

The overall health score for each zip code is just adding each individual score in the health tables above, then dividing it by 4. However, there were some edge cases where a zip code would appear in one dataset, but

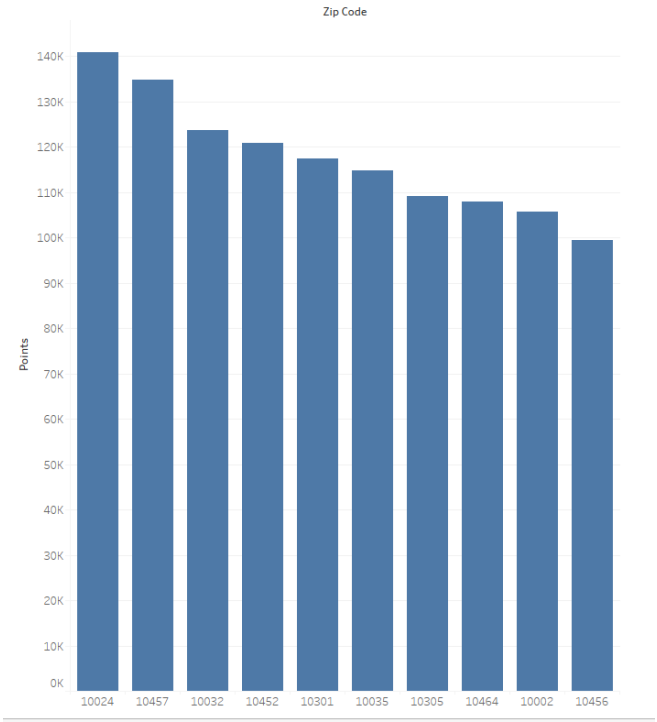

Figure 6: A brief snippet of the number of points each zip code earned to calculate the parks inspect scores table

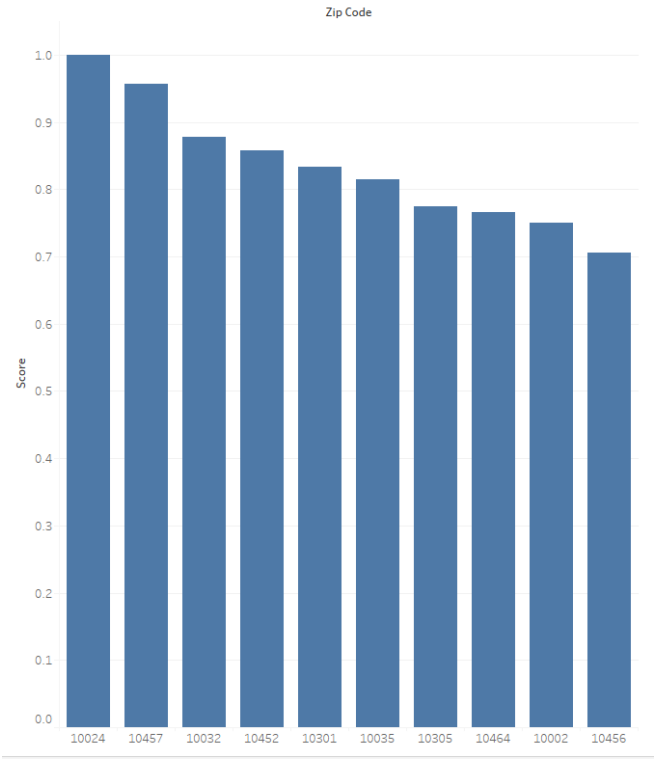

Figure 7: A brief snippet of the parks inspect scores table

not another. To remedy this, we'd refer to the scores of an existing zip code that's adjacent to the missing one when computing the overall average.

Quality of Housing. In order to assess the living conditions of a neighborhood, we utilized two datasets- Housing Maintenance Code Complaints and Housing Maintenance Code Violations.

Housing Maintenance Code Complaints

Gathering: This dataset contains information about the housing maintenance-related complaints made by the public over the phone, or through inspections. It contains 3.08 million records, is publicly available with NYC Open Data and is around 278 MB in size.

Cleaning: The dataset contained a total of 15 columns, out of which the ones relevant for our study were ComplaintID and Zip. Some of the entries did not have a valid zip code. Those entries were removed in the profiling MapReduce function.

Profiling: Profiling the dataset required obtaining data on complaint reports for each zip code. This was accomplished by running a MapReduce task in which the mapper outputted (zip code, 1) for each record and the reducer outputted (zip code, sum). The same reducer method was used as a combainer.

Housing Maintenance Code Violations

Gathering: This dataset contains information about the violations of different types of conditions, primarily in rental accommodation. It contains 8.06 million records. It's cumulative size is around 3.55 GB.

Cleaning: It consisted of 41 columns, out of which we only needed ViolationID and Postcode. Similar to the previous dataset, zip code validation was required as some entries had invalid zip code.

Profiling: Profiling the Housing Maintenance Code Violations followed the exact same steps as that of Housing Maintenance Code Complaints. We obtained the number of violations for each zipcode.

Housing Data, Together

Averaging: We now had two independent datasets which had to be correlated. We wrote a MapReduce job to get the average of Housing Complaints and Violations count combined together for each zipcode.

$$\text{Average(Zip)} = (\text{Housing Complaints}+\text{Housing Violations})/2 \quad (1)$$

Scoring: We had to build a score on a 0-100 scale for each neighborhood in order to finally be able to rank them. We used a MapReduce job, with the following formula for each zip-code:

$$\text{Score(Zip)} = 100-((\text{Average in Zip}/\text{Total Average})*100)$$

Ease of Transportation. In order to assess the connectivity of a neighborhood to other parts of the city, we utilized one dataset- Subway Stations.

Subway Stations

Gathering: This dataset consisted of information about every subway station in New York City. It was publicly available with NYC Open Data, had 473 records, and was 62.8 KB in size.

Cleaning: It consisted of the following columns: URL, Object ID, Name, Latitude-Longitude, Line Details, and Additional Comments. Out of these columns, the ones relevant for our study were Name and Latitude-Longitude. They were retrieved using a mapper-only job (resembling the filtering pattern). In addition to this, a zip-code field was manually appended to each subway station.

Profiling: This dataset, too, followed identical profiling steps as those in the safety data.

Scoring: Scoring this dataset was effortlessly simple, as we followed the same steps taken when scoring the individual safety datasets.

Nearby Restaurant Ratings. We factored this into our assessment by utilizing two datasets- NYC Restaurant Inspection Results and NYC Tripadvisor Restaurant Ratings.

NYC Restaurant Inspection

Gathering: This dataset contained information about all restaurant inspection reported to the New York Department of Health and Mental Hygiene. It contained 200K records, was publicly available with NYC Open Data, and occupied around 86 MB of storage.

Cleaning: The dataset contained a total of 27 columns, out of which the ones relevant for our study were the Inspection-ID, Zipcode, Date of inspection, and Score. They were retrieved using a MapReduce job where the mapper emitted (Inspection-ID, relevant features) and the reducer helped in emitting a single record per inspection by retrieving only the latest inspection.

