CitiBike: Case for Last-Mile Connectivity in the NYC MTA System

● ● ●

By: Aanvi Goel

Overview

- The Metropolitan Transport Authority serves as a daily ridership of more than 5 million and connects 472 subway stations

- The commuters occasionally require a second mode of transportation to reach their destination, also known as the last-mile connectivity

Project objective:

Evaluate the viability of CitiBike to improve the last-mile connectivity for the MTA system

Understanding the data

1. MTA Turnstile Data

```
mta_data.head()
```

	C/A	UNIT	SCP	STATION	LINENAME	DIVISION	DATE	TIME	DESC	ENTRIES	EXITS
0	A002	R051	02-00-00	59 ST	NQR456W	BMT	06/25/2022	00:00:00	REGULAR	7729818	2718760
1	A002	R051	02-00-00	59 ST	NQR456W	BMT	06/25/2022	04:00:00	REGULAR	7729828	2718769
2	A002	R051	02-00-00	59 ST	NQR456W	BMT	06/25/2022	08:00:00	REGULAR	7729835	2718793
3	A002	R051	02-00-00	59 ST	NQR456W	BMT	06/25/2022	12:00:00	REGULAR	7729863	2718853
4	A002	R051	02-00-00	59 ST	NQR456W	BMT	06/25/2022	16:00:00	REGULAR	7729932	2718904

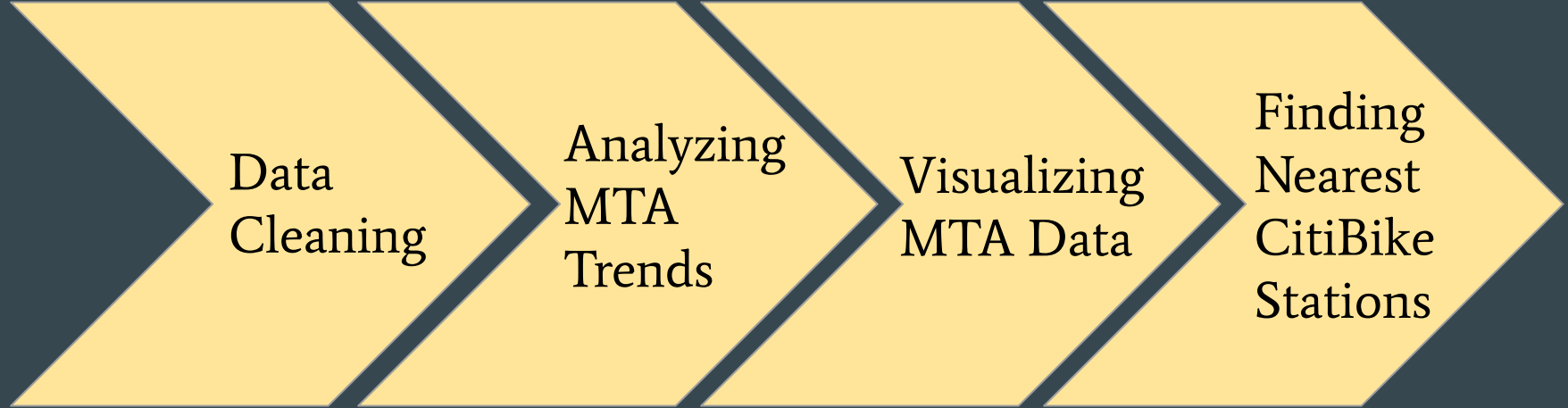# 2. MTA Station Data

	Station ID	Complex ID	GTFS Stop ID	Division	Line	Stop Name	Borough	Daytime Routes	Structure	GTFS Latitude	GTFS Longitude	North Direction Label	South Direction Label	ADA	ADA Direction Notes	ADA NB	ADA SB
0	1	1	R01	BMT	Astoria	Astoria-Ditmars Blvd	Q	N W	Elevated	40.775036	-73.912034	NaN	Manhattan	0	NaN	NaN	NaN
1	2	2	R03	BMT	Astoria	Astoria Blvd	Q	N W	Elevated	40.770258	-73.917843	Ditmars Blvd	Manhattan	1	NaN	NaN	NaN
2	3	3	R04	BMT	Astoria	30 Av	Q	N W	Elevated	40.766779	-73.921479	Astoria - Ditmars Blvd	Manhattan	0	NaN	NaN	NaN
3	4	4	R05	BMT	Astoria	Broadway	Q	N W	Elevated	40.761820	-73.925508	Astoria - Ditmars Blvd	Manhattan	0	NaN	NaN	NaN
												Astoria -					

