Capstone Project 1: Data Wrangling

Weather Data Cleaning

The 2016 NYC weather data was originally acquired from the Wunderkind API, and provided here.
https://www.kaggle.com/meinertsen/new-york-city-taxi-trip-hourly-weather-data

First we dropped all the columns that we do not think that we'll use.

As we check the data, we notice that it's fairly well organized. The rain and snow columns are are separate boolean columns. The data would be easier to read if it was a single column with a categorical values, so we combine the two columns into one precip_type column with values 'rain', 'snow', or an empty string for clear weather.

From examining the weather data, we notice something is off with the temperatures for certain days. Plotting the daily average temperatures for every month, we quickly notice that the for the days before the 12th of the month, the average temperatures rise and fall as you would expect throughout the year, and after the 12th of the month the average temperatures spread out and differ similar to how you expect them to differ between the various months. From this we can conclude that for the days that are before the 12, the month and day have been accidentally reversed in the data. After correcting for this, we plot the average temperatures again and we see that it now somewhat behaves how we would expect.

Now that the dates are correct, we know that our taxi data is very large so we will try to pick a handful of days so that we can have some rainy days, some snow days, and some clear days. We can choose these days by looking at the days with the most snow and rain readings. We need to be careful in choosing days however, as we can see that Jan 23 is the day with the most snow, but a quick google search reveals that it is a day of record blizzard snowfall and not necessarily a good representative of typical snowfall in New York City. We should also try to avoid certain days with parades and large scale road closures according to this list (https://www.bizbash.com/home/media-gallery/21076886/new-yorks-top-100-events-2016-parades-festivals-holiday-events#image-5c7f40851c74c5dab7c8ee7b)

Using this list of days, we can create a dataframe of only the days that we want to look at. However, since a lot of the days have their first weather reading occur at 00:51, the last reading of the previous day would be helpful in the case that the time of a taxi trip occurs closer to 11:51. So for each day that we have, we can get the last recording of the previous day and add it to our dataframe.

The simplified weather data is saved as weather_edit.csv.

Taxi Data Cleaning

The taxi data in NYC was acquired from NYC OpenData
https://data.cityofnewyork.us/dataset/2016-Yellow-Taxi-Trip-Data/uacg-pexx
With this set being 131M rows, some bandwidth issues were experience so we decided to download only the taxi trips where VendorID = 1, giving us 61M rows of taxi trip data.

First we drop the columns that we do not need, mainly leaving us with the pickup and dropoff datetimes and the trip distance. With the goal of exploring the relationship between weather conditions and automobile travel times in NYC, we calculate the average speed for each trip. To get the time for each entry, we first convert the pickup and dropoff columns to datetime objects, and subtract them to find the time difference. Then we take the trip_distance and divide by the time to get the average speed.

As we check the speed column, some values make no sense. The max is inf and the min is negative. So we drop the rows where the speed is less than 1, as well as greater than 60.

Using the same list of days we can keep only the taxi trips in those days.

The simplified taxi data is saved as taxi_edit.csv.

Merging the Data

With the right days in the taxi trip data, and the weather data, now we can merge them together on the pickup column. Using data.merge_asof(), we do a left join on the taxi data, matching each taxi trip with its nearest weather recording.

Save the merged data as data_merged.csv.