Capstone Two: Project Proposal

The NYC 2019 Mobility Report indicated an increase in traffic congestion in each New York borough for the past ten years. Public policy initiatives, such as greater access to public transit, will not immediately alleviate congestion in the city. To reduce traffic now, I can create a model that predicts a time duration for any Yellow Cab taxi in any part of the city based on features in the TLC Trip Record Data. This can help spread the distribution of traffic throughout the day.

Our data is based on 2019 TLC Record data. Our data source is a collection of 12 CSV Files, organized monthly. There are nearly 267 GB of data. We are given a dataset with more than a million objects. There are nineteen features:

- **Vendor_ID** (int64). The provider associated with the trip record. There are 2 taxi companies.
- **pickup_datetime** (datetime64[ns]). The date and time the trip started, measured by the time when the meter was engaged.
- **pickup_longitude** (float64). The longitude where the passenger entered the taxi, measured by the meter was engaged.
- **pickup_latitude** (float64). The latitude where the passenger entered the taxi.
- **dropoff_datetime** (datetime64[ns]). The date and time when the passenger arrived at their destination.
- **dropoff_longitude** (float64). The longitude of the passenger's destination
- **dropoff_latitude** (float64). The latitude of the passenger's destination
- **passenger_count** (int64). The number of passengers in the taxi.
- **Store_and_fwd_flag** (bool). Indicates whether the trip record was held in-vehicle memory before sending to the vendor because the vehicle did not have a connection to the server.
- **Fare_amount** (float64). The cost of the trip
- **Mta_tax** (float64). The tax paid from the trip
- **Tip_amount** (float64). The tip given after the trip
- **Tolls_amount** (float64). The cost of using Toll booths during the trip.

Additionally, we would want to create a **trip_duration** (float64) outcome variable using the difference of the dropoff and pickup datetime features.

The dataset is huge, and there are around 267 GB of data. Since there is so much data, I need to keep in mind data storage efficiency. The data is also in file-based storage.

Our dataset does not provide information on events that limit the flow of traffic, such as construction or holidays. I could create a **holiday** (bool) feature that indicates whether or not the pickup and dropoff datetime objects fall on a Holiday. Additionally, I could create a **day of the week** feature. For information on city-planned events that could affect traffic flow, we would need another dataset that provides information on the type of events scheduled, their duration, and location.

The datasets account for trip distance but not the routes taken by the taxi drivers. This feature could be helpful for taxi services looking to cut commute times so they can serve more customers at high-density hours.

Many services would find our model useful. Taxi services can use their model to anticipate where and when traffic density is the greatest. This allows for taxi services to better locate their drivers to get the most amount of business possible. Any business located in NYC that frequently uses taxis can better coordinate meetings so that they minimize commute time.

An initial plan to solve the time duration problem must first tackle the issue of storage efficiency. Each file should be merged so that all data can be accessed from the same dataframe. The datetime objects can be parsed to create days of the week features, month features, and separate time features. Pandas will be the main package used throughout this project. Since we are dealing with time-series data, we can use the Prophet library to handle correlation and other techniques that are affected by time. If deemed necessary, NYC Open Data features datasets from their City Planning office. Our dataframe, merged with a dataset on construction and city planning information, could lead to a more accurate model. Our outcome variable is the time duration, which is the difference between dropoff and pickup times. Errors in the data should be evaluated before EDA.

Given datetime objects, we can analyze the frequency of traffic over time. Latitudinal and Longitudinal data allows us to create choropleths to visualize the average time duration for Taxis throughout NYC. I could use a correlation matrix to root out multicollinearity (Fare_amount and Mta_tax, etc.) so that the model is interpretable. Afterwards, potential models will be evaluated based on their performance against the mean value, linear regression and each other. The model with the best performance on testing data and lowest error will be implemented.

The Github repository will include code for Data Wrangling, EDA, and Modelling. Documentation will be through a Project Report and a Slide Deck.