3. CitiBike Data

```
url ='https://gbfs.citibikenyc.com/gbfs/en/station_information.json'
```
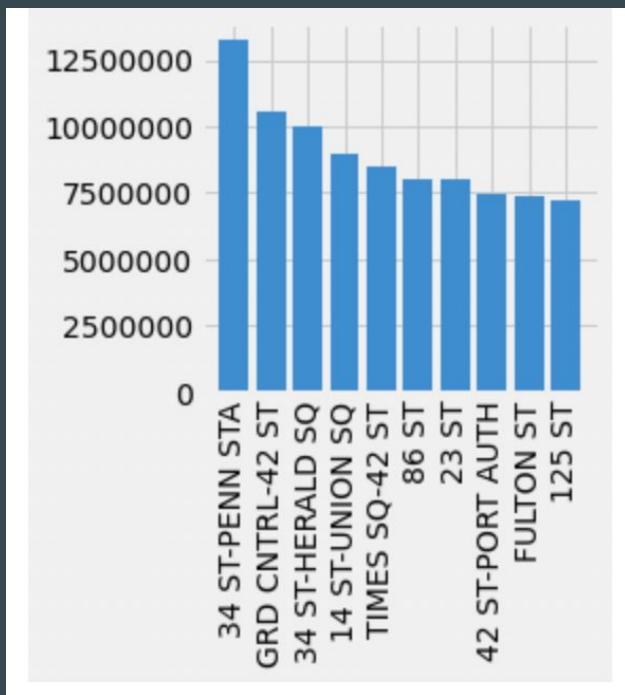
	has_kiosk	lon	name	external_id	station_id	lat	rental_methods	station_type	legacy_id	capacity	region_id	electric_bike_surcharge_wa
0	True	-73.993929	W 52 St & 11 Ave	66db237e-0aca-11e7-82f6-3863bb44ef7c	72	40.767272	[CREDITCARD, KEY]	classic	72	55	71	F
1	True	-74.006667	Franklin St & W Broadway	66db269c-0aca-11e7-82f6-3863bb44ef7c	79	40.719116	[CREDITCARD, KEY]	classic	79	33	71	F
2	True	-74.000165	St James Pl & Pearl St	66db277a-0aca-11e7-82f6-3863bb44ef7c	82	40.711174	[CREDITCARD, KEY]	classic	82	27	71	F
3	True	-73.976323	Atlantic Ave & Fort Greene Pl	66db281e-0aca-11e7-82f6-3863bb44ef7c	83	40.683826	[CREDITCARD, KEY]	classic	83	62	71	F
4	False	-74.001497	W 17 St & 8 Ave	66db28b5-0aca-11e7-82f6-3863bb44ef7c	116	40.741776	[CREDITCARD, KEY]	classic	116	0	71	F

Methodology

Data Cleaning

Analyzing MTA Trends

Visualizing MTA Data
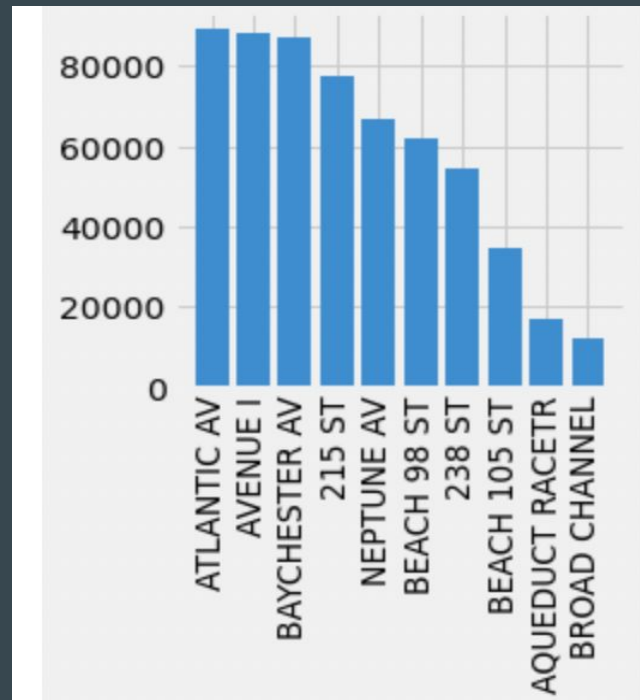
Finding Nearest CitiBike Stations

