FA19-BL-CSCI-P556

APPLIED MACHINE LEARNING
NEW YORK GREEN TAXI DATASET

MADHURA BARTAKKE (mabartak)
December 2, 2019

Abstract
The Taxi Limousine Commission(TLC) provided yellow taxi service for the city of New York. But these taxi services were concentrated to the Manhattan region of the city and did not ride to the outskirts. The green taxi service was hence initiated to cater to the demands outside of Manhattan which started in 2013. The data is vast and there are various conclusions we can draw from it. The aim is to maximize the revenue of the company by implementing various machine learning techniques of supervised and unsupervised algorithms to identify the ideal model to the data.

About Data
The dataset is about TLC green taxi service for years 2016 to 2018. The data is being consolidated from NYC Taxi Limousine Commission for each of the above years. The data set consists of 19 columns, which consist of numerical and categorical values. There are total of 22.5 million instances overall. The domain of the dataset is Public Transportation. The description of each column of the data is given as follows:

Header	Data Description
VendorID	Indicates the LPEP vendor that provided the record **1: Creative Mobile Technologies, LLC** **2: VeriFone Inc.**
lpep_pickup_datetime	The date and time when the meter was first engaged
lpep_dropoff_datetime	The date and time when the meter was disengaged.
passenger_count	The number of passengers riding. This is a driver-entered value
trip_distance	The distance of the trip stored by the taxi meter
PULocationID	The taxi zone where the taxi meter was engaged
DOLocationID	The taxi zone where the taxi meter was disengaged
RatecodeID	The rate code that is in effect according to the type of ride: **1: Standard rate** **2: John F. Kennedy Airport** **3: Newark Airport** **4: Nassau or Westchester** **5: Negotiated Rate** **6: Group Ride**
store_and_fwd_flag	This flag indicated whether the trip record was held in vehicle memory before sending to the vendor

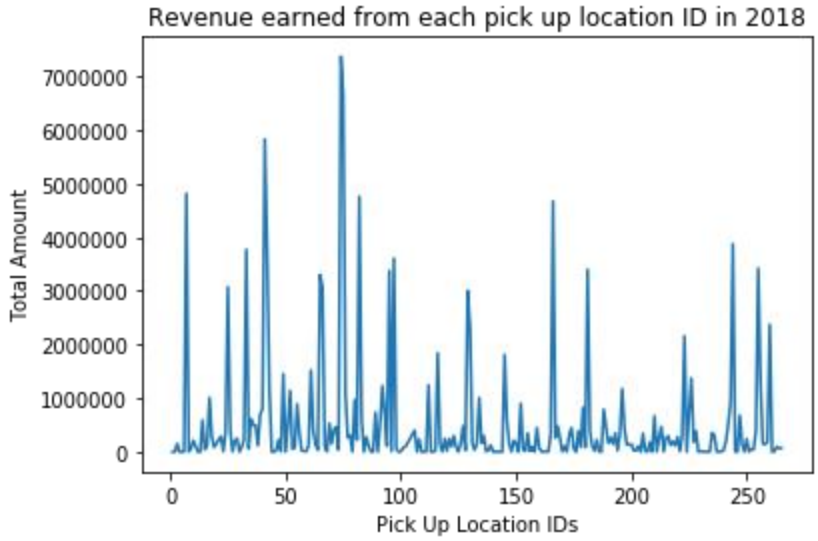
[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

