report

Predicting Hourly Taxi Demand¶

Summary¶

In this report, we will make hourly prediction of taxi demand in the Bronx, and determine the optimal number of taxis required accordingly. The data at hand is the historical trip data from taxis, combined with hourly weather data. The potential challenges and oppertunites of this analysis include:

- data quality:

  the data need to be cleaned and processed to remove bias arising from technical errors and outliers of data.

- data smoothing:

  the original time series of hourly demand can be noisy for trend and seasonality recognition. The occational zero value of hourly demand around mid-night can also be problematic for the analysis. Therefore, data smoothing is necessary prior to modeling.

- seasonality:

  The taxi demand is subject to, but not limited to, the followings seasonalities:

  - hour of the day: the taxi demand would surge during rush-hours.
  - day of the week: Friday night, Saturday and Sunday are more likely to yield higher demands than workdays.
  - week of the month: The bi-weekly payroll days in United States are typically the first and 15/16th days of each month according to Forbes. Thus the first and third week can potentially have surging demand. However, this hypothesis need to be tested with the data.
  - public holidays: We can encode all public holidays in the data in 2023 and test its importance in the prediction models.

  Note that to consider the potential impact of multiple seasonalities, we need to perform transformation to decompose the taxi demands time-series to its trend, seasonality and residuals.

- spatial factors:

  The spatial patterns in the taxi trip records may be helpful for taxi demand prediction. Specifcally, a surging number of drop-offs in the same borough may indicate that there is an event, and therefore impact the future taxi demand hours later. For example, if there are lots of drop-offs in Bronx or a neighoring borough at 9pm, it could be the cause of a concert at Bronx and it will lead to taxi demand surge in Bronx hours after the event. Therefore we need to encode the spatial factors of the drop-off locations and examine the impact of such features in our prediction models.

- Crowd effects:

  Similar to the spatial information, the abnormalty of the number of passengers in the taxi may indicate fluctutation of taxi demands hours later. e.g. group-trip for concerts. Therefore, we need to encode this information for predictions.

- Weather:

  we will incoporate the weather data into the prediction models.

- prediction models:

- Baseline Model:

  We will be using weighted moving average as the baseline models for hourly demand prediction. We will be using mean absolute percentage error (MAPE) for the performance evaluation. The rationals and details will be explained in the methods section below.

- Improved Models:

  We will be consder model improvement by testing the following models below. Details will be in the method section.

  - adding smoothing and decomposition features from Fourier decomposition, STL decomposition, and Holt-Winters Method (i.e. triple exponential moving average).
  - adding spatial and other interesting features.
  - test machine learning approach: random forest and xgboost.

- Best Model:

  xgboost model achieved lowest MAPE value on test-set after parameter tuning.

- Forecasting the First Week of September 2024:

  See details in the forecasting section.

- Optimal Number of Taxis:

  See detailed analysis in the final section.

Methods¶

Data Quality Control¶

We first conduct data quality control before analysis. This includes checking:

- Data summary

  The original TaxiTrip data contains 66,000 rows or records. The columns include: tpep_pickup_datetime, tpep_dropoff_datetime, passenger_count, trip_distance, PULocationID, and DOLocationID.

- duplications

  662 duplicated rows are removed from the data. This is probably due to techinical errors.

- Missing values

  5752 passenger_count entries and 38 trip_distance entries are missing values. we replace the missing value with the median value of the column with same pick-up and drop-off locationsIDs.

- Zero values

  13,369 trip-distances are zero. 2,685 records are with exact the same pick-up and drop-off date-time. We remove these 2,685 rows. For the rest of zero entries, we replace the zero trip-distances values with the median value of the subset trips data with the same pick-up and drop-off locations. Note that the high number of zero values implies that the original recorded trip distances may not be a reliable measurement of the trip information.

  294 passenger_count have zero values. Assuming this ride-hailing company did not mix delivery service records with ride-hailing records, we will replace the 0 values with the median non-zero value.

- Outlier values

  We examine the summary statistics of each numerical column and find outliers in passenger_count and trip_distance. There are 6 outliers in passenger_count values. i.e.

we have 6 records of which each contains 11 passengers in a taxi. we replace these passenger_count with the median value of the column.

There are also extremely outlier values of trip_distances (See Figure 1). The New York City has an area of 300.5 square-miles, therefore the theoretical maximum distances of a single drop-off in the NYC area should be sqrt(2)*sqrt(300.5) = 24.51 miles. However, consider that (i) there could be multiple drop-offs of multiple passengers in a single trip, and (ii) there were drop-offs outside of NYC, then the actual trip distances could be exceeding that limits. Therefore, instead we calculate the trip durations of each trip record, and calculate the expected trip_distances with the general speed limit of NYC, i.e. 25 mph. Specifically, for each suspicious outlier trip_distances (> 24.51 mile), we replace the recorded trip distance value with the lesser of these two values: (i) 25 mph times the recorded trip_durations for this trip and (ii) the original record trip_distance value.

