

# Project Proposal

New York City is one of the largest hotspots in the world that mostly relies on metro for transportations, one of the issues with metro stations is the visitors distribution. Where some stations are visited more than others, which creates another issue with resource management, distributing these resources should be heavily reliant on number of visitors for each station. Therefore, we propose using this data to understand and visualize the number of visitors for each station, to help us distribute resources correctly.

## Question/need:

Resource management for the following:

- Ads placements.
- Human resources distribution.
- Restaurants and shops distribution.
- Turnstile distribution.

Utilizing resource will help:

- The government to decrease unnecessary costs.
- Improve the experience for visitors.
- Help companies on advertisements locations.

## Data Description:

The New York subway MTA turnstile data is a series of data files containing cumulative number of entries and exits by station, turnstile, date and time. Data files are produced weekly, data records are collected typically every 4 hours with some exceptions.

In this analysis we use data between May 01, 2021 and October 02, 2021.

Variables included in initially processed data:

- C/A = Control Area (e.g., A002)
- unit = Remote Unit for a station (e.g., R051)
- SCP = Subunit Channel Position represents an specific address for a device (e.g., 02-00-00)
- station code = C/A + unit, locating a station
- turnstile = C/A + unit + SCP, locating a turnstile
- Station = Represents the station name the device is located at
- date = Represents the date (MM-DD-YY)
- time = Represents the time (hh:mm:ss) for a scheduled audit event
- datetime = date + time (MM-DD-YY hh:mm:ss)
- DESC = Represent the "REGULAR" scheduled audit event (Normally occurs every 4 hours)
- entries = The cumulative entry register value for a device
- exits = The cumulative exit register value for a device

## Tools:

Python, Pandas, SQL, Matplotlib, Jupyter notebook and NumPy