Profiling: Profiling the dataset implied gathering information about inspections in every zip-code. In order to achieve this, a MapReduce job was run where the mapper emitted (zip code, score-count, score-sum) for every record, and reducer then emitted (zip code, mean).

Scoring: In order to be able to eventually rank neighborhoods, we had to engineer a score on the 0-100 scale for each neighborhood. This was achieved by normalizing the scores using a min-max approach

NYC TripAdvisor Restaurant Ratings

Gathering: This dataset consisted of information about restaurants reatings reported across New York City in TripAdvisor. It was publicly available through a public API, had 5M records, and occupied 2.5 GB of storage.

Cleaning: It consisted of 55 columns, out of which the ones relevant for our study were Restaurant ID, Zipcode, Rating. They were retrieved using a MapReduce job where the mapper emitted (Restaurant ID, relevant features) and the reducer helped in emitting a single record per Restaurant ID. Most of the record are duplicates or not relevant anymore (too old). After cleaning, we end up with just 6K records corresponding to different restaurants in all of NYC

Profiling: Profiling the dataset implied gathering information about inspections in every zip-code. In order to achieve this, a MapReduce job was run where the mapper emitted (zip code, score-count, score-sum) for every record, and reducer then emitted (zip code, mean).

Scoring: The process of scoring the Restaurant TripAdvisor Dataset was identical to that of Restaurant Inspection.

At this point, the two datasets were averaged using the same method followed when averaging the safety datasets. We now had one dataset that reflected the restaurants quality of each neighborhood, based on the average score.

5 Analytics and Results

At this point, we are sure of every averaged dataset being ready for analytics. Therefore, as noticeable in the design diagram, these datasets are placed into the Hive warehouse. With that out of the way, the next step was to engineer SQL queries for gathering insight into the prepared data using Trino/Presto. We expect a user to have a criteria set which is nothing but a subset of our five factors. For the sake of simplicity, and in the interest of time, we decided to stick with subsets of lengths 1, 2, and 5. Beforing jumping into the results, we would like to take a moment to describe the queries engineered.
The query for the top-k areas by a single factor (subsets of length 1) would be as follows:

```
SELECT ZIPCODE, SCORE
FROM <table>
ORDER BY SCORE DESC
LIMIT <k>;
```

The query for the top-k areas by two factors (subsets of length 2) would be as follows:

```
SELECT coalesce(table1.ZIPCODE,
                table2.ZIPCODE)
```

```
AS ZIPCODE,
table1.SCORE + table2.SCORE
AS TOTAL_SCORE
FROM table1
FULL OUTER JOIN table2
ON table1.ZIPCODE = table2.ZIPCODE
ORDER BY TOTAL_SCORE DESC
LIMIT k;
```

Finally, the query for the top-k areas considering all factors (subset of length 5) would be as follows:

```
SELECT coalesce(table1.ZIPCODE,
                table2.ZIPCODE,
                table3.ZIPCODE,
                table4.ZIPCODE,
                table5.ZIPCODE)
AS ZIPCODE,
table1.SCORE + table2.SCORE +
table3.SCORE + table4.SCORE +
table5.SCORE AS TOTAL_SCORE
FROM
(
    table1
    FULL OUTER JOIN table2
    ON table1.ZIPCODE = table2.ZIPCODE
    FULL OUTER JOIN table3
    ON coalesce(table1.ZIPCODE,
                table2.ZIPCODE)
                = table3.ZIPCODE
    FULL OUTER JOIN table4
    ON coalesce(table1.ZIPCODE,
                table2.ZIPCODE,
                table3.ZIPCODE)
                = table4.ZIPCODE
    FULL OUTER JOIN table5
    ON coalesce(table1.ZIPCODE,
                table2.ZIPCODE,
                table3.ZIPCODE,
                table4.ZIPCODE)
                = table5.ZIPCODE
)
ORDER BY TOTAL_SCORE DESC
LIMIT k;
```

Obtaining the Top 10 Neighborhoods by Livability. We started by identifying the 10 most livable neighborhoods in New York City. In order to do this, we ran the third query from amongst the three listed above. The results are shown in Figure 8.

It is important to map these zip-codes to the more-relatable "areas", in order to better-understand the findings. This mapping is precisely what Table 1 depicts. Unsurprisingly, most neighborhoods are in and around the affluent Manhattan areas, while a few of them such as Ridgewood in Queens are traditionally more residential.

```
 ZIPCODE |    TOTAL_SCORE
---------+--------------------
 10002   | 320.35066298244567
 10011   |  312.3104289087029
 10013   |  311.8464772206866
 10003   | 307.50591144145267
 11385   | 303.60046565811325
 11201   |  302.3191040550732
 10027   |  301.4031131994186
 10012   | 300.07154029102094
 10014   |  298.3694137340087
 10026   |  296.0660611034547
(10 rows)
```

Figure 8: The 10 most livable neighborhoods of NYC

Zip-code	Area
10002	Downtown
10011	Chelsea
10013	Tribeca
10003	East Village
11385	Ridgewood
11201	Brooklyn Downtown
10027	Morningside Heights
10012	SoHo
10014	Meatpacking District
10026	Uptown

Table 1: Caption

With that done, it would be interesting to see if different sets of factors result in the same neighborhoods. That would give us some insight into the extent to which a factor affects the overall livability of a zip-code. In the interest of time, we confine ourselves to the analysis of single-factor and four arbitrarily-chosen two-factor subsets. We shall now dive into those results.

Analysis of Safety. Figure 9 shows the 10 safest neighborhoods in the city. Yet again, it is unsurprising that the most neighborhoods belong to the Manhattan borough. Perhaps the most surprising entrant was 10471, which happens to be zip-code in The Bronx which is notorious for being unsafe. However, upon looking at the data closer, we were able to infer that the number of complaints and shooting incidents reported in and around 10471 are far lesser than other areas in that borough. The most interesting, in our opinion, was 11694 which maps to the Rockaway Park in Queens. This was actually quite an accurate output, considering that 11692 (which is just 3 miles away) had significantly greater shooting incidents.

ZIPCODE	SCORE
11104	99.97852
10005	99.964645
11379	99.96043
10027	99.95209
10020	99.939865
10006	99.93625
10021	99.93324
11694	99.91263
10471	99.90863
11231	99.88034

(10 rows)

Figure 9: The 10 safest neighborhoods of NYC

Analysis of Housing. Figure 10 shows the 10 neighborhoods which has fewer housing complaints and violations. Among those, 4 were from Manhattan and 2 were from Staten Island.

Analysis of Transportation. Figure 11 shows the 10 neighborhoods which are ideally located to utilize the subway network. Considering that the NYC Subway is one of the most well-connected, it is certain that residents of these areas would barely have to face any problem due to the lack of public transport.