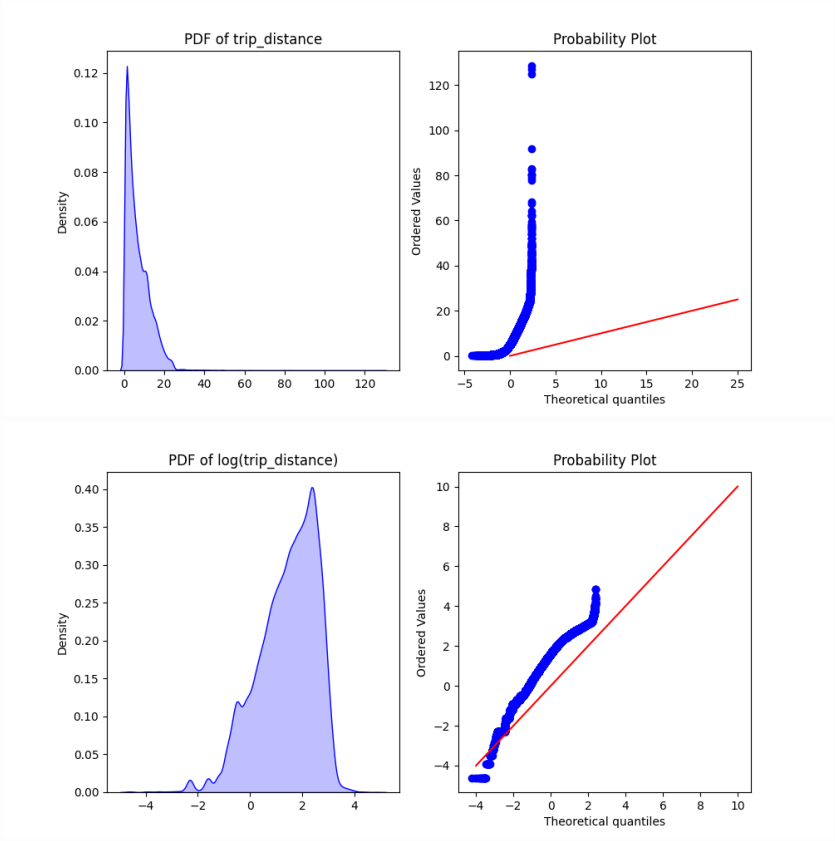
- trip durations

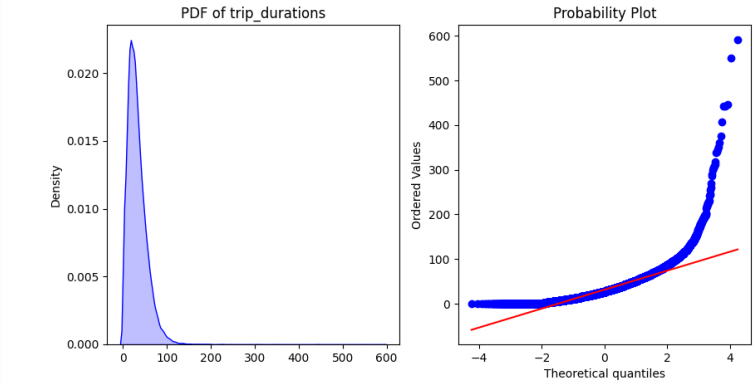
  Note we calculate the trip_durations from the pick-up date-time and drop-off date-time. There were 13 rows with pickup time later than drop-off time. Without proper hypothesis we will remove these 13 rows. There were also extremely long trip_durations (e.g. 3556 minutes = 59 hours). According to NYC government, "Both taxi and FHV drivers are prohibited from transporting passengers for more than 10 hours in any 24-hour period". Therefore we will remove all records that with trip_duration exceeding that limits.

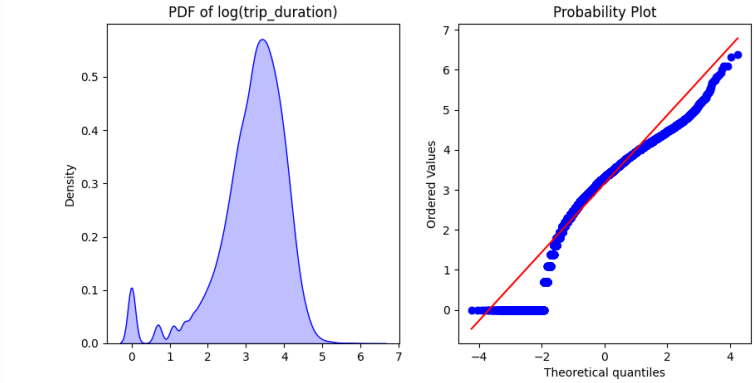
- Summary Statistics

  - At least 75% of trips have only one passenger.
  - The median trip distance is 4.9 miles. 75% of trips have trip distance < 10.1 miles.
  - The median trip duration is 27 minutes. 75% of trips have trip duration <= 43 minutes. Note that we will use this information later for calculating the optimal number of taxi deployment.

In [ ]:







Data Curation¶

We now curate hourly taxi demand data with the following steps:

- Attach the exact location information

  We first attach the exact pickup and drop-off location information from the taxi_zone_lookup to trip_dat. This information will be encoded as features for prediction models.

- Insert missing hours

  Note that some mid-night hours may have no taxi demand. Therefore we will insert rows with 0 taxi demand for these hours. This is necessary for data smoothing later.

