Report for Assiment 1

- PanZhiQing 24037665g
- repo : https://github.com/pzq123456/LSGI524A/tree/main/assiment1

Task1 [25 points]:

(1) How many valid bicycle trips were documented on 25 July 2019?

20187

(2) How many bike stations were used on that day?

535

(3) How many unique bikes were used?

3822

Task2 [25 points]:

Indicator	Trip duration(s)	Trip distance(m)
Max value	31243	20083.35
Min value	61	103.67
Median	799.0	1840.42
Mean	1187.6403626096003	2460.647581079681
25% percentile	465.0	1107.48
75% percentile	1403.0	3202.5
Standard deviation	1400.9766896637864	1950.106137982676

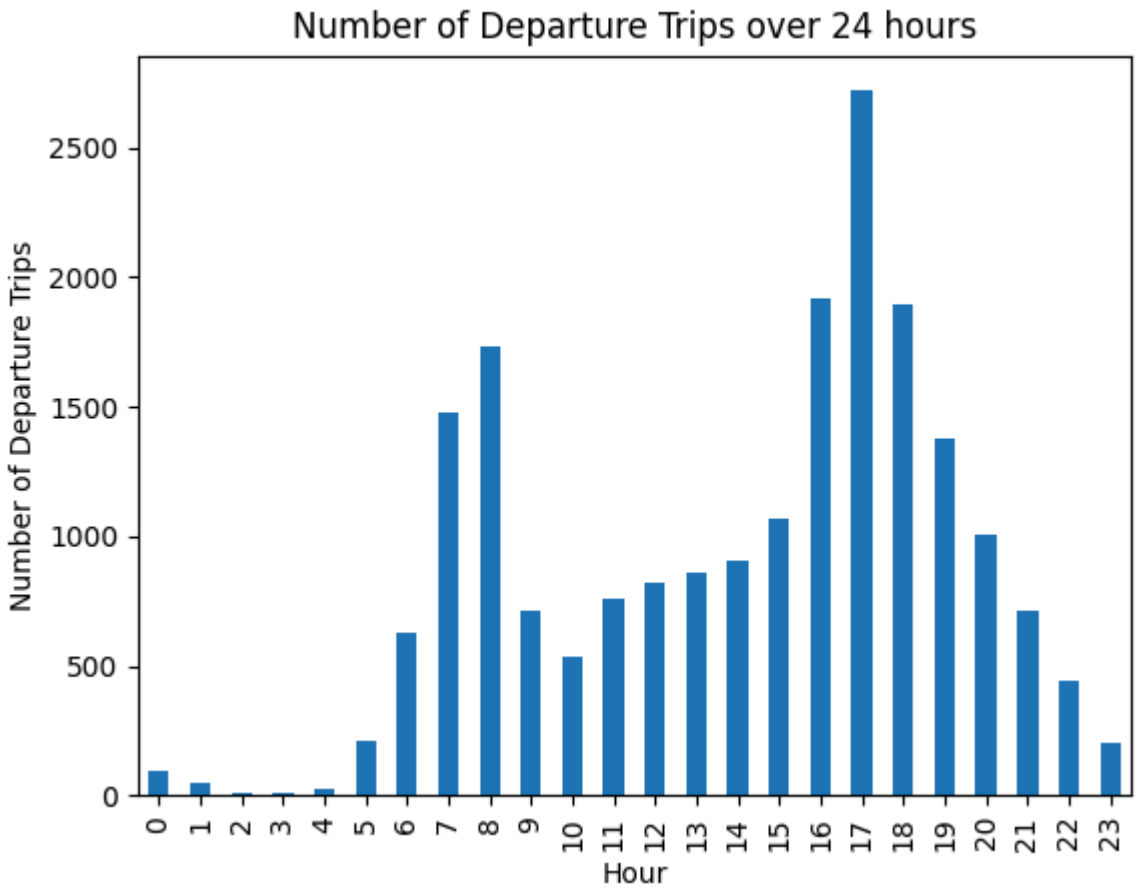
- Trip distance may be zero. I found 885 trips have the same original and destination station. Those can be considered as invalid records. Ignoring them, I get the results above, and you can check the original results below.