The first thing to observe, are the extremely scores. This can be attributed to the storing technique utilized given that each zip-code would barely have more than a certain number of subway stations. However, the interesting insight is that in this regard, Manhattan seems to have been overpowered (to some extent) by its southward counterpart- Brooklyn.

Analysis of Restaurants. Figure 12 shows the 10 neighborhoods which are ideal for a visit to a good restaurant.

The scores seem average with most being around 50 and only the Lower East Side being by far on top of the list with 80 score. However, the interesting insight is that even though the Lower East Side wins the 1st place, Chelsea and Clinton seems to have much more consistent good areas than the Lower East Side. Thus, while the

ZIPCODE	SCORE
17850	99.99999104146869
10325	99.99999104146869
10446	99.99999104146869
11122	99.99999104146869
10119	99.99999104146869
11677	99.99999104146869
10048	99.99998208293736
10364	99.99998208293736
10121	99.99998208293736
12238	99.99997312440605

Figure 10: The 10 neighborhoods of NYC with fewest housing complaints and violations

ZIPCODE	SCORE
11201	2.7484143
10013	2.7484143
11207	2.536998
10011	2.3255813
11101	2.1141648
10007	2.1141648
11208	1.9027485
11217	1.9027485
11226	1.6913319
11218	1.4799154

(10 rows)

Figure 11: The 10 most well-connected areas of NYC

Borough	Neighborhood	ZipCode	rank	score
Manhattan	Lower East Side	10009	0	80.000000
Queens	Southwest Queens	11415	1	60.000000
Queens	Southwest Queens	11416	2	53.960122
Manhattan	Chelsea and Clinton	10018	3	53.921984
Manhattan	Chelsea and Clinton	10019	4	53.790422
Manhattan	Chelsea and Clinton	10020	5	52.371773
Manhattan	Chelsea and Clinton	10011	6	52.192026
Staten Island	South Shore	10308	7	50.295338
Queens	Southwest Queens	11414	8	49.367108
Manhattan	Chelsea and Clinton	10001	9	47.329912

Figure 12: The 10 best restaurants areas of NYC

Borough	Neighborhood	ZipCode	rank	score
Manhattan	Chelsea and Clinton	10018	0	153.732114
Manhattan	Chelsea and Clinton	10019	1	153.103572
Manhattan	Chelsea and Clinton	10020	2	152.311638
Manhattan	Chelsea and Clinton	10011	3	151.070482
Queens	Southwest Queens	11414	4	149.169488
Manhattan	Chelsea and Clinton	10001	5	146.654042
Manhattan	Lower East Side	10003	6	146.377581
Manhattan	Greenwich Village and Soho	10013	7	144.857466
Brooklyn	Northwest Brooklyn	11205	8	144.244648
Bronx	Hunts Point and Mott Haven	10454	9	144.238682

Figure 13: Top 10 zip-codes by restaurants and safety

lower-east side might seem a good place for a one-time dinner, Chelsea and Clinton could offer more variety in terms of good restaurants.

Analysis of Restaurants and Safety. The results are depicted in Figure 13. With crime under consideration now, Manhattan is preferred among the other neighborhoods since it offers a lot of high quality restaurants in a relatively safe area, compared to Queens who may have scored well on restaurants but could not beat Manhattan when factored in with safety. We also see how Chelsea's safety scores pushed these zip-codes to the top while the previous the zip-code which was previously the best, did not even make it to the list. Therefore, Chelsea seems to be a good choice when eating good and safe.

Analysis of Housing and Transportation. The first of the randomly chosen subset, happened to be that of housing and transportation. The results are depicted in Figure 14. Manhattan is back to being where it is expected to. Also, the presence of Queens (11417) here coupled with its absence from the previous set (transportation only) results indicate that housing plays a crucial role in the livability of neighborhoods there. This isn't particularly surprising given that Queens is widely regarded as a largely residential borough. Also, we can observer some degree of similarity with the cumulative results displayed earlier. This can be attributed to the fact that the introduction of more factors would skew our results to the cumulative ones.

Analysis of Health and Safety. We have combined two subsets (Health and Crimes) into 1. The results are depicted in Figure 15. Manhattan is favored over the other neighborhoods when it comes to crime along with health. We compared these results with the top 10 neighborhoods by health and crime separately. When compared with the top 10 healthiest neighborhoods, it was seen that the top 5 zip codes were the same in both. But whereas in the 10 safest neighborhoods, we were not

```
 ZIPCODE |    TOTAL_SCORE
---------+--------------------
 10013   | 102.53762901664523
 11201   | 102.44833933500993
 10007   | 102.10234849719303
 10011   | 101.92301177821356
 11101   |  101.9163693870547
 11217   | 101.48480613847381
 11417   | 101.37078257149781
 10023   | 101.19853688986825
 10001   | 101.13016135940481
 10019   | 101.06884423199385
(10 rows)
```

Figure 14: Top 10 zip-codes by housing and transportation

```
 ZIPCODE |    TOTAL_SCORE
---------+--------------------
 10002   | 178.64614133466907
 10024   | 178.15804005281694
 10032   | 170.62857401807207
 10044   | 170.55112890916388
 10035   | 168.79398465248644
 10027   | 165.53879250824315
 10452   | 164.44492462497703
 11215   | 164.28005402240882
 10013   |  163.8510482642257
 10040   | 162.13123523448462
```

Figure 15: Top 10 zip-codes by health and safety

able to find the same areas in both despite most it had many areas from Manhattan. We might thus draw the conclusion that the combined health and crime score was decently skewed by the health criterion.

Analysis of Health and Housing. Figure 16 is a picture of the top ten zip codes with the best health and housing scores (generated from Tableau). The scores are ordered from highest to lowest

To determine why these zip codes were presented, we first took a look at the top ten zip codes with the best health scores (see figure 17).

Using these ten zip codes, we took a look at how each fared in the emergency room scores table. Some did very well, while others performed very poorly (figure 18).

We ran the same query for the daily parks score table,

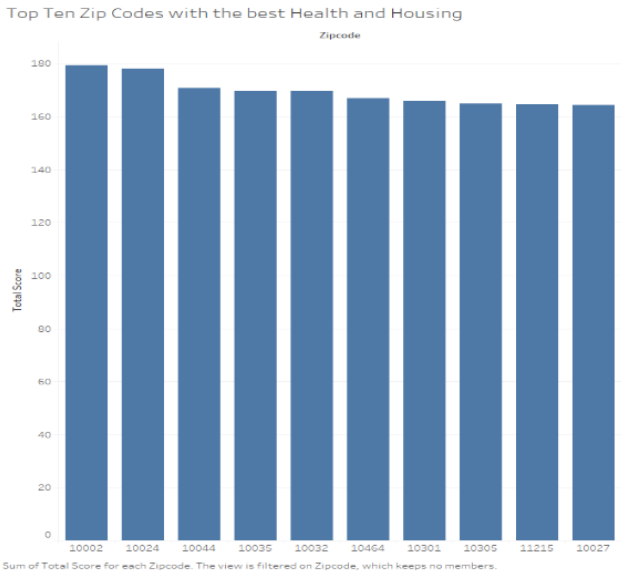

Figure 16: A bar graph showing the top ten zip codes with the best scores when considering both health and housing