- Data smoothing

  The zero taxi demand may lead to singularity problems in model fitting and performance evaluation. Thus we need to perform data smoothing on the zero taxi demand hours. Specifically, for each hour where taxi demand = 0, we take the previous and next hour data, assign the same values, i.e. the round value of average of the three rows, to these three rows for each column of num_rows,sum_passenger_count, Manhattan, Queens Brooklyn Outside of NYC Staten Island Unknown EWR. For sum_trip_distance and sum_trip_durations we only assign the average vaues of the three rows. For avg_trip_durations, assign the max value of these three rows to all three rows. Note that after this imputation we still have 132 rows with num_rows = 0, which indicates that there exist windows of three hours with taxi demand < 3. For these remaining 132 missing data we replace the zero value with the first non-zero value prior to that hour.

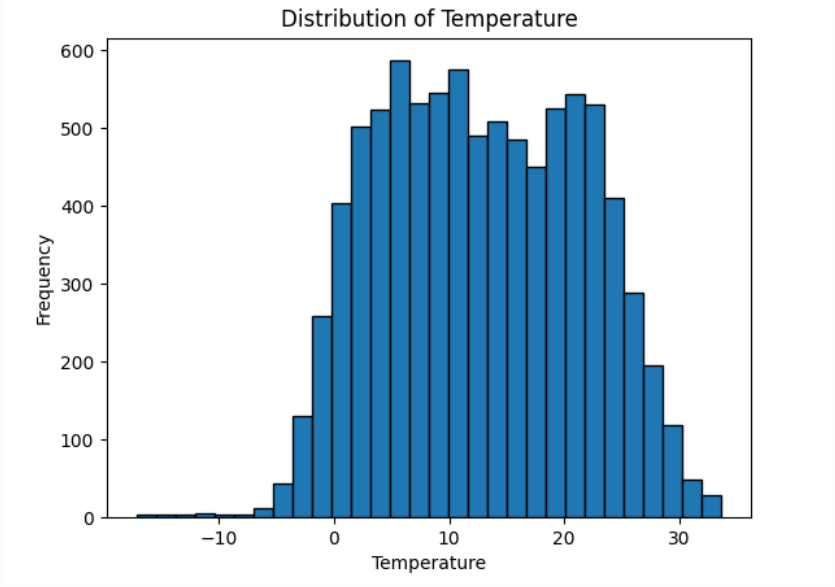
- Feature Curation

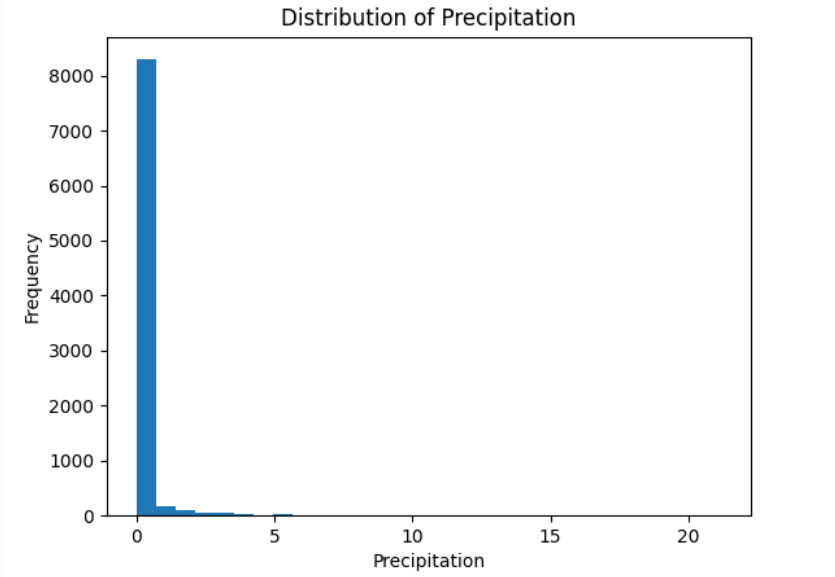
  We add extra features that may be of interest to the data including:

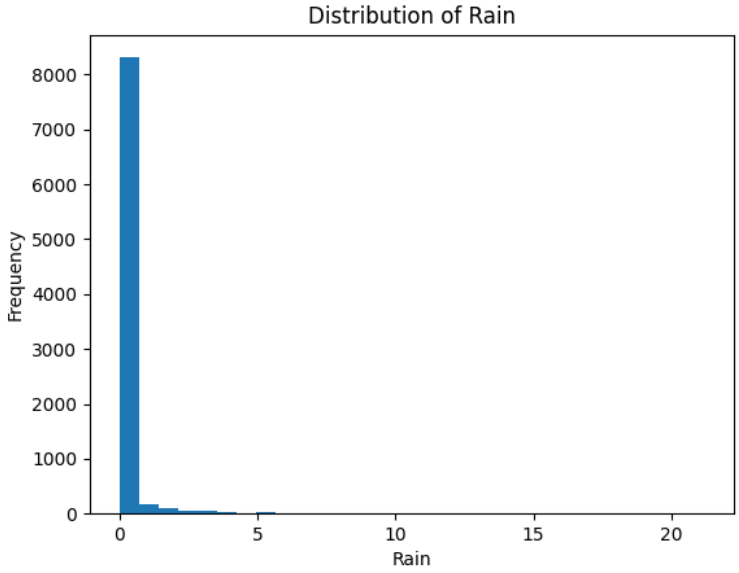
  - day of the week
  - week of the month
  - holiday index: we mark the public holidays in NYC in the data with a binary variable

