Mohammed sareh assiri
Project proposal

## Overview:

In this project, I plan to recommend a shortlist of stations with the highest amount of foot traffic. For this project, I primarily use the MTA turnstile data to come up with our shortlist, but I also consider weather data from the NOAA to see if bad weather affected the ranking of certain subway stations.

## Goal:

My project involved exploring the MTA turnstile data.

The goal uses this data to recommend placement of street teams for some organizations so they can get interested NYC commuters signed up to their email list.

Datasets:

I will write a Python script to grab the last three months of data from the MTA Turnstile repository, which stores the counts of entries and exits for each turnstile in each subway station in the MTA transit system. The data is added as text files to this repository each week on Saturday and contains readings of the turnstile counters at approximately four-hour intervals. The MTA defines the attributes of these turnstile datasets in a data dictionary per the following:

C/A : Control Area (A002)

UNIT: Remote Unit for a station (R051)

SCP : Subunit Channel Position represents a specific address for a device (02-00-00)

STATION: Represents the station name where the device is located

LINENAME: Represents all train lines that can be boarded at this station

DIVISION: Represents the Line originally the station belonged to BMT, IRT, or IND

DATE: Represents the date (MM-DD-YY)

TIME: Represents the time (hh:mm: ss) for a scheduled audit event

DESC: Represent the “REGULAR” scheduled audit event (Normally occurs every 4 hours). Audits may occur more than 4 hours due to planning, or troubleshooting activities. Additionally, there may be a “RECOVER AUD” entry: This refers to a missed audit that was recovered.

ENTRIES: The cumulative entry register value for a device

EXIST: The cumulative exit register value for a device

I also create the following columns:

entry_diff: Total entries for a given time interval (difference between rows for “ENTRIES”)

exit_diff: Total exits for a given time interval (difference between rows for “EXITS”)

TOTAL_TRAFFIC: Sum of entry_diff and exit_diff

DOW: Day of the week for a given date of the counter reading

For weather data, I will request the last three months of weather observations from the NOAA's National Climatic Data Center, using CENTRAL PARK (Station ID: GHCND: USW00094728) as a proxy for the entire city's weather. I will select the only two attributes in this data set that were relevant to rainfall, which is as follows:

DATE: Date of observation

PRCP: Inches of precipitation for the day of observation

## Assumptions:

Counter Values: I assumed the "ENTRIES" and "EXITS" columns reflected cumulative counts that could only increase as time moved forward. Thus, I will remove any rows with negative values in my "entry_diff" and "exit_diff" columns (approximately 1.4% of the rows).

Target Metric: I did not differentiate between "ENTRIES" and "EXITS" for a station, but rather relied on "TOTAL_TRAFFIC" to determine which station would have the most foot traffic at a given time and thus be the best place for the street teams to visit.

Outlier Counts: I assumed that any "entry_diff" or "exit_diff" value over 86,400 (based on total seconds in a day) was an outlier that needed to be removed.

Tools:

To analyz the data I will use different tools like SQl Browser,Excel,Jupytor python language.

And I will use a different library for Example : numpy,pands,matplotlib,statistics.

## Recommendation:

To help the team make decisions about where to allocate their resources at their available times, I built a scheduler for them in a jupyter notebook. The tool allows the user to enter three parameters: day of the week, time of day (in 4-hour blocks), and weather forecast (i.e. "rain" or "no rain"). Based on these conditions, the scheduler outputs the ten stations with the highest average foot traffic.