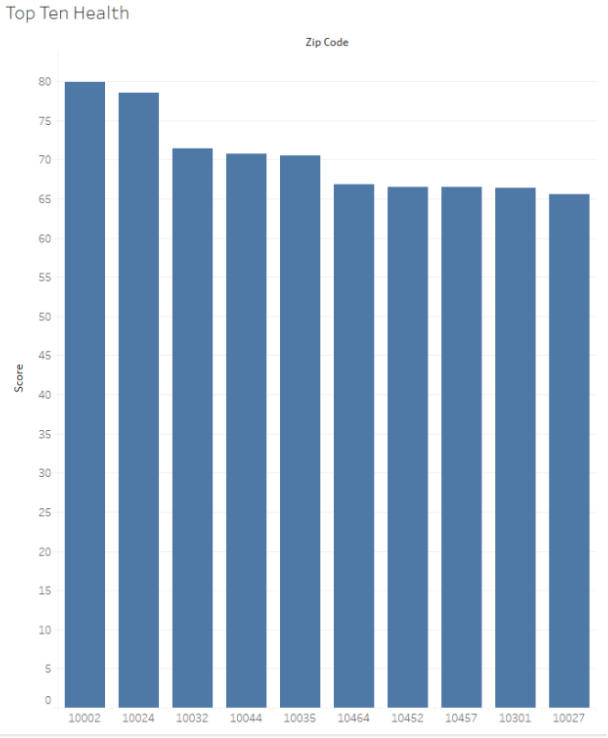

Figure 17: A bar graph showing the top ten zip codes with the best health scores

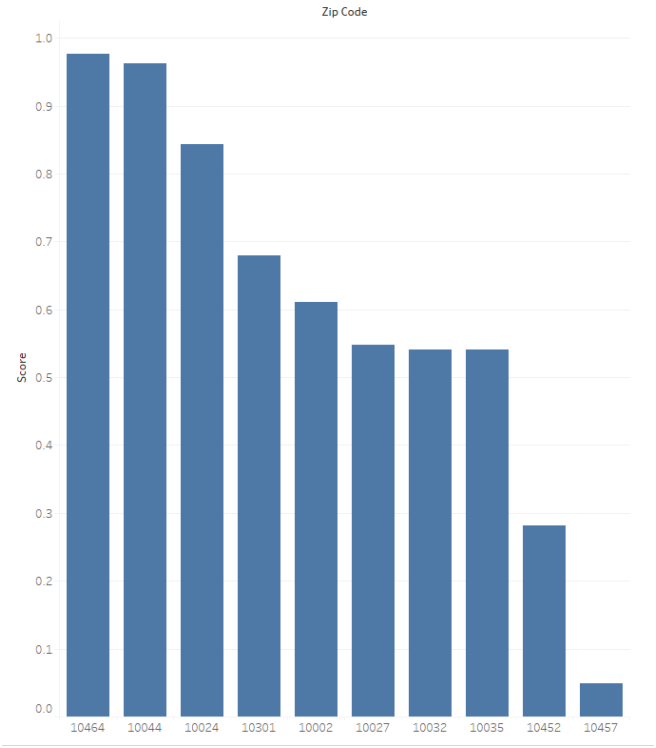

Figure 18: A bar graph depicting how the top 10 zip codes performed in the ER score table

and figure 19 has the results.

Note that only 9 results were returned; this is because zip code 10044 does not appear in the daily parks score table. In order to estimate what its score would be, we referred to a score from an existing nearby zip code in the same table. In this instance, that would be zip code 10002 (figure 19).

Running the same query for the parks inspection score table yields the results in figure 20.

Note that only 9 results were returned; this is because zip code 10044 does not appear in the parks inspect score table. In order to estimate what its score would be, we referred to a score from an existing nearby zip code in the same table. In this instance, that would be zip code 10002 (again, refer to figure 20).

And lastly, running the same query for the rodents table yields the results in figure 21.

We've discovered some interesting facts from these results. Some zip codes that scored well in the ER score table earned mediocre (or even low) scores in the other three tables. We would expect a zip code that earned a high ER score would also earn high scores in the other tables.

What's even more interesting is that a zip code's daily cleaning score does not necessarily correlate with its park inspection score. For example, parks in zip code 10002 were cleaned the most often. However, zip code 10002 did not get the highest score in the parks inspection re-

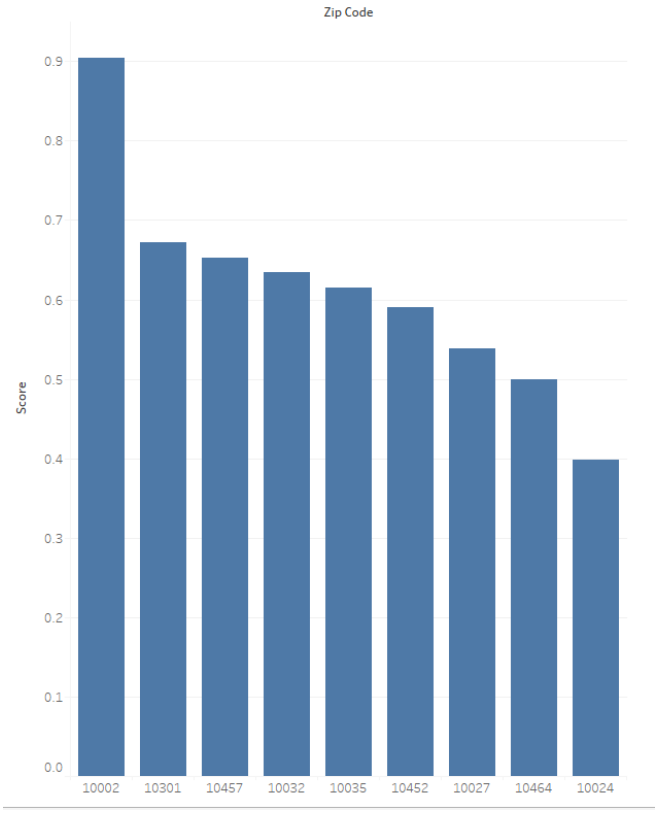

Figure 19: A bar graph depicting how the top 10 zip codes performed in the Daily Parks score table