Indicator	Trip distance(m)
Max value	20083.35
Min value	0.0
Median	1753.74
Mean	2352.772557091197
25% percentile	1016.56
75% percentile	3096.995
Standard deviation	1972.3091923774828

Task3 [25 points]:

Data visualization based on the processed bike-sharing data. Please use the skills you have gained in data visualization to present answers to the following questions. Both figures and corresponding descriptions should be included in your report.

(1) How does the number of departure trips change over 24 hours? Is there any rhythm or pattern?



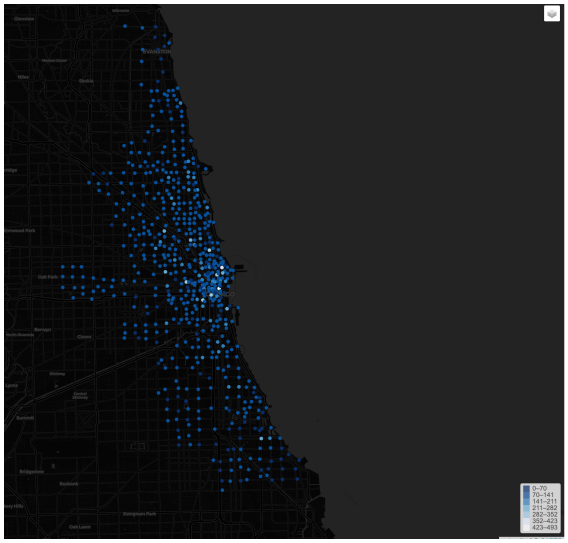
> figure 1: The number of departure trips over 24 hours. The x-axis represents the hour of the day, and the y-axis represents the number of departure trips.

There are two peaks at around 8 am and 5 pm, and the peak around 5 pm is the highest, which may be due to people using shared bicycles when going to and from work. The usage is minimal in the early morning, which may be due to most people sleeping.

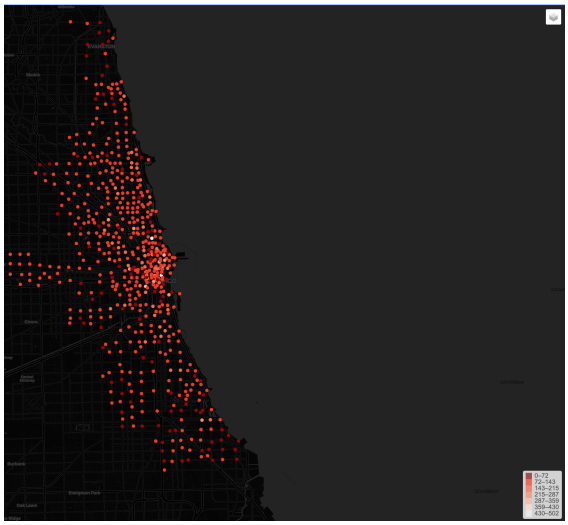
(2) What is the distribution of the number of departure trips at different stations? What about the distribution of arrival trips?

We can find St. Clair St & Erie St station, Racine Ave & Randolph St, Cityfront Plaza Dr & Pioneer Ct station have the most departure trips, and the most arrival trips are at the same stations. The most busy stations are mainly concentrated in the East of the city, especially in the downtown area(near the lake harbor).

For the departure distribution, the high number of departure trips is more dispersed specially than the arrival distribution. We can infer that living areas are more dispersed than working areas, and people may live in different areas but work in the same area.