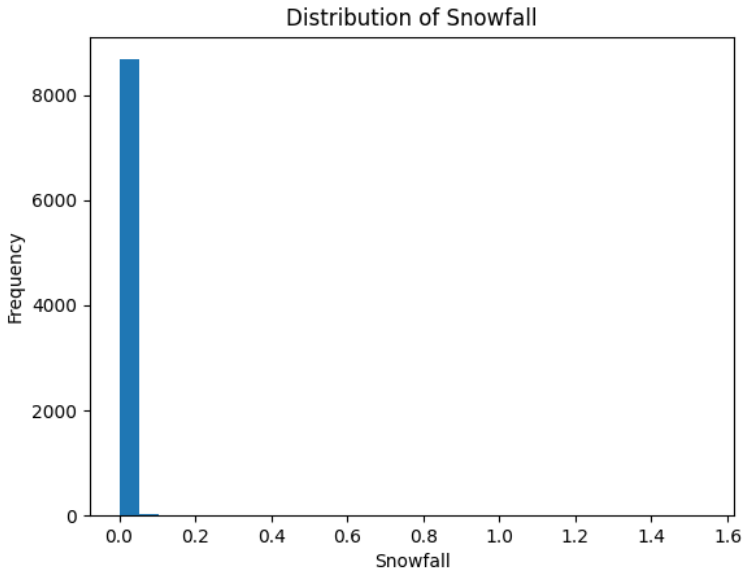
(1 or 0)

- weather data from Bronx_Weather_Data2023.csv. We examined the summary statistics of all 4 features and find no apparent outliers.









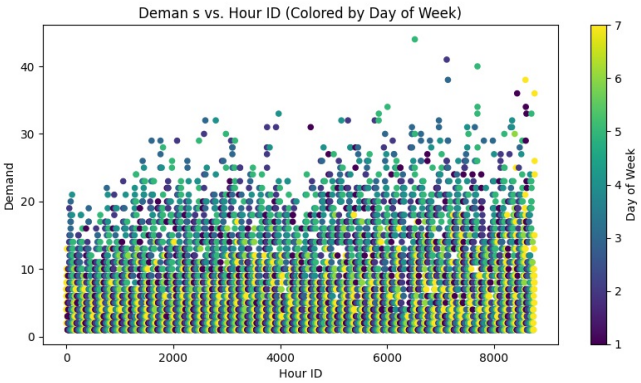
Data Modeling¶

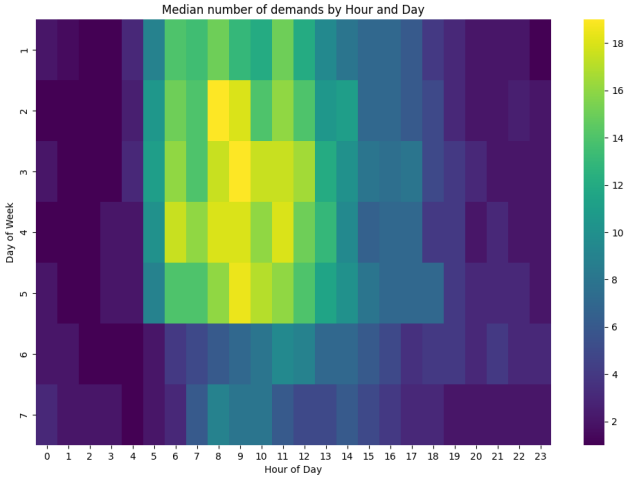
We model the number of hourly taxi demands in the Bronx area and determine the optimal number of taxis required to meet this demand.

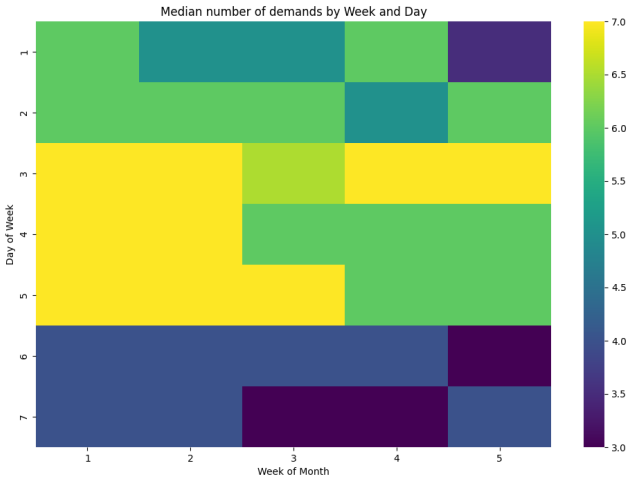
The time-series of the hourly taxi demand is shown in the Figure below. We also show the seasonality of the demand by hour of the day, day of the week, week of the month. The heatmap of hour-of-day and day-of-week shows a clear pattern of higher demand from 8am to 11am on Tuesday to Friday, with a minor outlier at 9pm on Saturday night. This could indicate the demand in Bronx are largely driven by the the work commute. The heatmap of day-of-week and week-of-month shows a clear parttern of higher demand at the first 2 weeks, which could be due to the payroll schedules and its first month effect on consumer behavior (Justine, 2010).