MTA Ridership Trends
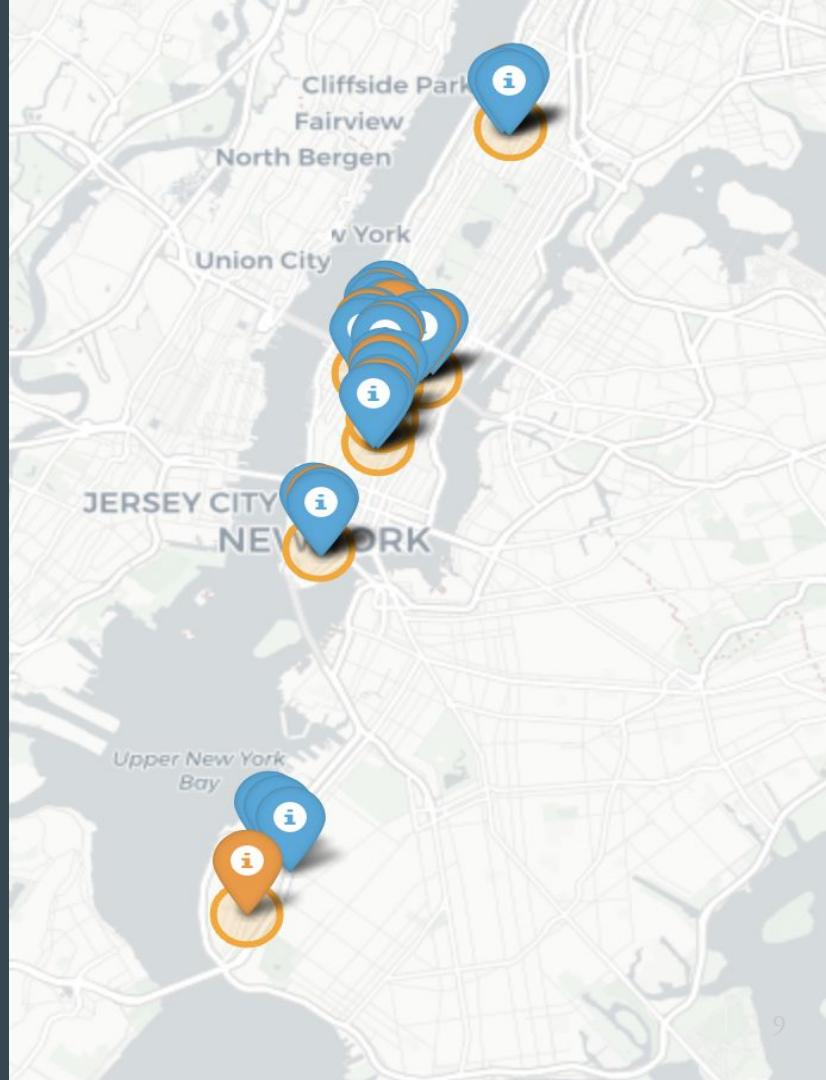
Stations with Most Daily Exits