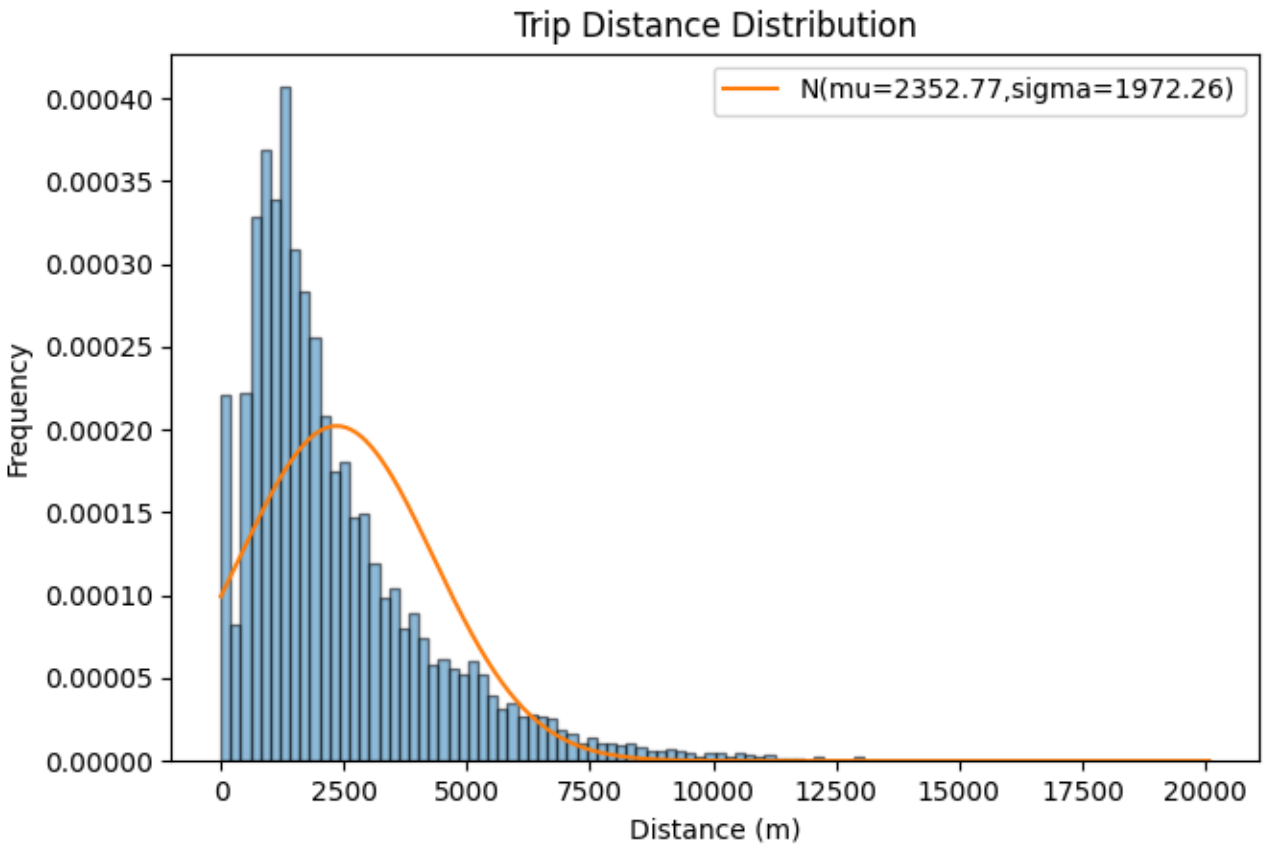
> Figure 2: The spatial distribution of the number of departure trips at different stations.



> Figure 3: The spatial distribution of the number of arrival trips at different stations.

I have make a simple web page to show the distribution of the number of departure and arrival trips at different stations. You can visit it here for more details. Click on each station to see the number of departure and arrival trips.