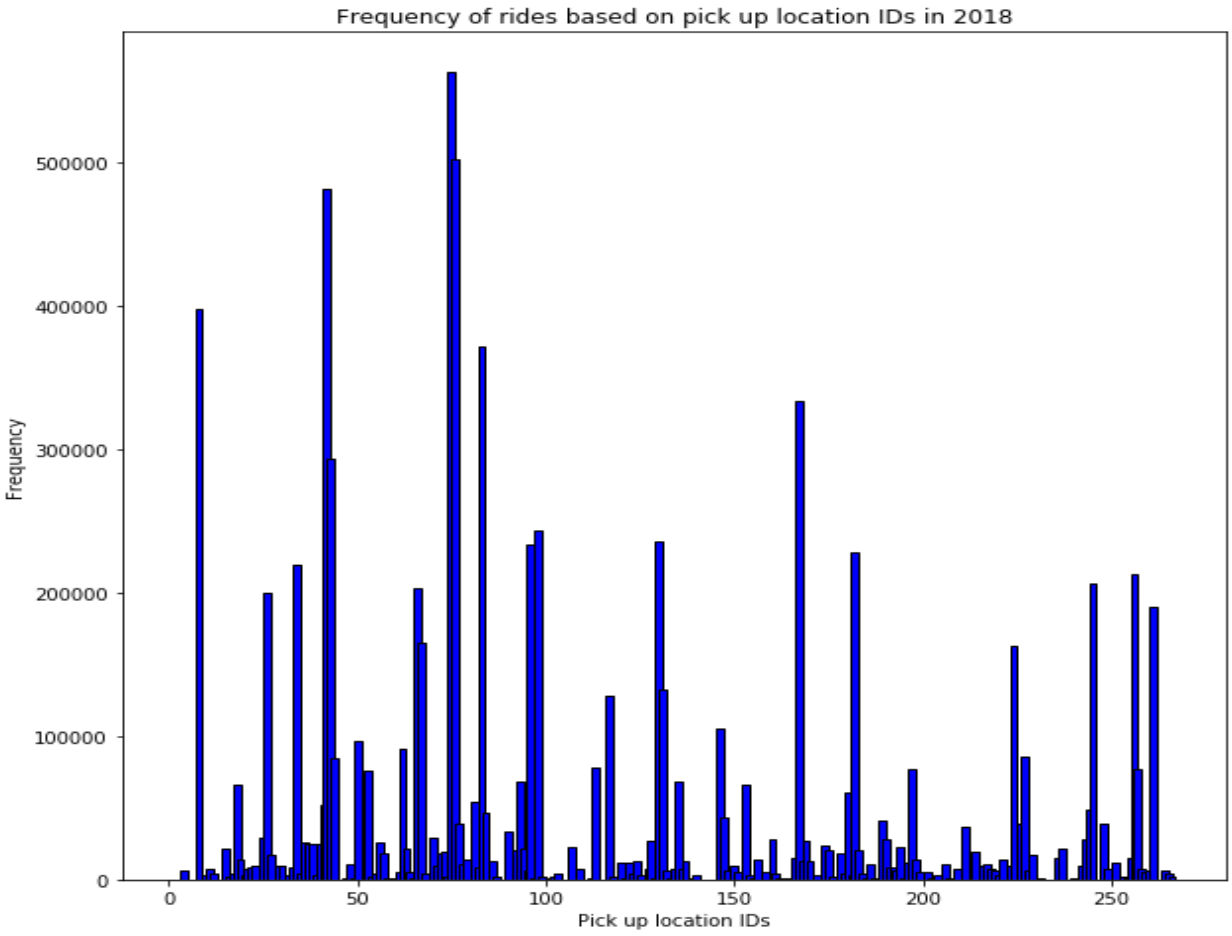
	aka “store and forward” because the vehicle did not have a connection to the server. **Y: store and forward trip** **N: not a store and forward trip**
payment_type	An integer indicating the type of payment method used by the customer. **1: Credit Card** **2: Cash** **3: No charge** **4: Dispute** **5: unknown** **6: Voided trip**
fare_amount	The fare of the ride calculated by the taxi meter on the basis of distance travelled and time spent.
extra	Miscellaneous extras and surcharges like rush hour surcharge and overnight surcharge
mta_tax	A $0.50 tax automatically added based on metered use
improvement_surcharge	A $0.30 improvement surcharge assessed on hailed trips at the flag drop
tip_amount	The tip amount which is automatically added in a credit card transaction.
tolls_amount	Total amount paid to tolls on the trip.
total_amount	Total amount charged to customers, not inclusive of cash tips.
trip_type	An integer indicating the kind of trip automatically assigned to the ride based on the metered rate but that can alter by the driver. **1: Street-hail** **2: Dispatch**
congestion_surcharge	A $2.75 charge on Green Taxis that pass through Manhattan south of 96th street

EDA:

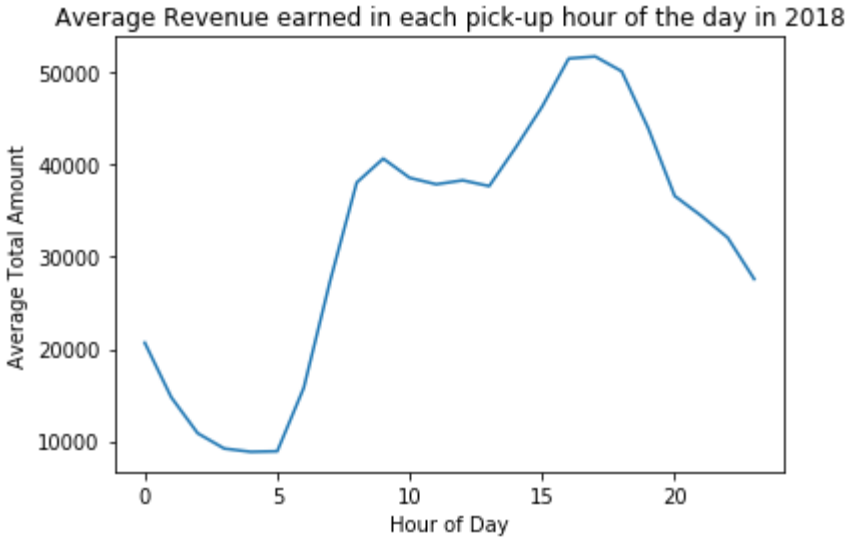
Given EDA is for the year 2018. Similar EDA was done for the year 2017 and 2019.

