EDA Project

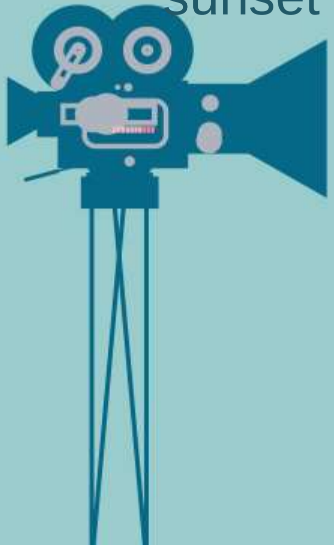
Scenes Filming

By: Shahad Abdulsalam

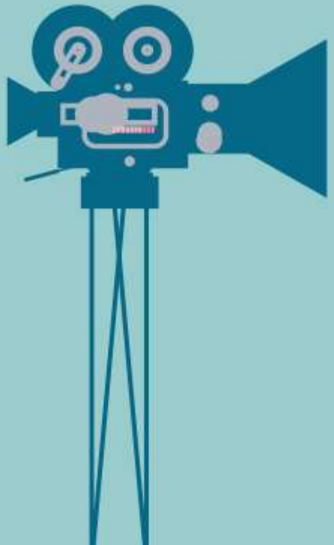
The Problem:

A production company seeking for 3 stations to filming 3 scenes.

- The **1st** scene should be filming at the Christmas night with overcrowded station.
- The **2nd** scene should be filming at the early dawn hours with few entries/exists.
- The **3rd** scene should be filming on one of the winter days at the sunset moment with moderate traffic.

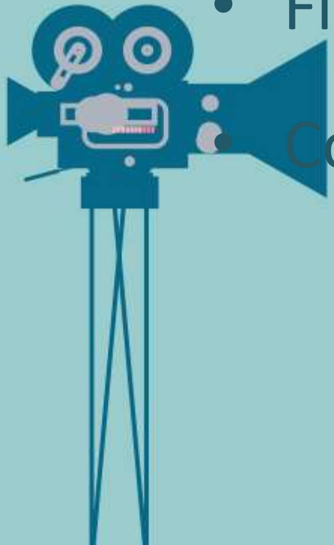


DATA	TOOLS	Deliverables
MTA turnstile data	• sqlite • python libraries (pandas, numpy, os) • python visualization libraries (seaborn, matblotlip)	Determine the suitable stations and days for each scene