Stations with Least Daily Exits

Visualizing Station Locations
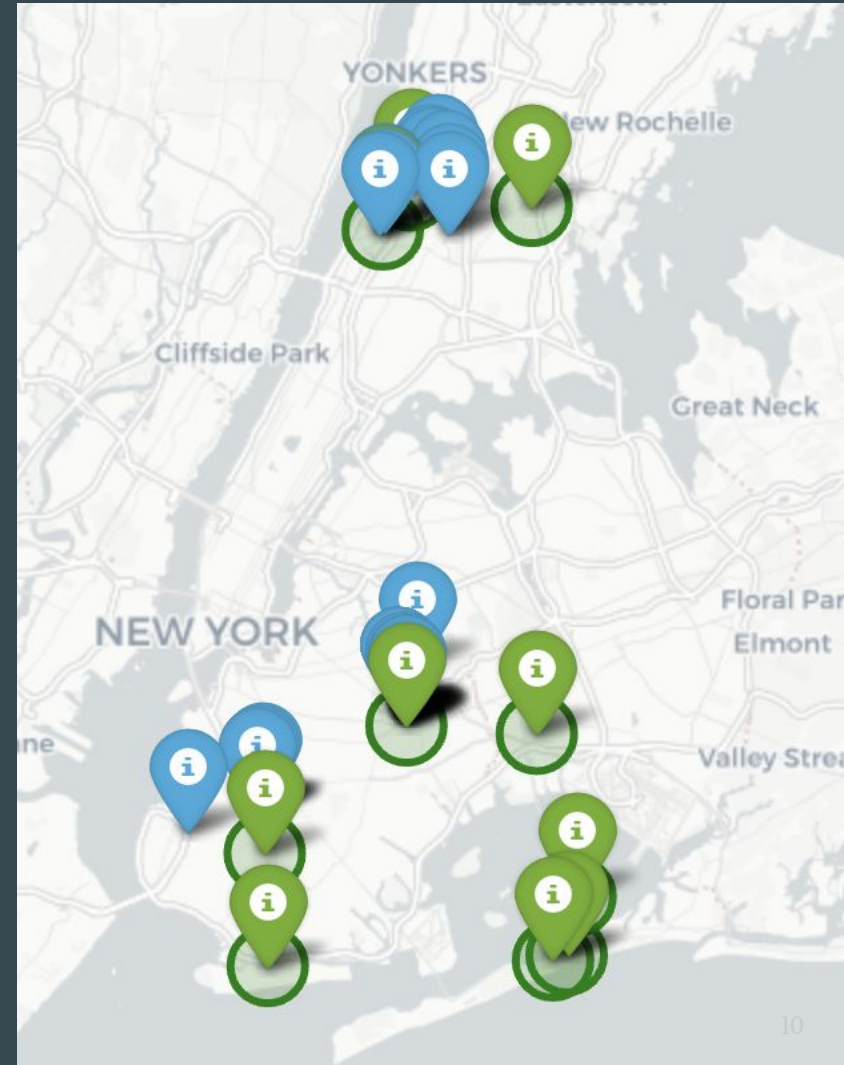
Stations with Most Daily Exits