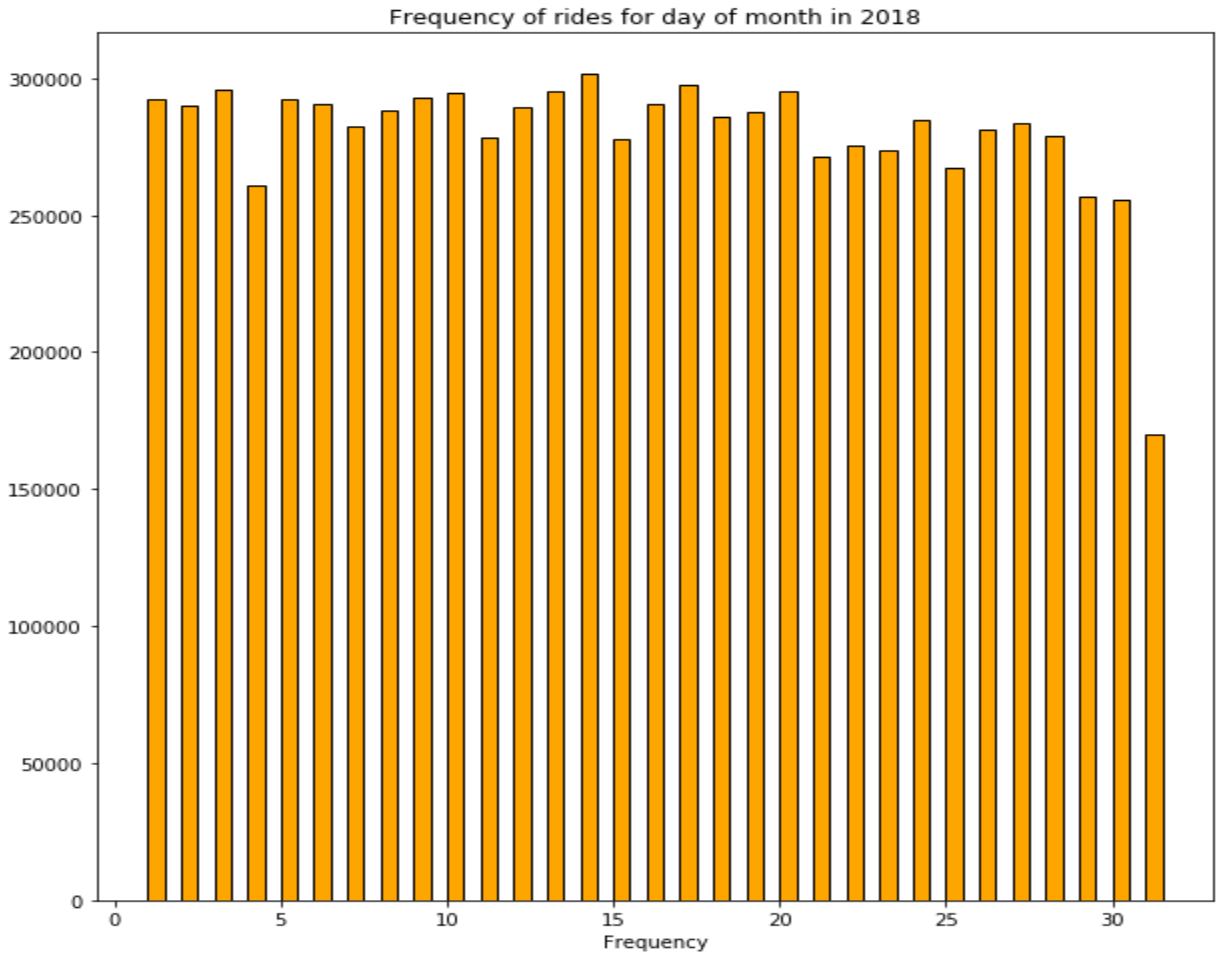
[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