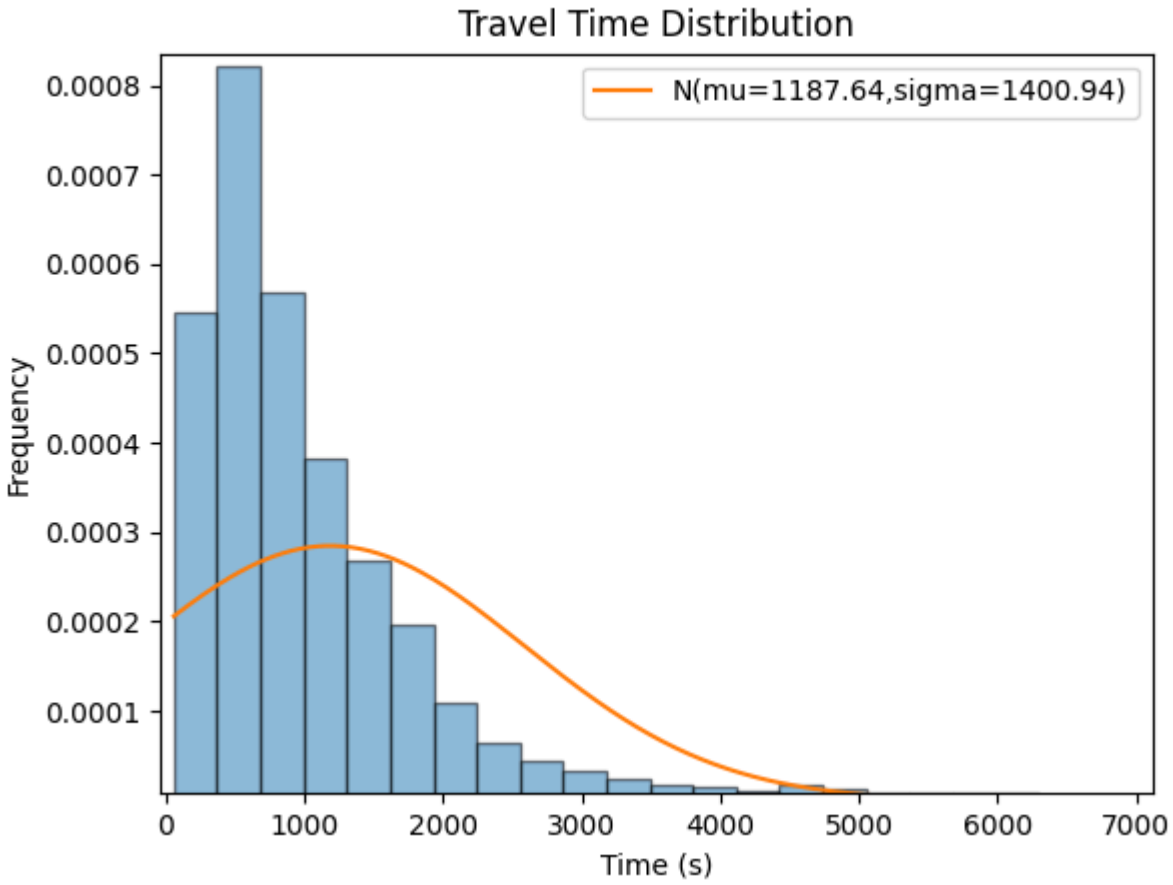
(3) What is the distribution of the trip distance (measured as straight-line Euclidean distance)? What will you conclude from this distribution?



Mean distance is about 2500 meters. Most trips are short, and the number of trips decreases as the distance increases.

(4) What is the distribution of the travel time (i.e., trip duration)?

1. For duration :



> Figure 2: The distribution of the travel time (i.e., trip duration). The x-axis represents the travel time, and the y-axis represents the frequency of trips.

We can find that the most frequent travel time is around 500 seconds, and the distribution is right-skewed. Most trips are short, and the number of trips decreases as the travel time increases.