- Setting the filter radius = 0.5 miles
- MTA Stations are in Orange with a radius marker
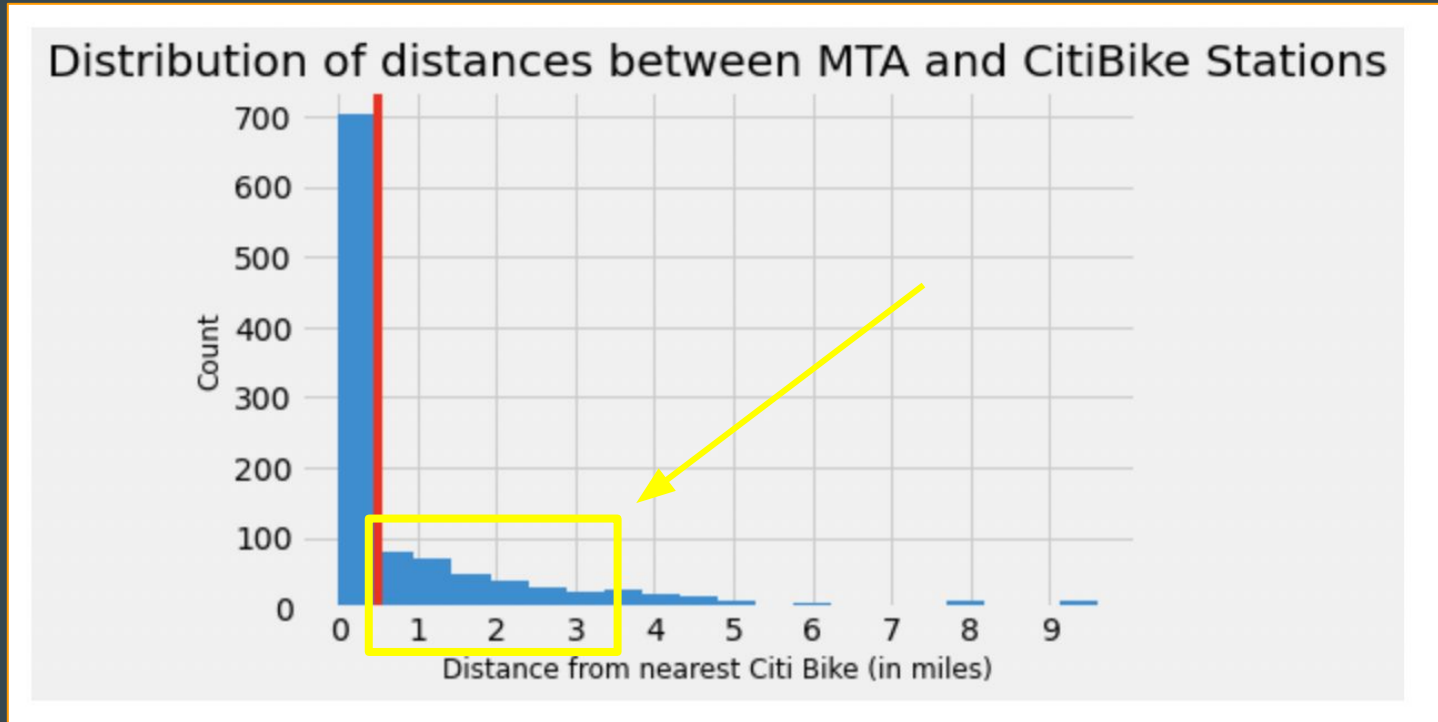- The nearest CitiBike stations for each MTA station are plotted in Blue
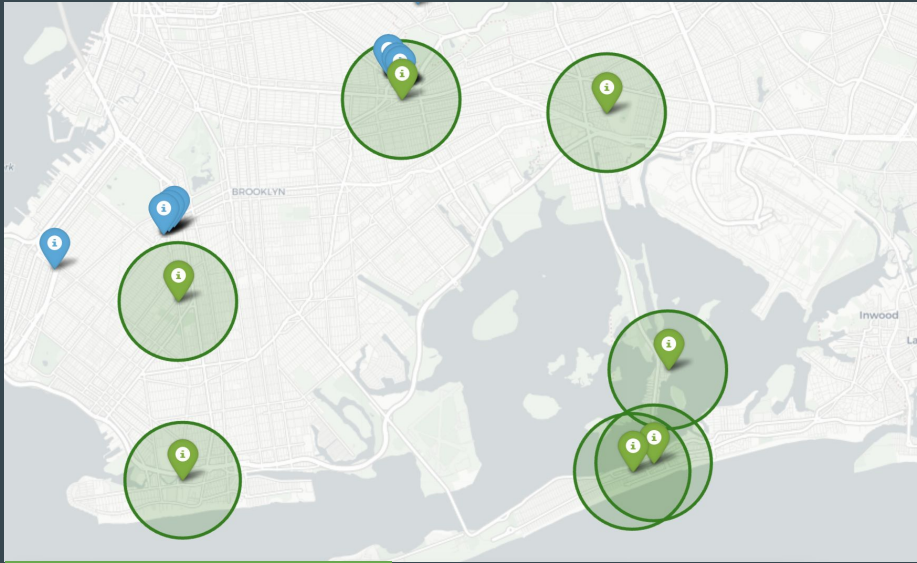
Visualizing Station Locations