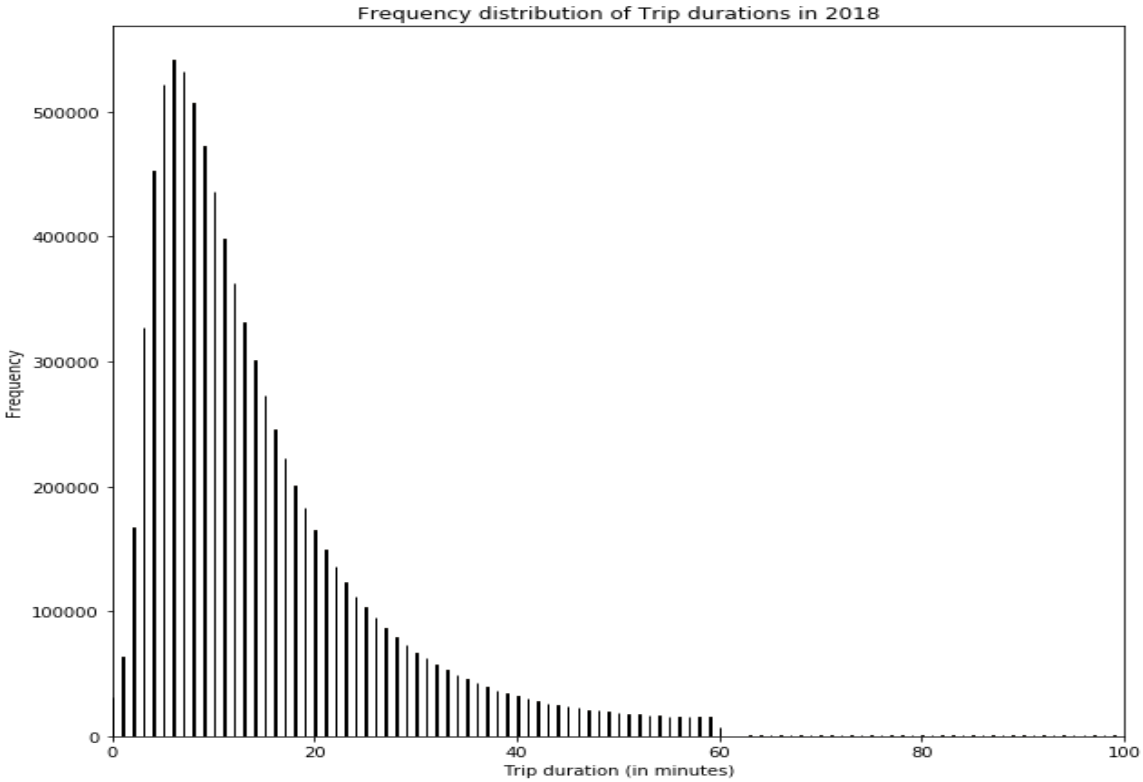


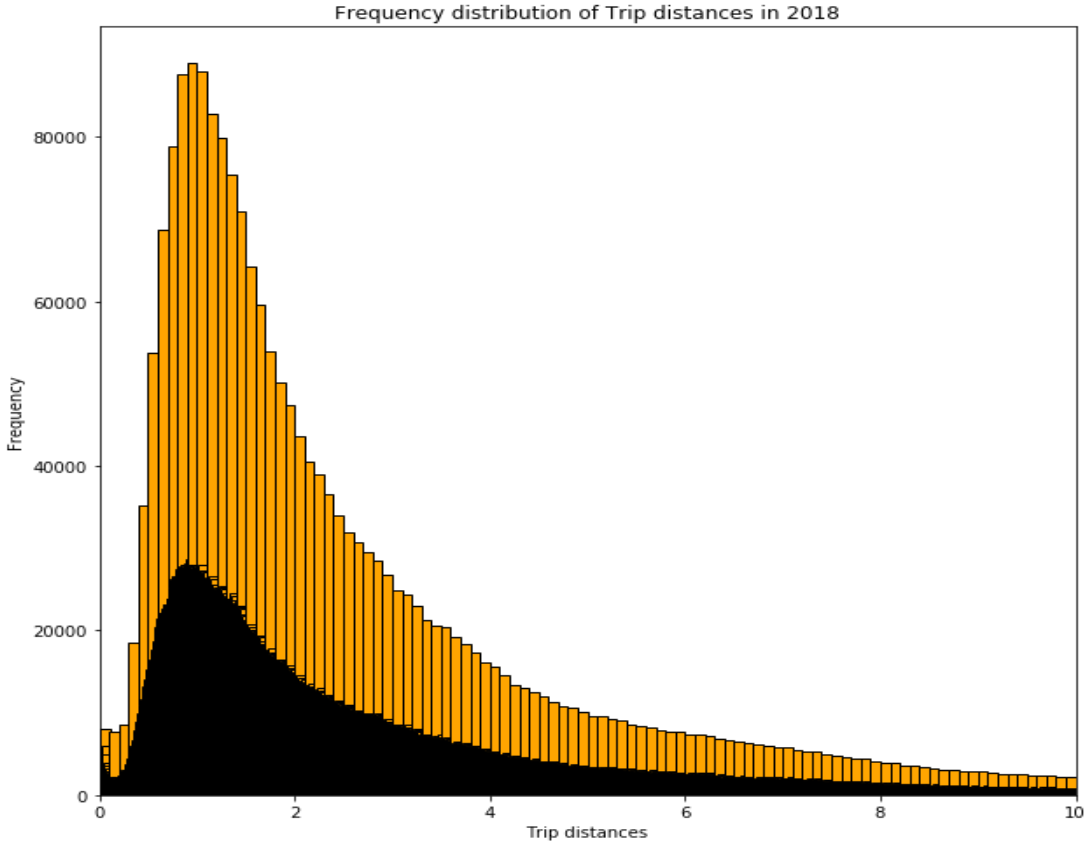
[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