2. For start and end time :

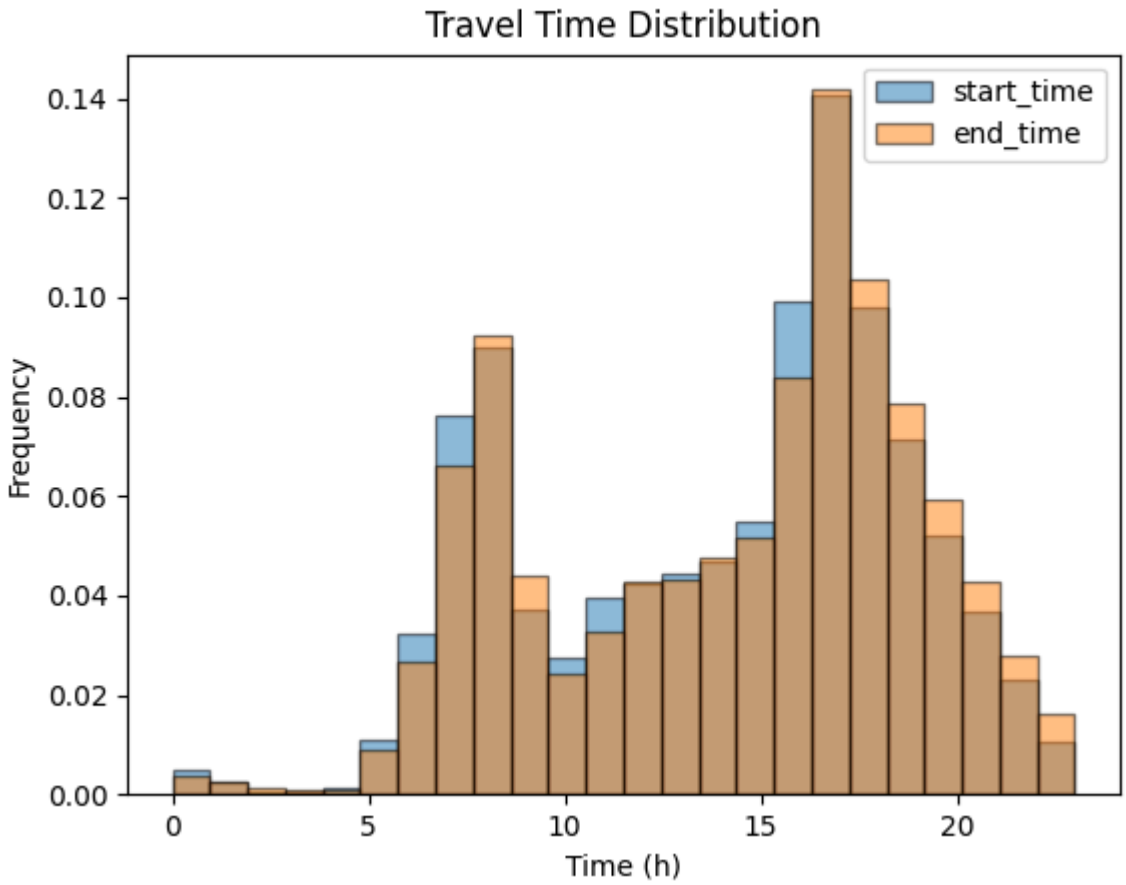
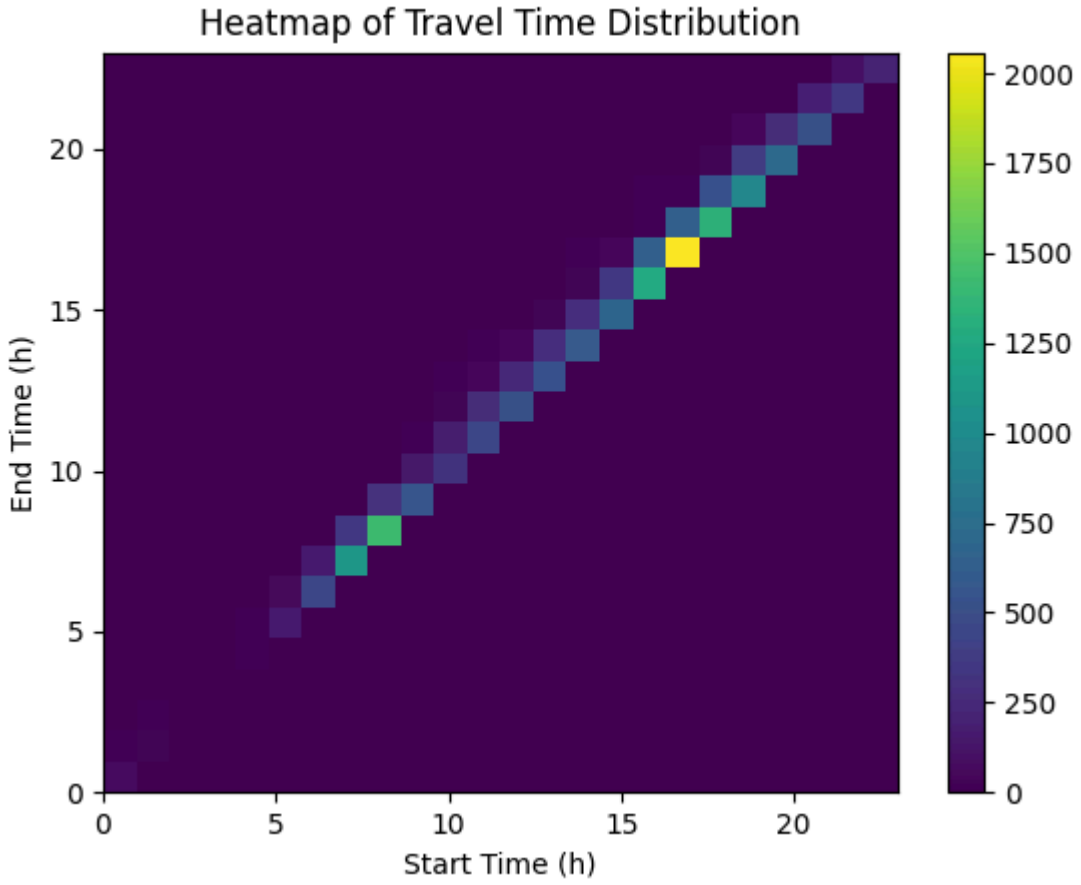


Figure 3: The distribution of the start and end time of trips. The x-axis represents the time, and the y-axis represents the frequency of trips. More light color represents more trips.