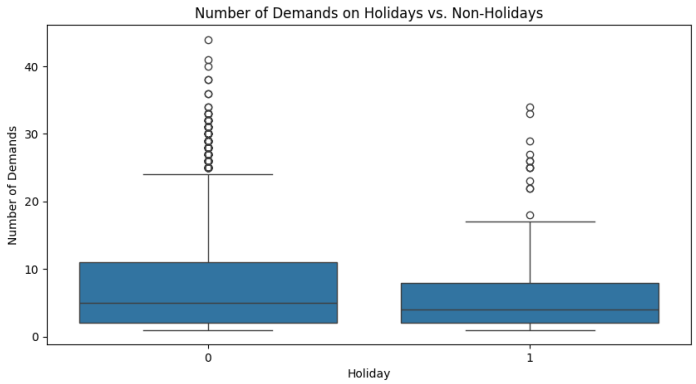
We also show the association between demand with holidays, temperature_2m, precipitation,rain and snowfall. The boxplot shows the interesting fact that the holidays have significantly lower demands, which supports the hypothesis that the Bronx taxi demand are largely driven by the work commute. We also observed a signficant negative correlations between temperature and demand, though the relationship can also be seen as nonlinear. A

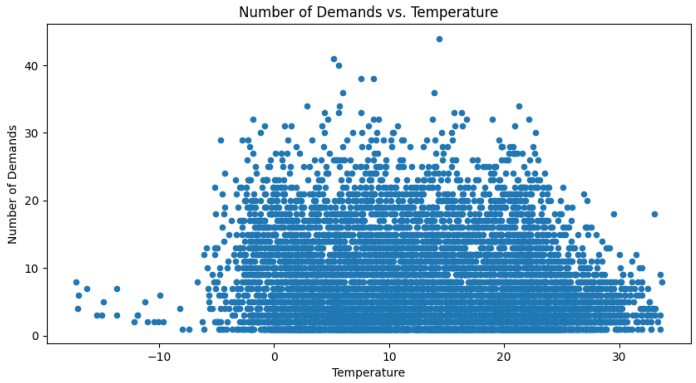
significant positive correlation between precipitation and demand are also observed. Note that the rain volumn and precipitation are highly corelated (p>0.99). Therefore we dropped the rain volumn as a feature. Finally, the snowfall is not significantly correlated with taxi demand even restricted to snow days (pval=0.33). Therefore we also remove the snowfall features from the data.

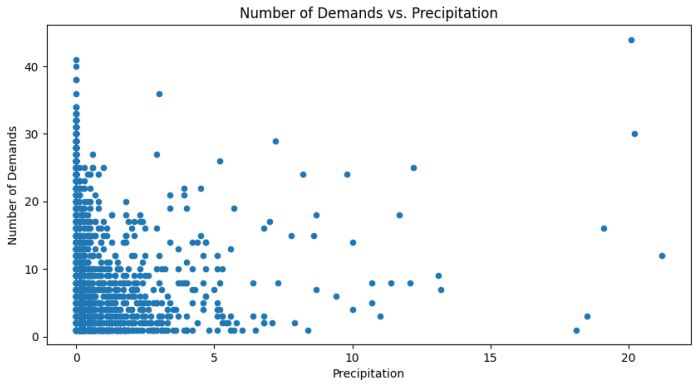


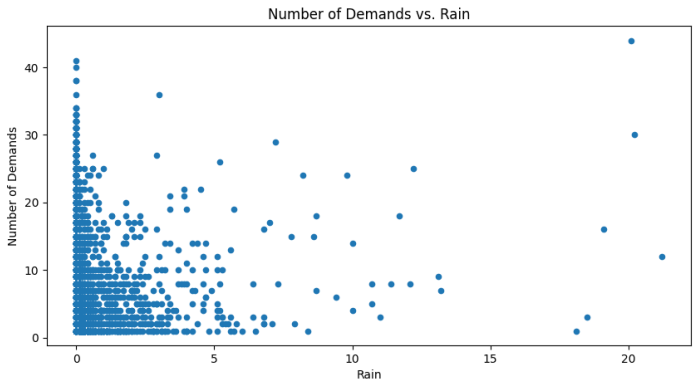


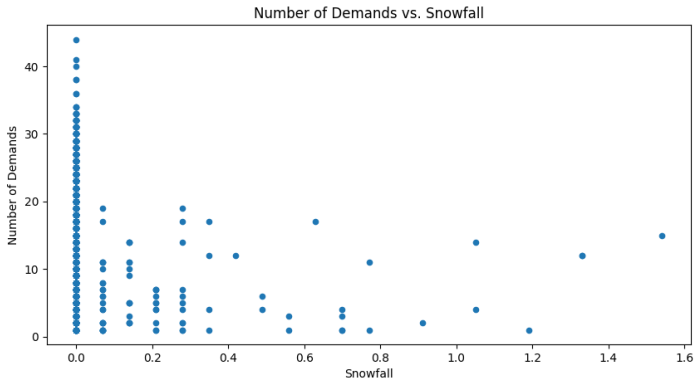



Number of Demands on Holidays vs. Non-Holidays
Number of Demands
0
10
20
30
40
0
1
Holiday


Number of Demands vs. Temperature
Number of Demands
0
10
20
30
40
−10
0
10
20
30
Temperature


Number of Demands vs. Precipitation
Number of Demands
0
10
20
30
40
0
5
10
15
20
Precipitation


Number of Demands vs. Rain
Number of Demands
0
10
20
30
40
0
5
10
15
20
Rain



Baseline Model¶

The most intuitive modeling of the time-series demands is to use the previous houly demands to predict the future demand. Considering the strong cyclical nature of houly taxi demand, we adop the weighted moving average model as the baseline model for this task. Note that we prefer weighted moving average over simple moving average because of the assumption that taxi demand at a specific time t is more likely to be close to the demand at t-1 than that from a more distant timestamp.