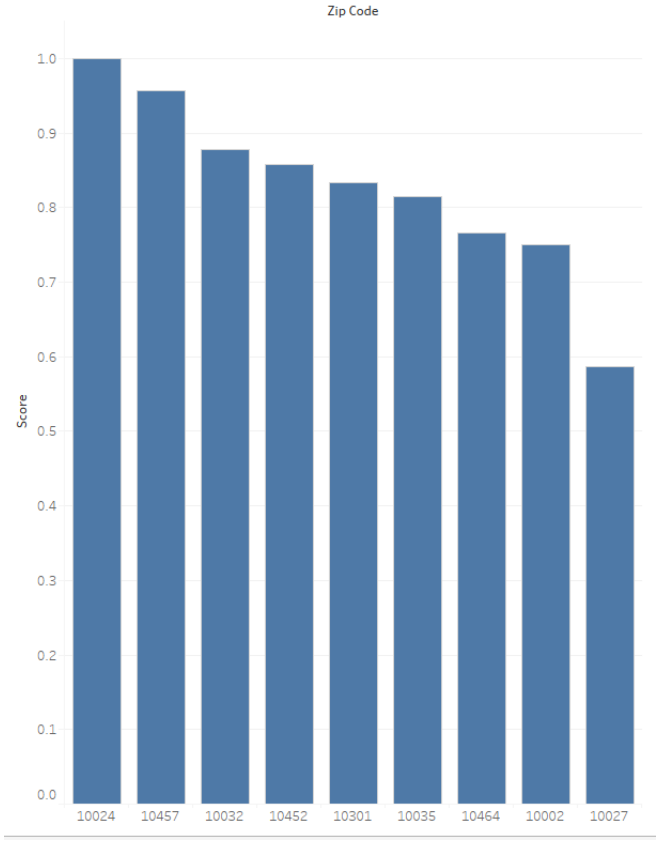

Figure 20: A bar graph depicting how the top 10 zip codes performed in the Parks Inspect score table

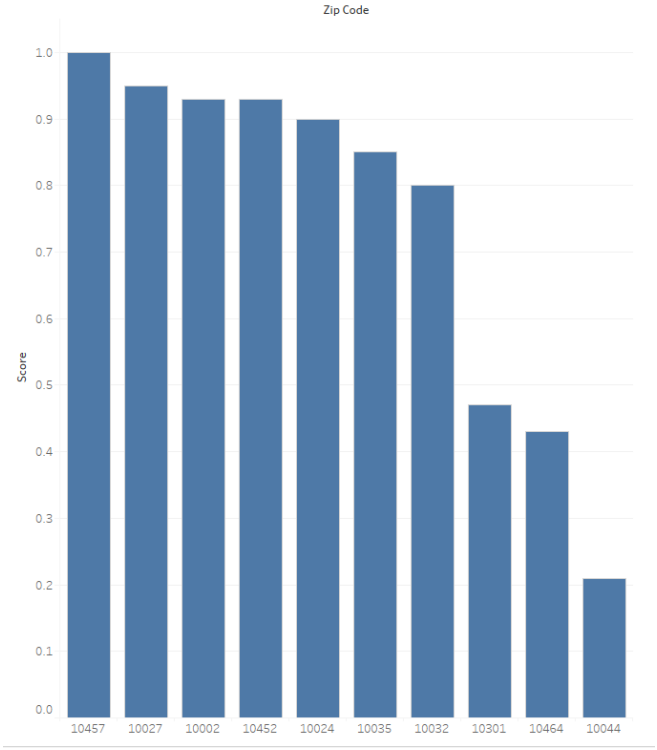

Figure 21: A bar graph depicting how the top 10 zip codes performed in the Rodents Inspection score table

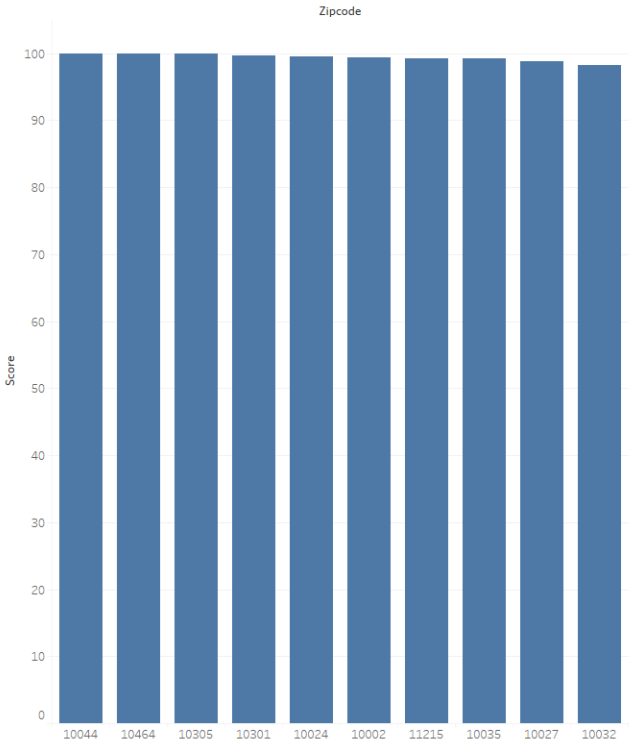

Figure 22: A bar graph depicting the top ten zip codes with the best housing scores

sults score table. We would expect that zip codes with a higher number of parks cleaned would have a higher parks inspection score (or vice versa).

Despite these anomalies, taking the sum of all the scores for each zip code and dividing it by four yields the results in the overall health scores table.

We take the zip codes from the top ten zip codes with the best housing and health and see how they fared in just the housing category (figure 22).

Regarding these zip codes, this is how they fared in the housing average table (figure 23).

The housing average table is just the average count of the number of complaints and violations a particular zip code had. The higher the average, the lower the overall score.

We were interested in seeing how these zip codes fared in the health category. There could be a correlation between health and housing based on the datasets.

Taking zip code 10464 for example (one of the higher scoring zip codes), Figure 24 depicts how it fared across the various health score tables.

Zip code 10464 scored very well in the ER score table, but earned a mediocre score in the parks inspect table. It earned poor scores in the rodents and daily parks score tables.

Narrowing things down further, Figure 25 depicts how zip code 10464 fared in just the housing complaints and housing violations tables.

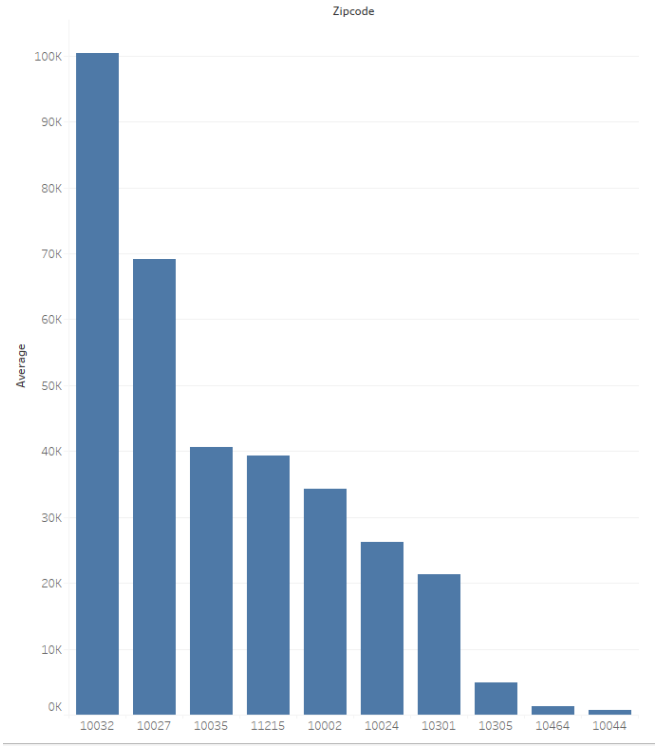

Figure 23: A bar graph depicting the average number of complaints/violations each of the top ten zip codes have