Stations with Least Daily Exits

- Setting the filter radius = 1 mile
- MTA Stations are in Green with a radius marker
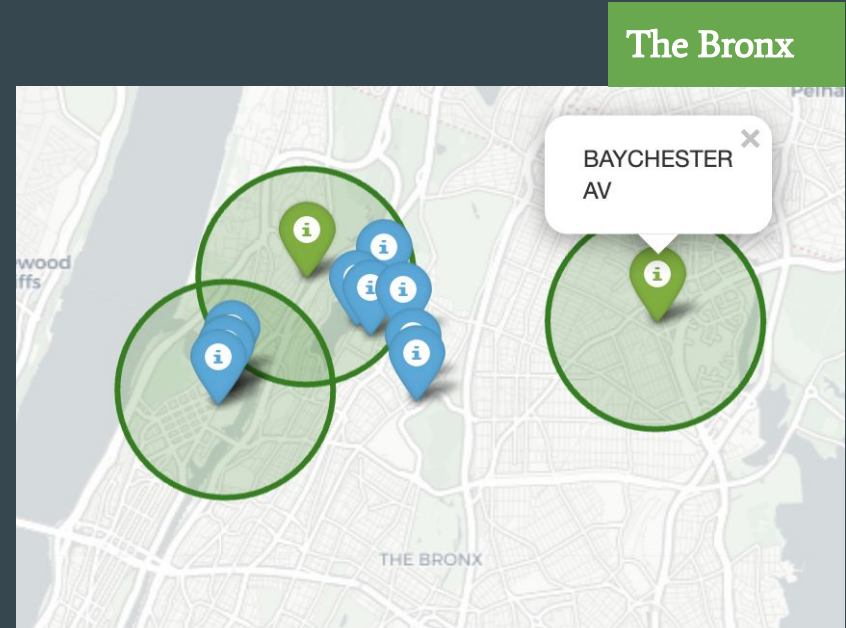- The nearest CitiBike stations for each MTA station are plotted in Blue

Spotlight: Stations with Least Daily Exits

Distribution of distances between MTA and CitiBike Stations

Brooklyn and Queens

The Bronx

BAYCHESTER AV

Conclusions

1) Areas further away from city-center, have fewer CitiBike stations

2) CitiBike Stations in boroughs such as Bronx, Brooklyn and Queens have >1 mile distance between the MTA stations and CitiBike stations

Recommendations

To improve the commuter experience and CitiBike ridership:

→ CitiBike should build new bike stations in low-presence areas such as Bronx, Brooklyn and Queens

→ For the areas with existing CitiBike stations, CitiBike should increase the number of stations to be within 0.5 miles of the MTA stations

Future Work

- Analyze Citi Bike station/vehicle status data to analyze current bike availability per CitiBike station

- This data can be combined with the analysis on MTA Station ridership improve capacity based on day of week & time of day

Link to CitiBike Data:
https://github.com/NABSA/gbfs/blob/master/gbfs.md#geofencing_zonesjson

APPENDIX

Data Cleaning Process:

- Correcting negative counters
- Removing rows with overflow counters as they were smaller in number and had minimal impact if dropped

```
#Based on the above data there are 2 anomalies - 1) Negative counters 2) Counter overflow

#Taking the abs of DAILIY EXIT values for the turnstiles that are counting in reverse
mta_daily_exits['DAILY_EXITS']=abs(mta_daily_exits.DAILY_EXITS)

#Since the numbers of rows with counter overflow is relatively low, I have decided to drop
the rows with counts>5000
mta_daily_exits=mta_daily_exits[mta_daily_exits.DAILY_EXITS < 5000]

mta_daily_exits.DAILY_EXITS.describe()
```

- Dropping the 'bad' counters - defined as counters which may have malfunctioned and did not increase count over the duration of days
- Dropping data from TRAM between Manhattan st and Roosevelt Island St to only consider sunway data

```
#There are 2 stations for the Tram between Manhattan and Roosevelt Island. We'll also remove
these two rows.
tram=mta_daily_exits['C/A'].isin(['TRAM1','TRAM2'])
mta_daily_exits=mta_daily_exits[~tram]
```