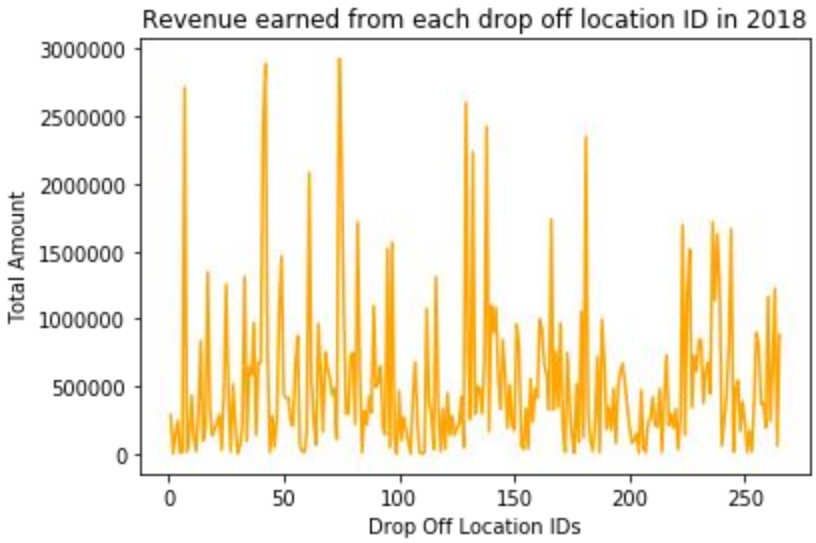


[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html





[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



Research Questions

1. **Which rides are the most profitable for the drivers on the basis of location, type of ride and time?**

Data Preprocessing:

1. Raw data is a monthly data which I consolidated by year.
2. Next, I combine all 3 years (June 2016 – June 2019) dataset into a single dataset which results into 22.5 million instances.
3. Valid Data:
    1. 'tolls_amount' – Since toll amount cannot be negative, I am filtering positive values for toll amounts.
    2. 'fare_amount' – Since the initial base fare charge is $2.5, I have taken fare amount values greater than or equal to $2.5.
    3. 'passenger_count' – Since an XL ride can take up to 6 passengers, I have limited the maximum passenger count to 6.
    4. 'RatecodeID' – Valid Rate code ID range from 1-6. Hence, a constraint has been put to remove any invalid category values
    5. 'trip_distance' – Rides whose trip distance lesser than or equal to zero are considered as cancelled rides. Hence, trip distances which are greater than zero are considered.

4. Date and Time format: 'lpep_pickup_datetime' and 'lpep_dropoff_datetime' store pickup and drop off date and time of every unique ride. Each of these columns were separated according to year, month, day, hour and minutes.
5. Dropped columns:
    1. 'ehail_fee' – This column consists entirely of zeros values which can be ignored.
    2. 'lpep_pickup_datetime' – Since date and time of pickup of customer has been incorporated by the separate columns created for year, month, date and time, this column for removed.
    3. 'lpep_dropoff_datetime' -- Since date and time of drop off of the customer has been incorporated by the separate columns created for year, month, date and time, this column for removed.

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

4. ‘store_and_fwd_flag’ -- The store and forward flag indicates if the record was initially held in vehicle memory due to some connection issue. This column does not add any value to our research and hence it is dropped.