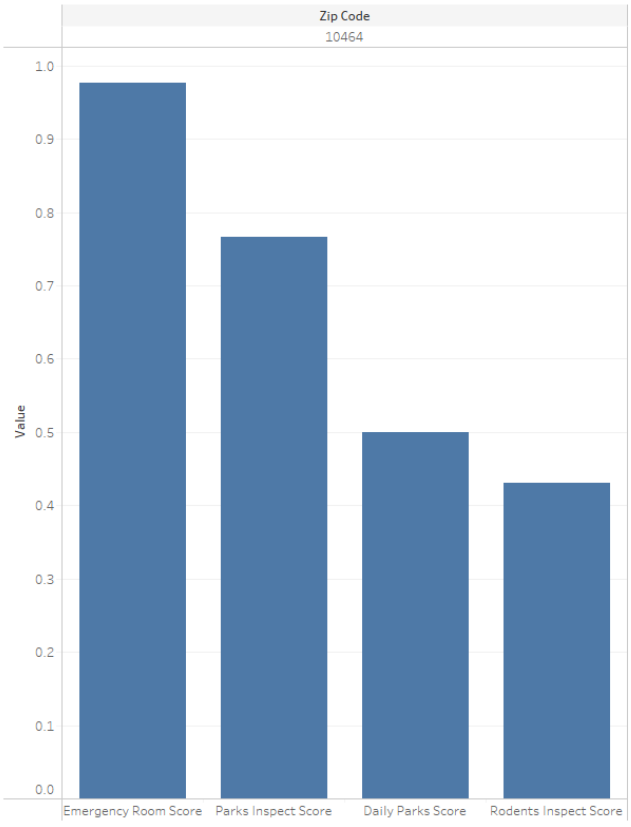

Figure 24: A bar graph depicting what scores zip code 10464 earned across the health score tables

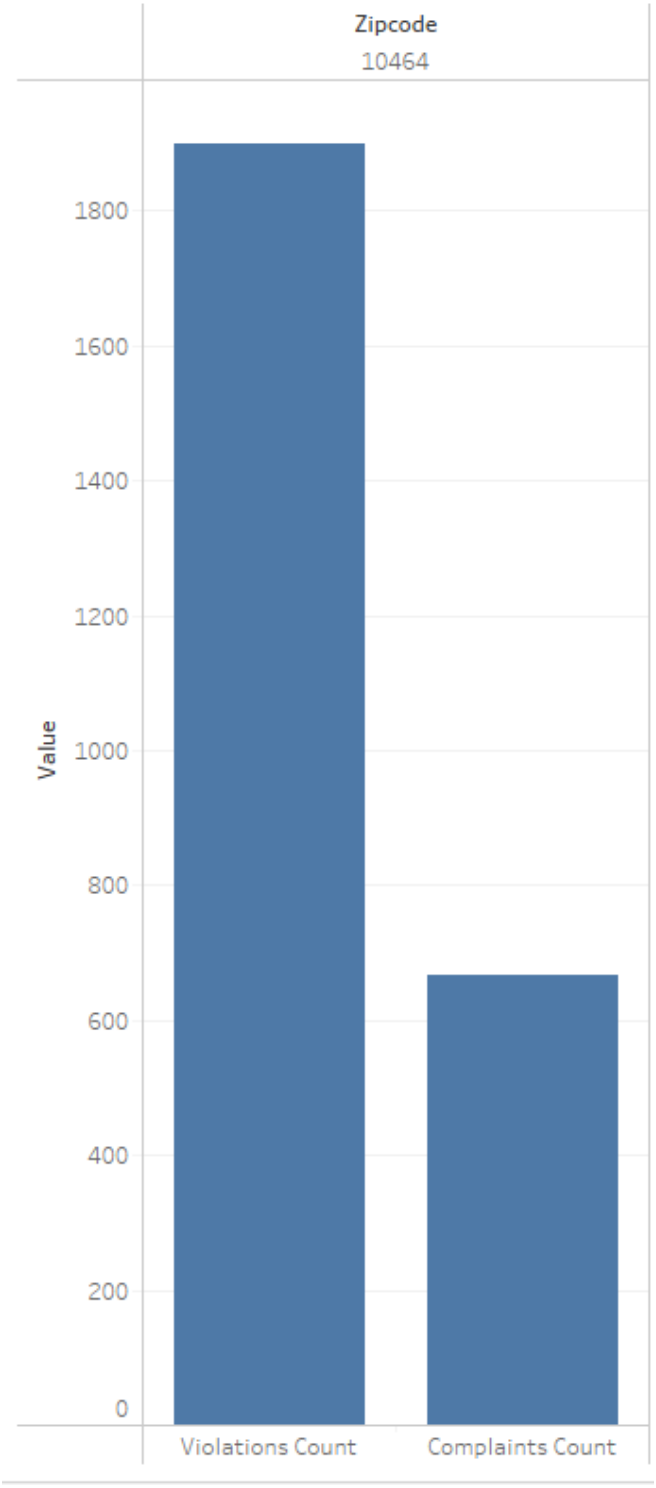

Figure 25: A bar graph depicting how many complaints and violations zip code 10464 has

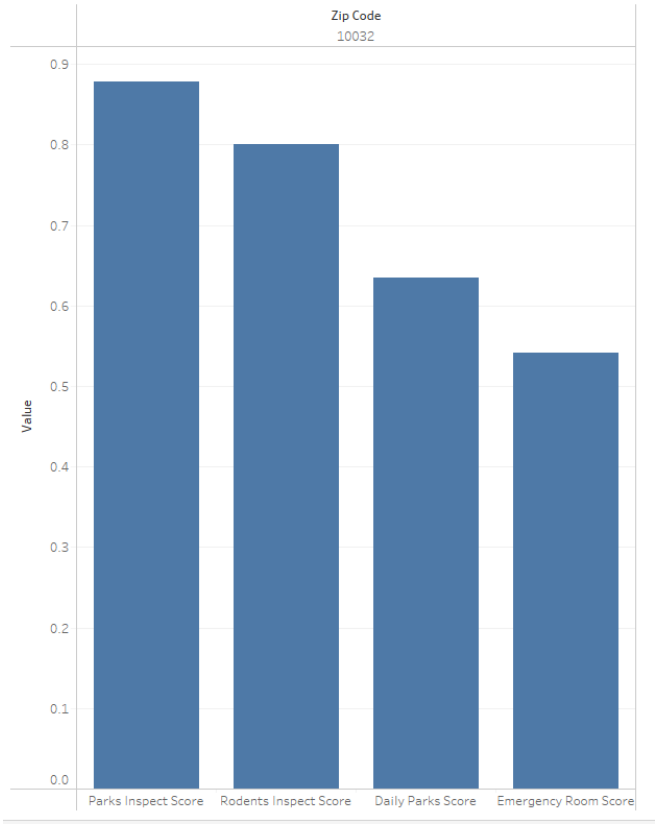

Figure 26: A bar graph depicting what scores zip code 10032 earned across the health score tables

Now looking at the opposite end, we examine a zip code with the lowest score in the top 10 housing scores, zip code 10032 (see figure 26).