MTA Exit Traffic Per Day of Week
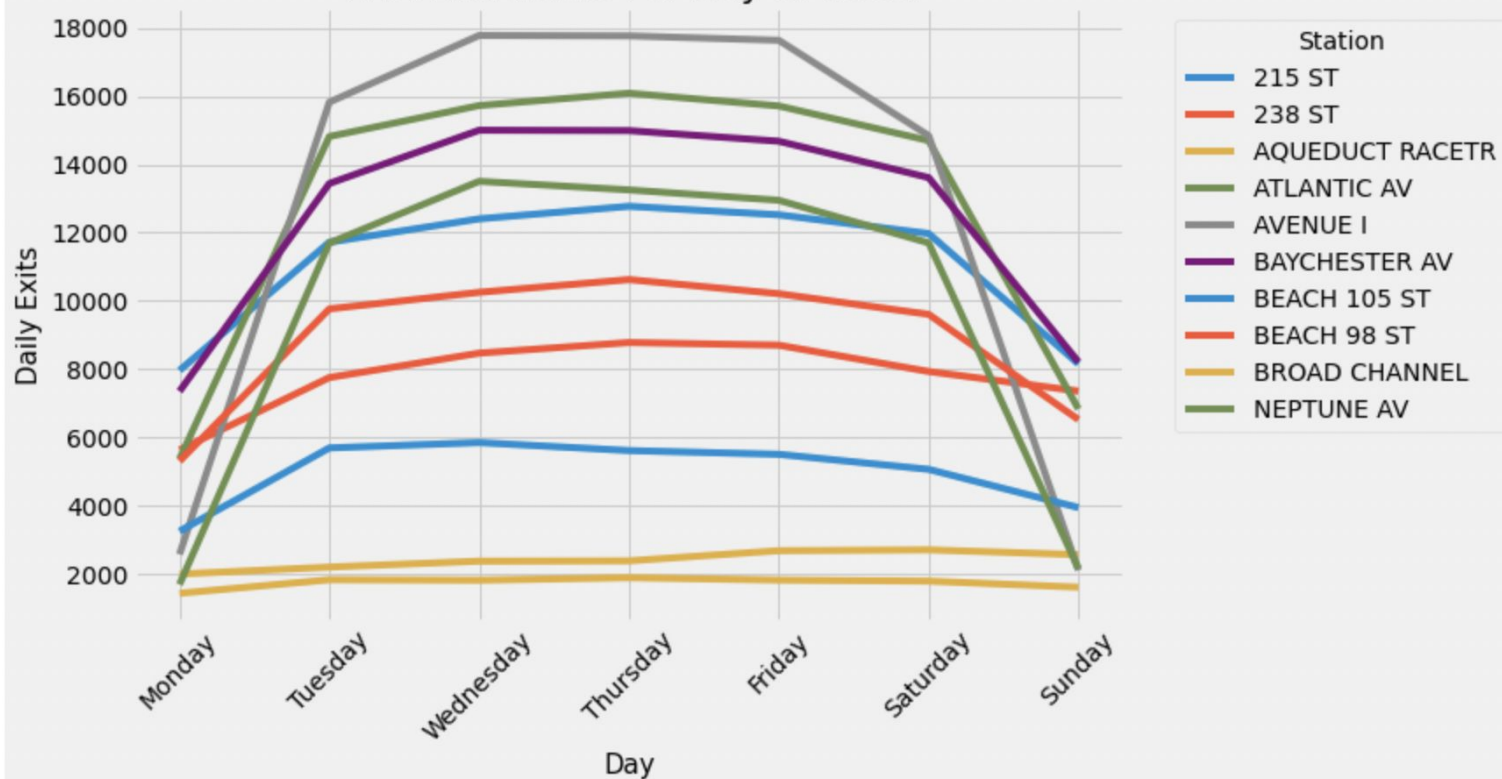
```
#Finding respective matches for STation names in the MTA Turnstile Data and the MTA Station Data using FuzzyWuzzy
https://www.geeksforgeeks.org/how-to-do-fuzzy-matching-on-pandas-dataframe-column-using-python/

match1 = []
match2 = []
p= []
list1 = station_to_date.STATION.tolist()
list2 = station_loc_data.STOP_NAME.tolist()

threshold = 50

for i in list1:
    match1.append(process.extractOne(i, list2, scorer=fuzz.ratio))
station_to_date['MATCHES'] = match1

for j in station_to_date['MATCHES']:
    if j[1] >= threshold:
        p.append(j[0])

    match2.append(','.join(p))
    p= []

station_to_date['MATCHES'] = match2
```

```
#Creating an algorithm using the Haversine formula to find distances between MTA Stations and Nearest 3 CitiBike stations

import haversine as hs
from haversine import Unit

def get_distances(df1, df2):
    dist_df=pd.DataFrame()

    for i,row1 in df1.iterrows():
        for j,row2 in df2.iterrows():
            loc1 = [row1['LAT'], row1['LONG']]
            loc2 = [row2['lat'], row2['lon']]

            dist= hs.haversine(loc1,loc2, unit=Unit.MILES)
            df2.at[j,'dist']=dist
            df2.at[j,'station_name'] = row1['STATION']

        dist_append= df2.sort_values(by=['dist'])[0:3]
        dist_df=pd.concat([dist_df,dist_append], ignore_index=True)


    return dist_df
```