Normalization:

The continuous variables were normalized using the formula:

$$Xnorm = \frac{X - Xmin}{Xmax - Xmin}$$

Model used:

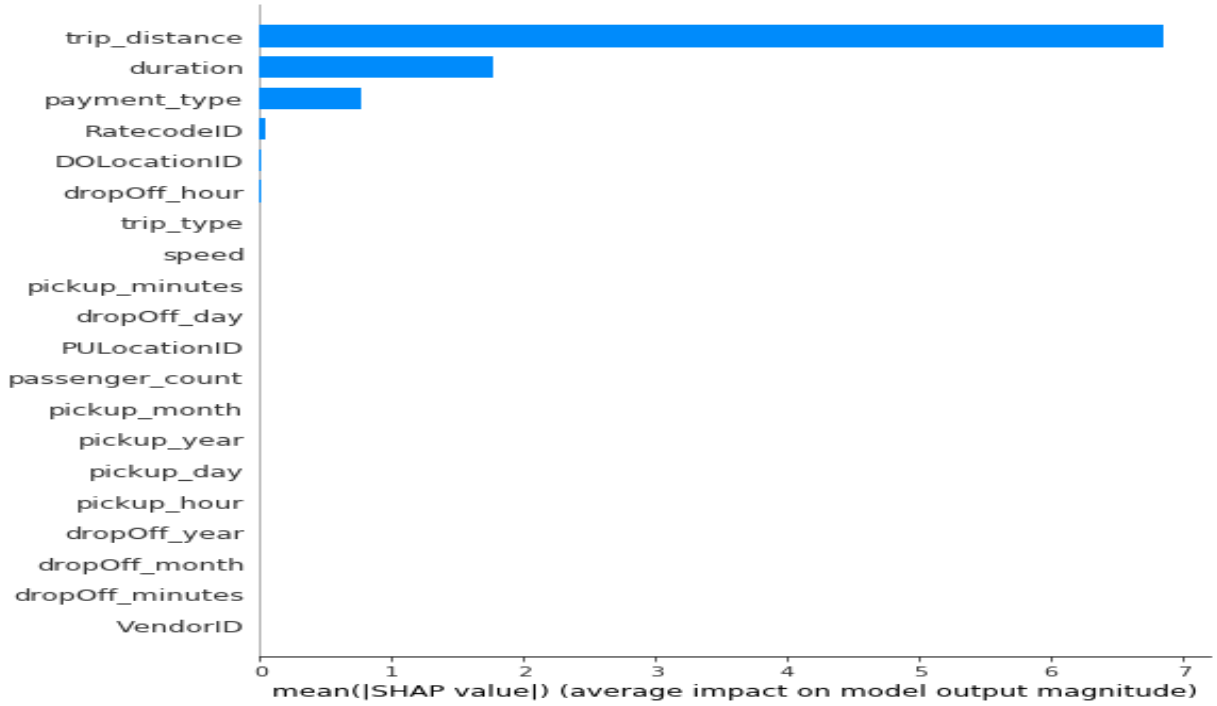
The model chosen to fit the model is Random Forest Tree Regressor. The aim of this research question is to explain the prediction of the total amount (profit) by analysing how each feature affects the output instance. We judge the feature importance using the SHapley Additive exPlanations (SHAP) that uses game theory to interpret the model chosen. SHAP has two estimation approaches KernalSHAP and TreeSHAP. TreeSHAP is the estimation approach used to predict the Shapley values here as Tree based models as that would help us correctly estimate SHAP values when features are dependent. Also it is computationally less expensive when compared against KernalSHAP . The features can be interpreted on a global as well as a local level.

The model chosen to fit the model is Random Forest Tree Regressor, where our input features are 'VendorID', 'RatecodeID', 'PULocationID', 'DOLocationID', 'passenger_count', 'trip_distance',’duration’ , 'payment_type', 'trip_type', 'pickup_year', 'pickup_month', 'pickup_day', 'pickup_hour','pickup_minutes', 'dropOff_year','dropOff_month','dropOff_day','dropOff_hour','dropOff_minutes',’speed’.‘total_amount’ is the output variable as is constituted of approximately the sum of the input features given below. Hence these features, 'fare_amount', 'extra', 'mta_tax', 'tip_amount', 'tolls_amount', and 'improvement_surcharge', have been removed to build the model.

Conclusions:

The model gives 89.55% accuracy when trained on data from 2018 and tested on 2019 data hence this model was chosen.