Despite having the lowest score in the top 10 housing zip codes score table, zip code 10032 earned pretty good scores in the parks inspect and rodents tables. It earned somewhat mediocre scores in the ER and daily parks score tables.

Once again, we examine how zip code 10032 fared in the individual housing tables (see figure 27).

The results, again, are somewhat unexpected. We would expect that zip code 10032, as it has a lower overall score in the top ten health and housing zip code list, would have a lower score across all of the health categories compared to zip code 10464. Zip code 10032 scored well in the parks inspection and rodents score tables, whereas zip code 10464 scored relatively well in the ER and parks inspection score tables. Then again, the housing category, as its name implies, only takes into account housing complaints and violations. Park inspection and cleaning records wouldn't be included.

What does make sense from the gathered data is that zip code 10032 has a lower score than zip code 10464 does in the Emergency room score table. From the housing violations and housing complaints tables, zip code 10032 has far more records than zip code 10464 does. It would

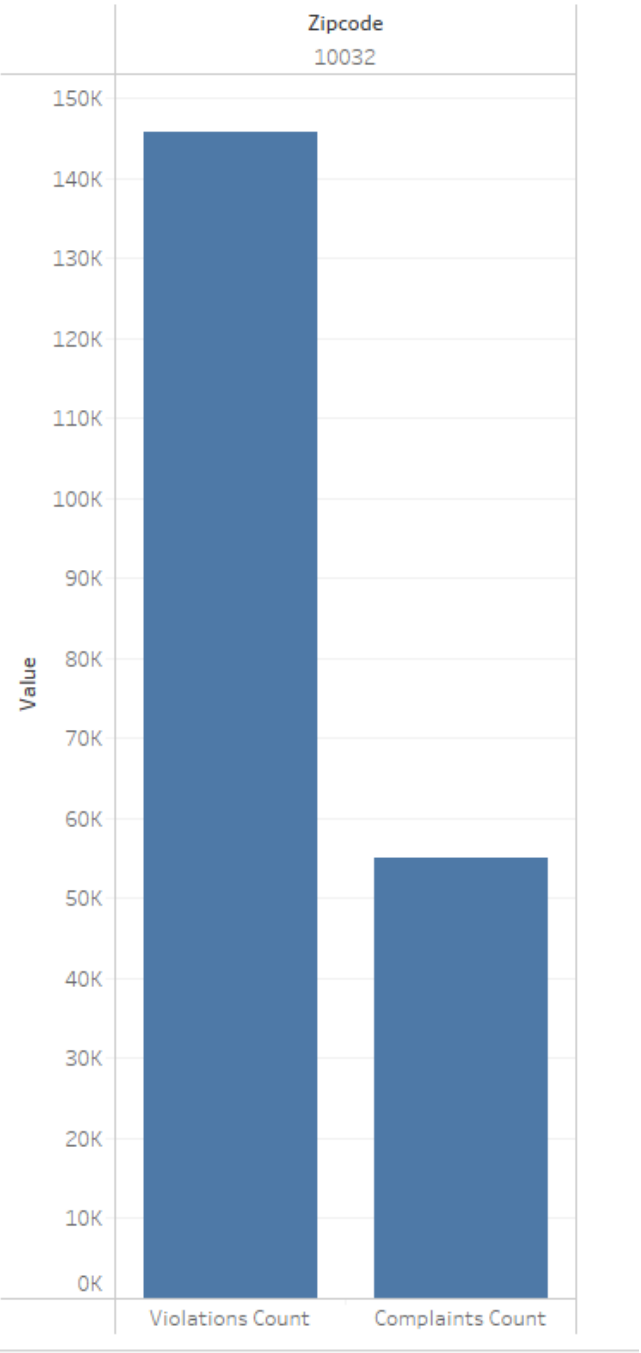

Figure 27: A bar graph depicting how many complaints and violations zip code 10032 has

make sense then that residents living in zip code 10032 would have a higher probability of being admitted to the ER for various injuries and illnesses.

6 Conclusions and Goodness

Our analytic defines livability to be a heterogeneous combination of safety, health & hygiene, quality of housing, ease of transportation, and nearby eatery ratings. Considering these factors, some of the most "livable" neighborhoods inferred from the returned zip-codes are Downtown, Lower East Side, Chelsea, Midtown West, Tribeca, East Village, and Ridgewood.

These results show some extent of intersection with recently published news and survey articles such as [Yor22], [Rah22], and [Hoo22]. Driven by the fact that our results are in the vicinity of those published by reputed agencies such as NBC New York, Timeout, and StreetEasy, we believe that our study can be trusted.

With that said, it is important to note that livability is a very subjective term. Therefore, we would like to state that it is important to run the analytic on one's own criteria before drawing any concrete conclusions.

References

[Cod] US Zip Codes. "https://www.unitedstateszipcodes.org/". In: Find adjacent zip codes.

[Cod+17] Codebender et al. "https://stackoverflow.com/questions/32068129/parse-csv-file-in-java-and-delaing-with-empty-values". In: Parsing strategies. 2015 and 2017.

[Data] NYC Open Data. "https://data.cityofnewyork.us/City-Government/Daily-Tasks-Park-Cleaning-Records/kwte-dppd". In: Daily Tasks Park Cleaning Records.

[Datb] NYC Open Data. "https://data.cityofnewyork.us/City-Government/Parks-Inspection-Program-All-Sites-MAPPED-/buk3-3qpr". In: Parks Inspection Program – All Sites (MAPPED).

[Datc] NYC Open Data. "https://data.cityofnewyork.us/City-Government/Parks-Supervisor-Inspections-Feature-Findings/wag2-kf63". In: Parks Supervisor Inspections - Feature Findings.

[Datd] NYC Open Data. "https://data.cityofnewyork.us/City-Government/Parks-Supervisor-Inspections-Inspection-Results/uwim-9338". In: Parks Supervisor Inspections - Inspection Results.

[Date] NYC Open Data. "https://data.cityofnewyork.us/Health/DOHMH-New-York-City-Restaurant-Inspection-Results/43nn-pn8j". In: DOHMH New York City Restaurant Inspection Results.