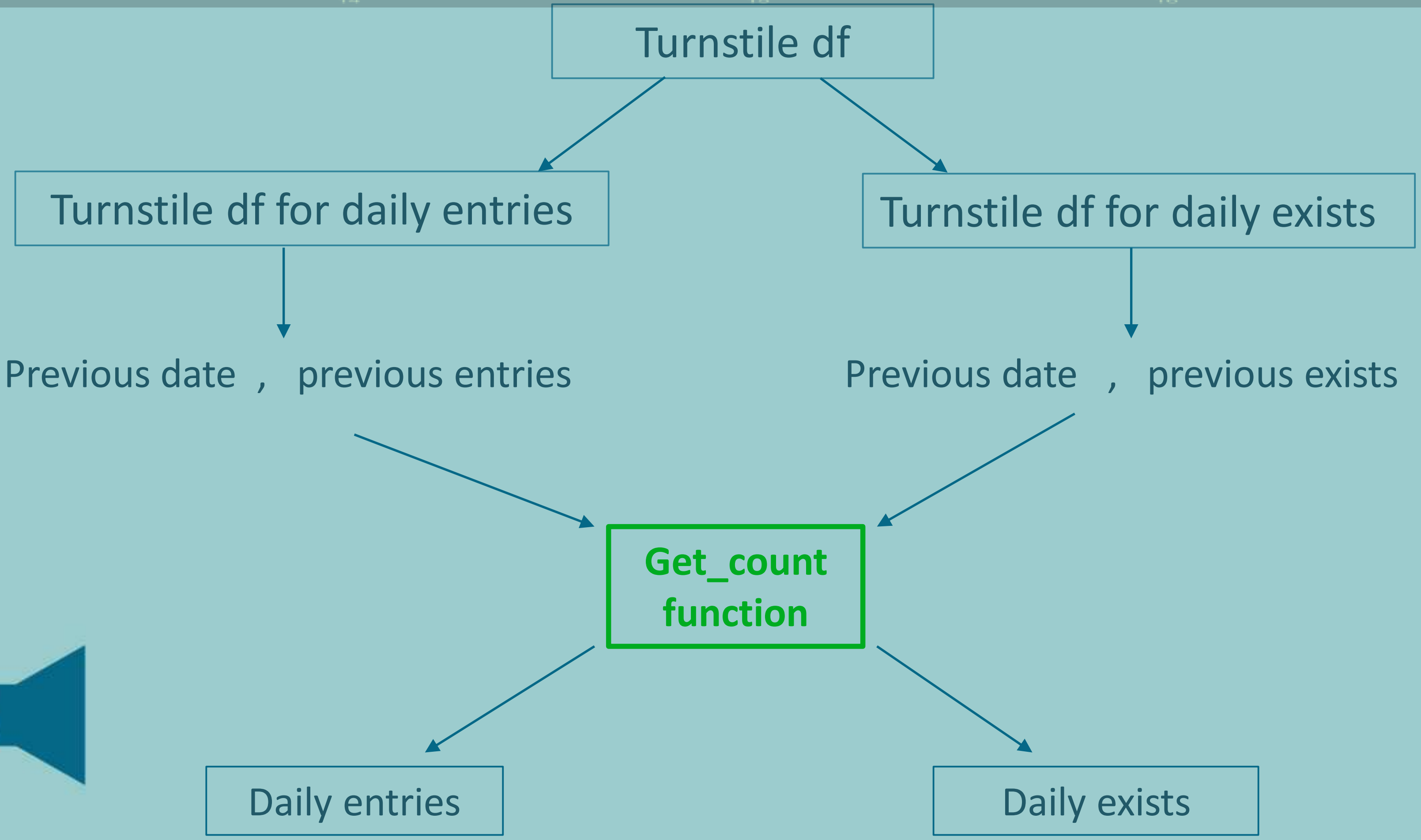


Data Cleaning/ Preprocessing

- Drop the null values.
- Drop The duplicates.
- Remove the leading and trailing spaces from columns titles.
- Fix the reversed counter if its exist.
- Convert date/ time columns to a datetime type



Creating two dataframes:



Turnstile df

Turnstile df for daily entries

Turnstile df for daily exists

New columns

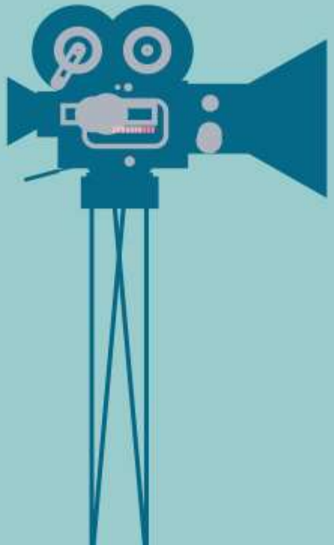
Previous date , previous entries

Previous date , previous exists

Get_count function

Daily entries

Daily exists



What does get_count function do?

before we use get_count function, we’ve to check if the counter has a reversed values in some rows .

Reversed counter is previous entries < entries or previous exists < exists

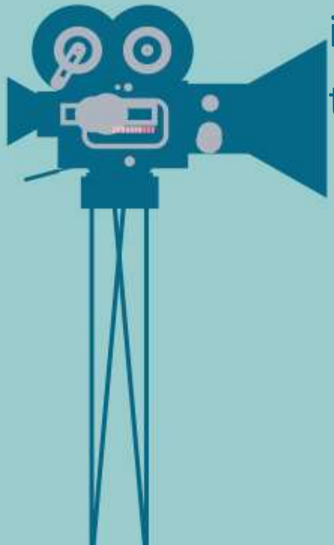
So get_count() will fix this problem by creating a daily count column for each dataframe.

Daily count columns:

subtract previous count from the current count.

if its less than 0 it will turn it into positive value by multiply it by (-)

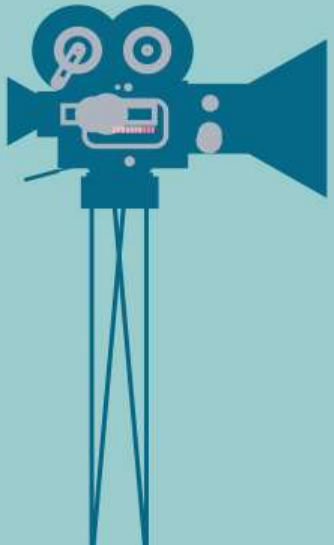
if counter greater than max_counter(~1million) it will return the minimum of (previous, current)



Filtering :

1st scene

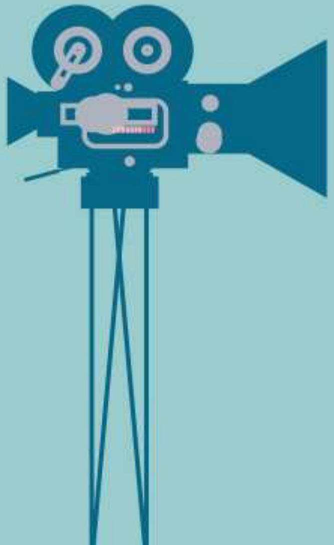
- Sort dataframes descending by (daily entries , daily exists) columns
- Filter them by day = (12/24/2017) time = (20:00, 21:00, 22:00, 23:00)
- Concatenate them



The 1st location will be in station (14 ST-UNION SQ) Christmas night at 20:00:00

2nd scene

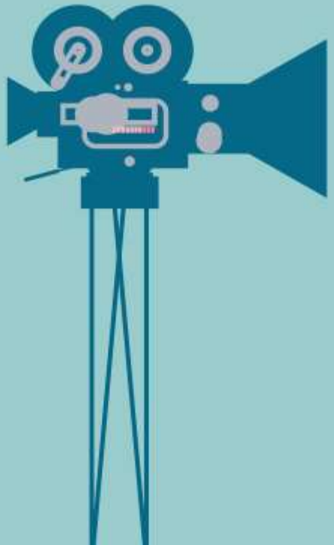
- Sort dataframes ascending by (daily entries , daily exists) columns
- Filter them by any day I chose (03/01/2017) time = (03:00, 04:00, 05:00, 06:00)
- Concatenate them



The 2nd location will be in station (GUN HILL RD) on 03/01/2017 at 3:00-4:00 AM

3rd scene

- Sort dataframes descending by (daily entries , daily exists) columns
- Filter them by day = (02/07/2017)   time = (17:00, 18:00)
- Concatenate them and take the median of (daily entries , daily exists)



The 3rd location will be in station (34 ST-PENN STA) on 02/07/2017 at 17:00 the sunset moment

Thank you