> Figure 4: The heatmap of the start and end time of trips. The x-axis represents the start time, and the y-axis represents the end time. The color represents the number of trips. More light color represents more trips.

We can find that there are two peaks at around 8 am and 5 pm, which is consistent with the results of the first question. Most trips are short, in another word, most trips have very close start and end times, and that is why the heatmap is mainly concentrated on the diagonal.

Task4 [25 points]:

Suppose the bike-sharing operator plans to manage efficiently by dividing bike stations into multiple service zones based on the distance between stations. Some clustering algorithms (e.g., DBSCAN, SVM) could be useful for the operator.

Please refer to this website to cluster all bike stations in Chicago using the Density-based Spatial Clustering of Applications with Noise (DBSCAN) algorithm packaged in scikit-learn. The maximum distance between two stations is 600 meters, and the number of samples in a neighborhood for a point to be considered as a core point is 2 stations. The other parameters are set as default. Please list the number of clusters and the station ids in each cluster in your report.

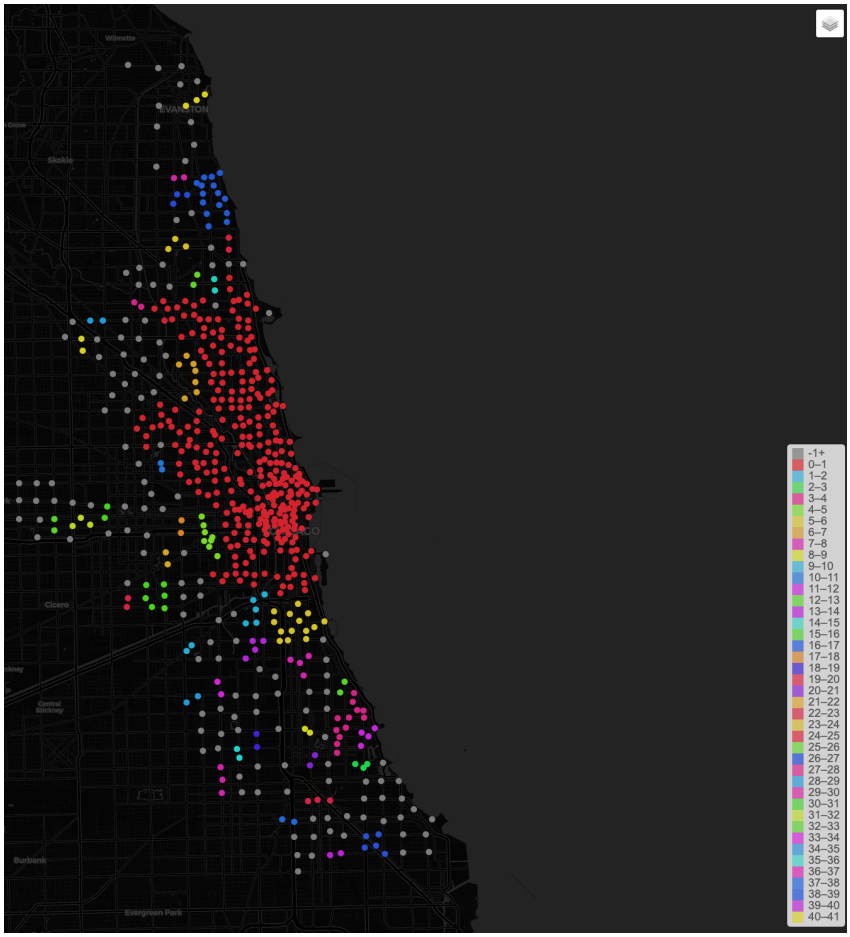
The number of clusters is 42, and the station ids in each cluster are as follows:

```
cluster,id
0,"[2, 3, 4, 5, 6, 7, 13, 14, 15, 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31,
32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43,
44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55,
56, 57, 58, 59, 60, 61, 62, 66, 67, 68, 69, 71,
72, 73, 74, 75, 76, 77, 80, 81, 84, 85, 86, 87,
88, 89, 90, 91, 92, 93, 94, 96, 97, 98, 99, 100,
103, 106, 107, 108, 109, 110, 111, 112, 113, 114,
115, 116, 117, 118, 119, 120, 123, 125, 126, 127,
128, 129, 130, 131, 132, 133, 134, 135, 136, 137,
138, 140, 141, 142, 143, 144, 145, 146, 152, 153,
154, 156, 157, 158, 159, 160, 161, 164, 165, 166,
168, 169, 170, 171, 172, 173, 174, 175, 176, 177,
178, 180, 181, 182, 183, 185, 186, 188, 190, 191,
192, 194, 195, 196, 197, 198, 199, 202, 205, 208,
210, 211, 212, 213, 214, 217, 218, 219, 220, 222,
223, 224, 225, 226, 227, 229, 230, 231, 232, 233,
234, 236, 238, 239, 240, 241, 242, 243, 244, 245,
246, 250, 251, 253, 254, 255, 256, 257, 259, 264,
268, 273, 274, 277, 282, 283, 284, 285, 286, 287,