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



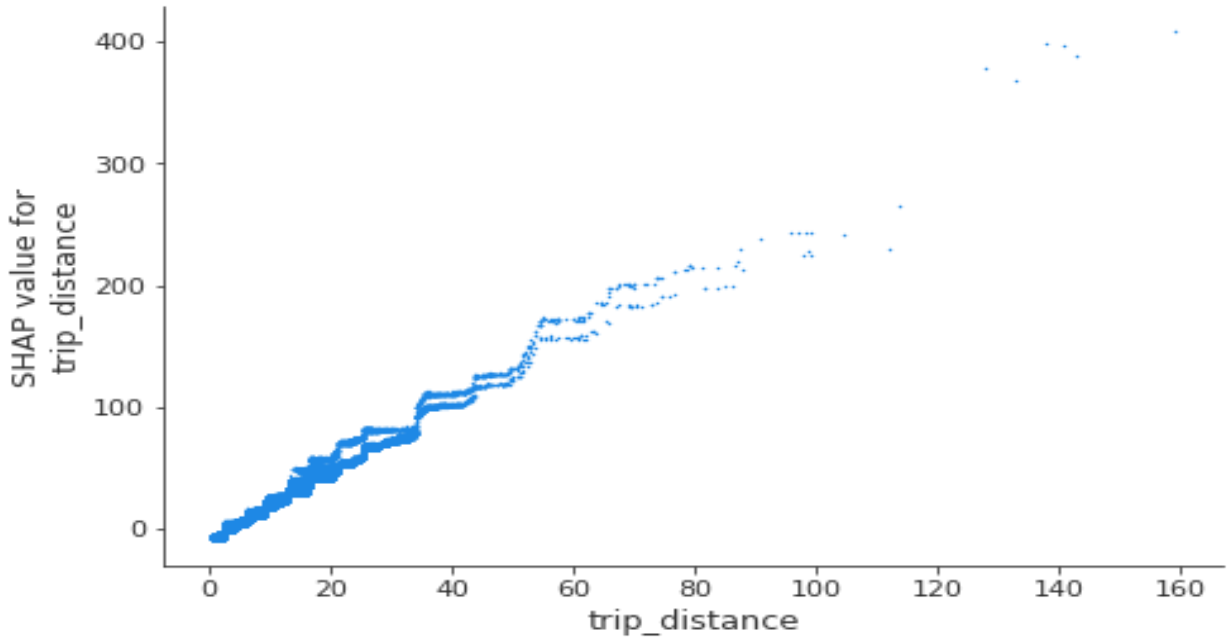
1.1.Shap feature importance

The features are given importance based on their individual effect on the output variable, 'total_amount'. This summary plot depicts the global importance of every feature on the output. The y-axis has the features and the x-axis has the mean absolute Shapley values per feature.

$$I_j = \sum_{j=1}^{j=n} | \theta_j^i |$$

Where n = the number of features and θ = shapley values

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



1.2.Shap dependence plot

A dependency plot is also displayed which shows the effect of a single feature on the predictions of the model. Since our summary plot indicates that 'trip_distance' effects the total amount the greatest, a dependency plot of 'trip_distance' is created.

2. Predict the trend in the revenue during holiday season like Christmas

Data Preprocessing:

1.Pre-processing: For this question, I had to do a different pre-processing of data since I wanted the date in year-month-date format. For this question our focus is mainly on two columns of the dataset namely 'total_amount' and 'pickup_date'.

2.For date to be categorized as a holiday or not, I first imported the US Federal calendar and weekend calendar. We then marked these holidays and weekends as Boolean value 1 and the rest as 0.

3.We grouped the data yearly. The training data consists of the year 2017, 2018 and 6 months for 2019 while I have predict the forecast values for the year 2019.

4.The rest of data pre-processing will be the same as stated earlier.

Model used:

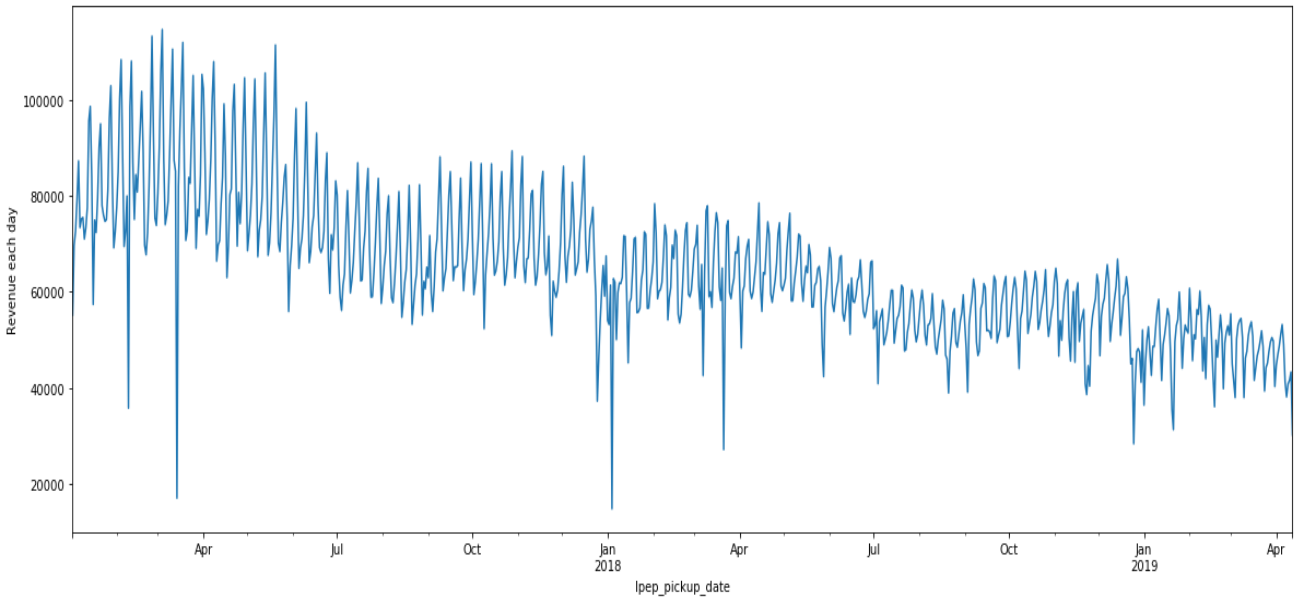
Here, I have used ARIMAX model which stands for Autoregressive Integrated Moving Average. ARIMA model basically explains the time series based on its own past values and lagged forecast errors. It creates an equation based upon the average of the past values to forecast the future or the predicted values. ARIMA models are denoted by ARIMA(p,d,q) which stands for seasonality, trend and noise respectively.