We choose Mean Absolute Percentage Error (MAPE) as the performance evaluation metric for this analysis. Among all performance evaluation metrics, including mean square errors (MSE), mean absolute errors (MAE) and root mean squared errors (RMSE), we prefer MAPE for the following reasons:

- (i) it is more sensitive to larger percentage errors, therefore ideal to evaluate model performance during low-demand periods in order to reduce operational cost of idle vehicles;
- (ii) the Bronx data here has no apparent outlier demand values, i.e. the high demand period data are reliable. Therefore the sensitivity of MAPE to larger percentage error is a desirable property to favor models with accuate predictions of the high demand and thus high revenue periods;
- (iii) it is widely accepted in the literature of traffic predictions for the reasons listed above.

We decided the best baseline model is WMA with window size equals to 2, after manually tunning of the chocie of window size to achieve best MAPE values on the complete data. The final WMA model achieved the MAPE with 0.18 as below. We will use this result as baseline to improve our prediction models.

Machine Learning Models¶

There has been extensive publications that adopt machine learning algorithms to pursue taxi demand predictions. The advantage of using machine learning algorithm over moving average methods includes:

- It has the potential to incoporate features that reflect the cyclical and seasonal patterns.
    - Fourier transformation is a widely accepted method to model cyclical patterns and nested seasonalities, of which the frequency can be incorporated into machine learning algorithm as predictive features in comman practise.
    - Seasonal Decomposition of Time Series (STL) can also be used to capture long-term trends and seasonal cycles.
    - Holt-Winters' Seasonal method is also suitable for our analysis since it captures the stable seasonlaities with fixed periods (e.g. hour, day, week).
    - Other transformation methods such as Wavelet transform and box-cox transformation may be of interest but unlikely to have significant boost to our analysis due to their nature. Therefore we will not consider these transformations in this analysis.
- It allow more flexible weighting strategy for the utilization of the lagged features. In fact, WMA assigns the historical taxi demand at t-1, t-2..t-N a weight vector with fixed values

(e.g. N,N-1,..1) for predicting taxi demand at time-point t. In contrast, machine learning algorithms can incorporate historical taxi demand as input features and obtain data-driven estimations of the weight vector to understand time dependencies.

- It allows the incorporating of additional features such as weather, spatial factors (drop-off locations)...etc.
- There is a wide range of machine learning algorithms, with diverse underlying mechanisms, to compare and choose.

Improved models¶

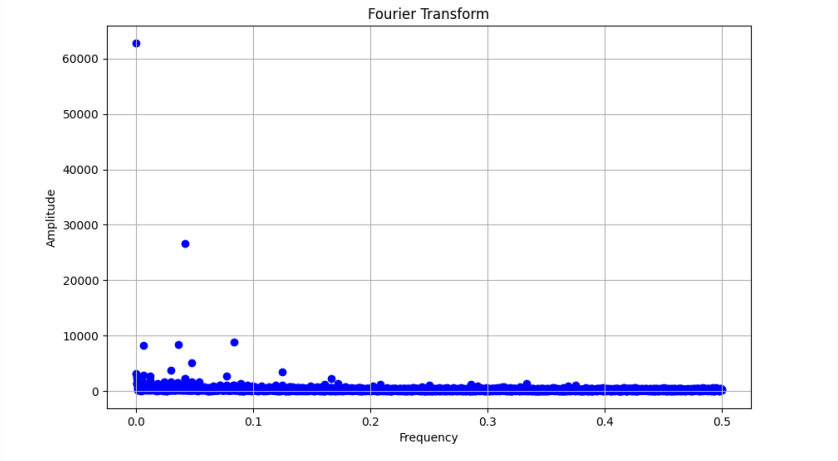
Before we make selections of machine learning algorithm, we conduct extra feature curation to address the oppertunities above:

data split:

we split the time-series data into training and test set. The training data contains data of January to September, the test data contains data from October to December. Note that we intentionally put the September 2023 into the training set to learn the patterns in September, such that the prediction of September 2024 would not miss important signals.

Fourier transformation:

we fist examine the fourier features on training data. The figure shows there were 5 peaks on amplitude vs frequency plot. The highest peak is from DC component. The rest peaks are according to frequency: 1/24, 1/(24*7), 1/12, 1/28. Thus we incorporate the 11 fourier features from the top 6 signals, while reomving the DC component. The amplitude and frequency are extracted for each month (Jan, Feb....etc) seperately.



STL decomposition:

We extract features from STL decomposition to capture the trend and seasonality. Three seasonal period is considered: semi-daily (12), daily (24), weekly (24*7). We applied STL to each month seperately and combined the seasonal, trend and residuals features together. In total we created 9 STL features (i.e. seasonal, trend and residuals for each of the three periods).