288, 289, 290, 291, 292, 293, 294, 295, 296, 297,
298, 299, 300, 301, 302, 303, 304, 305, 306, 307,
309, 310, 311, 312, 313, 314, 315, 316, 318, 319,
320, 321, 323, 324, 327, 329, 330, 331, 332, 333,
334, 337, 338, 340, 343, 344, 346, 347, 349, 350,
359, 364, 365, 370, 374, 394, 414, 465, 475, 477,
478, 481, 482, 486, 502, 504, 505, 506, 507, 509,
511, 620, 621, 623, 624, 626, 627, 628, 635, 636,
638, 639, 657, 659, 666, 672, 673]"
1,"[9, 645]"
2,"[95, 428, 653]"
3,"[121, 248, 267, 322, 328, 416, 417, 418, 419, 420, 423, 426]"
4,"[122, 215, 261, 275, 317, 342, 383, 631]"
5,"[147, 148, 149, 150, 184, 193, 237, 263, 272, 335, 401, 402, 403, 405]"
6,"[162, 163, 228, 258, 308, 492, 493]"
7,"[179, 201, 407, 410]"
8,"[204, 421]"
9,"[206, 207, 209, 280, 339]"
10,"[216, 622]"
11,"[247, 345, 424, 425]"
```

```
12,"[252, 413]"
13,"[262, 278, 279, 548]"
14,"[325, 463]"
15,"[348, 439, 441, 442, 444, 445]"
16,"[353, 354, 432, 447, 449, 451, 453, 515, 517, 519, 520, 522, 523, 525, 660]"
17,"[381, 382]"
18,"[386, 593]"
19,"[393, 400, 652]"
20,"[430, 431]"
21,"[436, 437]"
22,"[440, 443]"
23,"[452, 455, 456]"
24,"[454, 458]"
25,"[462, 464]"
26,"[466, 467, 526]"
27,"[474, 476]"
28,"[495, 630]"
29,"[514, 527]"
30,"[528, 534]"
31,"[537, 539, 545]"
32,"[540, 543]"
33,"[549, 552]"
34,"[551, 594]"
35,"[562, 647]"
36,"[563, 565, 566]"
37,"[568, 572]"
38,"[577, 578, 583, 588, 665]"
39,"[584, 585]"
40,"[590, 591]"
41,"[596, 603, 605]"
```

Bonus for Task4 [+10 points]:

Please visualize the clusters using matplotlib or any Python packages you prefer. Here is a reference about how to visualize clusters. If you complete this Bonus part, please embed the figure into your report.



> Figure 5: The clustering results of bike stations in Chicago. The color represents different clusters.

You can visit it here for more details. Try to choose the cluster layer to see the clustering results.

For better visualization, I wrote a simple random color generator. The generator uses the HSL color space to control the color of adjacent categories in different hues, and uses a seed to ensure that the generated colors are the same each time. You can check the source code for more details.

Because the minimum category is only 2 instead of 3, otherwise we can use the point set centroid algorithm to get the center of each category, and use the convex hull algorithm to get the boundary of each category.