[Datf] NYC Open Data. "https://data.cityofnewyork.us/Health/Emergency-Department-Visits-and-Admissions-for-Inf/2nwg-uqyg". In: Emergency Department Visits and Admissions for Influenza-like Illness and/or Pneumonia.

[Datg] NYC Open Data. "https://data.cityofnewyork.us/Health/Rodent-Inspection/p937-wjvj". In: Rodent Inspection.

[Dath] NYC Open Data. "https://data.cityofnewyork.us/Housing-Development/Housing-Maintenance-Code-Complaints/uwyv-629c". In: Housing Maintenance Code Complaints.

[Dati] NYC Open Data. "https://data.cityofnewyork.us/Housing-Development/Housing-Maintenance-Code-Violations/wvxf-dwi5". In: Housing Maintenance Code Violations.

[Datj] NYC Open Data. "https://data.cityofnewyork.us/Public-Safety/NYPD-Complaint-Data-Historic/qgea-i56i". In: NYPD Complaint Data Historic.

[Datk] NYC Open Data. "https://data.cityofnewyork.us/Public-Safety/NYPD-Shooting-Incident-Data-Historic-/833y-fsy8". In: NYPD Shooting Incident Data (Historic).

[Datl] NYC Open Data. "https://data.cityofnewyork.us/Transportation/Subway-Stations/arq3-7z49". In: Subway Stations.

[DG04] Jeffrey Dean and Sanjay Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters". In: 2004 (cit. on p. 2).

[Hoo22] Finn Hoogensen. "https://pix11.com/news/local-news/streeteasys-10-nyc-neighborhoods-to-watch-in-2023/". In: StreetEasy's 10 NYC neighborhoods to watch in 2023. Dec. 2022 (cit. on p. 16).

[Map] Zip Data Maps. "https://www.zipdatamaps.com/". In: Find adjacent zip codes.

[MwB13] Mena, weaselflink, and Bernhard Barker. "https://stackoverflow.com/questions/16817031/how-to-iterate-over-regex-expression". In: Regex iteration strategies. 2013.

[Rah22] Anna Rahmanan. "https://www.timeout.com/newyork/news/these-10-nyc-neighborhoods-will-be-the-hottest-ones-to-watch-in-2023-121522". In: These 10 NYC neighborhoods will be the hottest ones to watch in 2023. Dec. 2022 (cit. on p. 16).

[Set+] Raghav Sethi et al. "Presto: SQL on Everything". In: (cit. on p. 2).

[Shv+10] Konstantin Shvachko et al. "The Hadoop Distributed File System". In: 2010 (cit. on p. 2).

[Thu+09] Ashish Thusoo et al. "Hive – A Petabyte Scale Data Warehouse Using Hadoop". In: 2009.

[Web] NYC Gov Website. "https://www.nyc.gov/assets/doh/downloads/pdf/ah/zipcodetable.pdf". In: Zip Code Table.

[Yor22] 4 New York. "https://www.nbcnewyork.com/news/local/15-most-coveted-nyc-neighborhoods-revealed/3903790/". In: NBC News Article. Oct. 2022 (cit. on p. 16).

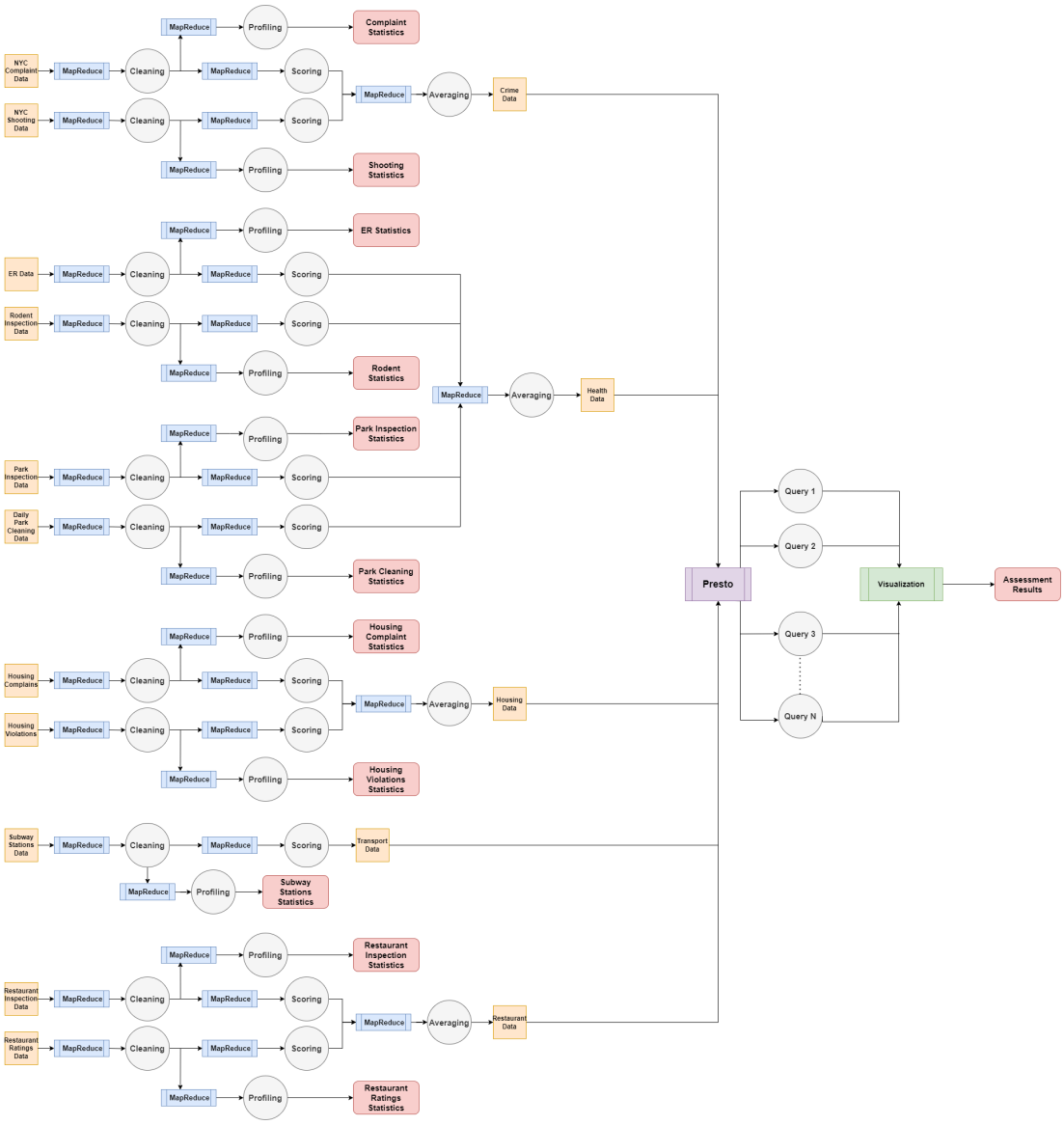

Figure 28: Design Diagram