Holt-Winters' (HW) seasonal method

Also know as Triple Exponential Smoothing, the Holt-Winters' seasonal method is popular for capturing the level, trend and seasonal components. The major benifits of HW method is it is able to capture and adjsuts for trend over-time, which Fourier transformation could not. It is also been proven that it works excellent for tiem-series with regular seasonalities (e.g. day and week). Thus we also extract the HW trend, level, and seasonal features for each month by re-using the code from Gregory Trubetskoy (https://grisha.org/blog/2016/02/17/triple-exponential-smoothing-forecasting-part-iii/). Note that HW method has three parameters alpha, beta and gamma, which controlls different levels of smoothing on level, trend and seasonality. We use cross-validation to tune the parameter values for each one of the two

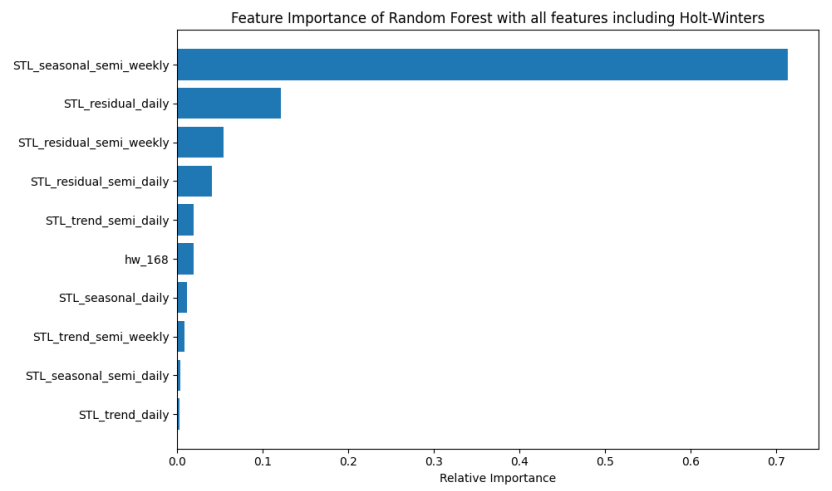
seasonlities: daily and weekly. Then we apply the HW transformation onto test data with the same parameter set values seperately for daily and weekly seasonlities. Therefore we will have 2 HW features (ie. predictions) extracted for training and test set seperately.

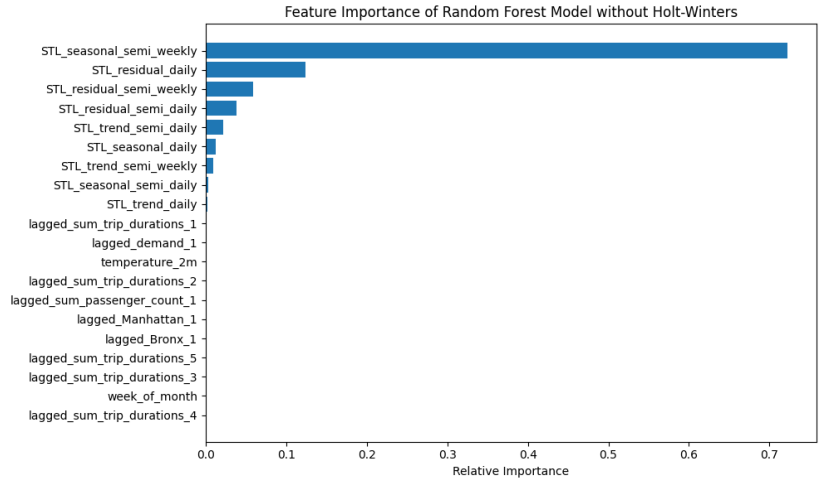
Final predictor list

- Lagged demand: We also include the lagged taxi demand for each time-point as predictors. i.e. for each hour t, the number of taxi demand at t-1, t-2, t-3, t-4, t-5 are included as predictors.
- Lagged passenger counts from the previous 1-5th hours.
- Lagged trip durations from the last 5 hours.
- Lagged Bronx drop-offs from the last 5 hours.
- lagged Manhattan drop-off from the last 5 hours.
- day_of_week.
- week_of_month.
- holiday
- temperature_2m
- precipitation
- Fourier features: amplitude 1, frequencies 2....
- STL features: stl_...
- Holt Winters features: hw_...

Improved Model I

We first start with a simple random forest method and find the RandomForest with STL features can achieve MAPE = 0.15. Interestingly, the random forest with HW features have slightly worse performance comapring to the RF model by dropping HW features. This implies that (i) HW Features and STL features may captured similar patterns; and (ii) W and STL feature groups may have multicollinearity, (iii) the RF could be overfitting on training data due to the hight number of features. In order to address these issues, we will explore model improvement.





Improved Model II

Xgboost is a better solution to address high-dimensional features and multicollinearity issues since (i) it has embedded feature selection process; and (ii) the tree based algorithms are not directly impacted by multicollinearity. Therefore, we implemented the xgboost algorithm on the training data, and optimize the model by tune the parameter values to achieve the best performance on the test-set. As a result, we received the optimal MAPE 0.067.

We also explored the feature importance of the xgboost model, and find the nonzero important features are:

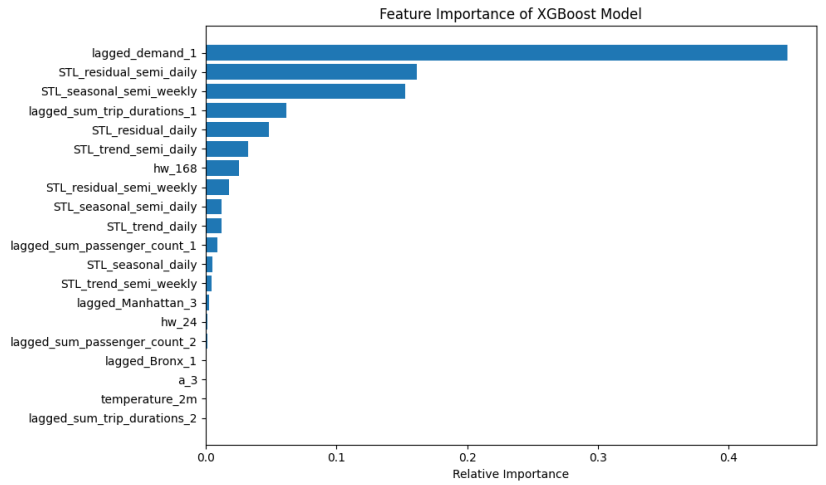
- the last hour demand
- STL lagged predictions
- the last hour total trip durations
- HW weekly seasonaly predictions
- the last hour total passengers
- the total drop-offs at Manhattan 3 hours before

The last two features are interesting since they may be viewed as evidence for our hypothesis: that the Bronx taxi demand is subject to impact of gatherings and events in NYC, especially at Manhattan.

In [ ]:

```
import pandas as pd
model_comparison_tab =
pd.read_csv('/Users/laijiang/Documents/Pers/datatest/Data/save/model_com
)
print(model_comparison_tab)
```

```
      Model      MAPE                                       Notes
0       WMA  0.181333  Weighted Moving Average with window size 2
1     RF_HW  0.193581            Random Forest with all features
2  RF_no_HW  0.150511         Random Forest without Holt-Winters
3   XGBoost  0.066465                  XGBoost with all features
```



Best Model¶

The comparison table is shown above. The results shows the best model is xgboost, which achived MAPE = 0.066 on test-set comparing to WMA and Random Forest.

Forecasting the First Week of September 2024¶

We use our best model XGboost to make predictions on the first week of 2024. In order to make prediction on September 2024, we need to generate hourly feature values for these XGboost predictions with non-zero importance. However, we do not have 2024 Bronx taxi demand data to deduct the required predictors. Even though we can manually curate some features such as temperature and precipitation, day-of-the-week ,...etc, however, these features are of zero importance for XGboost predictions. In fact, the xgboost model are major driven by the seasonal and trend features of demand.

Therefore, we will use the predicted taxi demand of frist week of September 2023 to forecast the First Week of September 2024. The assumption is that the factors of multiple seasonality and trend has major impact of taxi demands, and this assumption is supported by our best model.

Optimal Number of Taxis¶

In order to determine the optimal number of taxis required to meet the predicted hourly demand in the Bronx, I will make the following assumptions:

- the houly average trip durations in the first week, September 2023 is a reliable indicator of that in first week, September 2024.
- The taxi are constantly avaiable during the hour between trips. i.e. the Taxi drivers are constantly on duty when deployed. The pick-up time for taxi driver to get back to Bronx for the next request can be ignored.
- Oversupply: If the taxi supply is over the true demand, then the idle driver will lead to operational cost. The minimum wage at NYC is $15 per hour, while a recent NYC law have required companies to pay delivery workers $17.96 an hour. In our calculation, we will use a=$17.96 hourly pay as the baseline idle cost.i.e. Every hour every idle taxi driver will generate extra $17.96 operational cost.
- undersupply: If the taxi supply is under the true demand, then the cost is (i) revenue oppertunity cost, which we will assume to be $19.62 per-trip according to a NYC article (https://www.nytimes.com/2022/11/17/nyregion/taxi-fare-hike-nyc.html); plus (ii) customer wait times will increase, leading to dissatisfaction. it will bring future potential revenue lost if the customers seek service from a competitor or public transportaion. Given this, I would make assumption that a high percentage of customers, specifically 80% would drop-off the demand pool. i.e. if the surplus of taxi demands are not met in this hour, then only 0.2 of the surplus demands last hour will be added to the demand pool of the next hour.

We conduct simulation analysis by simulating different values of taxi supplies: S. i.e. the number of Taxis deployed. For each value of S ranging from 10 to 100, we will calculate the expected total revenue for the first week, September, 2024. Specifically, the calculation goes as follows:

- For the first hour, let D=the number of demands, TD = the avearge trip durations (minutes) of this hour, S = supply of taxi.
- If S * 60 >= D * TD, then the revenue of this hour is D * 19.62 - (S * 60 - D * TD) * (17.96/60), i.e. income - idle driver cost.
- If S * 60 < D * TD, then the revenue of this hour is 19.62 * (S * 60 / TD). Update the next hour demand by D = D + 0.2 *(D - S * 60 /TD). Note that we recognize this approach may result in fractional demand for the next hour. However, we expect that, by the law of large numbers, the approximate distribution will converge to the true demand distribution over time.

Conclusion:

The optimal number of taxi supply is 19 for first week of September, 2024. Correspondingly, the expected total revenue is $25057.32 before expenses and deductions. The figure below shows the relationship between the total revenue and supply of taxi.