- P denoted the number of lagged values that have to be added or subtracted from the target which captures the "autoregressive" nature of ARIMA. This results in improved predictions on local growth or decline in our data.

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

- D is basically the degree to which our data is going up or down. So if d=0 that means out data does not go up or down, d=1 means that our data trends linearly, d=2 means our data trends exponentially. To summarize, d denotes the number of times that the data have to be difference to produce a signal (which has constant mean over time).
- Q captures the moving average part of ARIMA. It represents the number of prior or lagged values for the error term that are added or subtracted to Y
- The main aim is to select the best (ie. optimal) set of parameters that yields best performance for our model. In simpler terms we take the lowest AIC value.

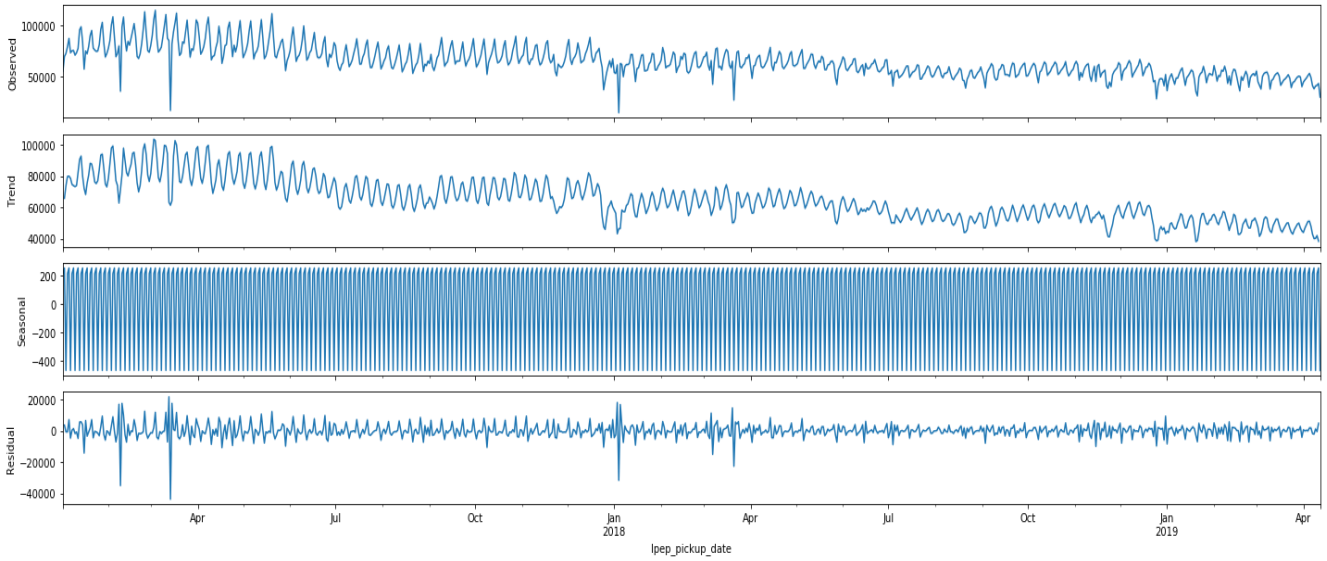
Conclusions:



2.1.Revenue for each day (Jan 2017-Apr 2019)

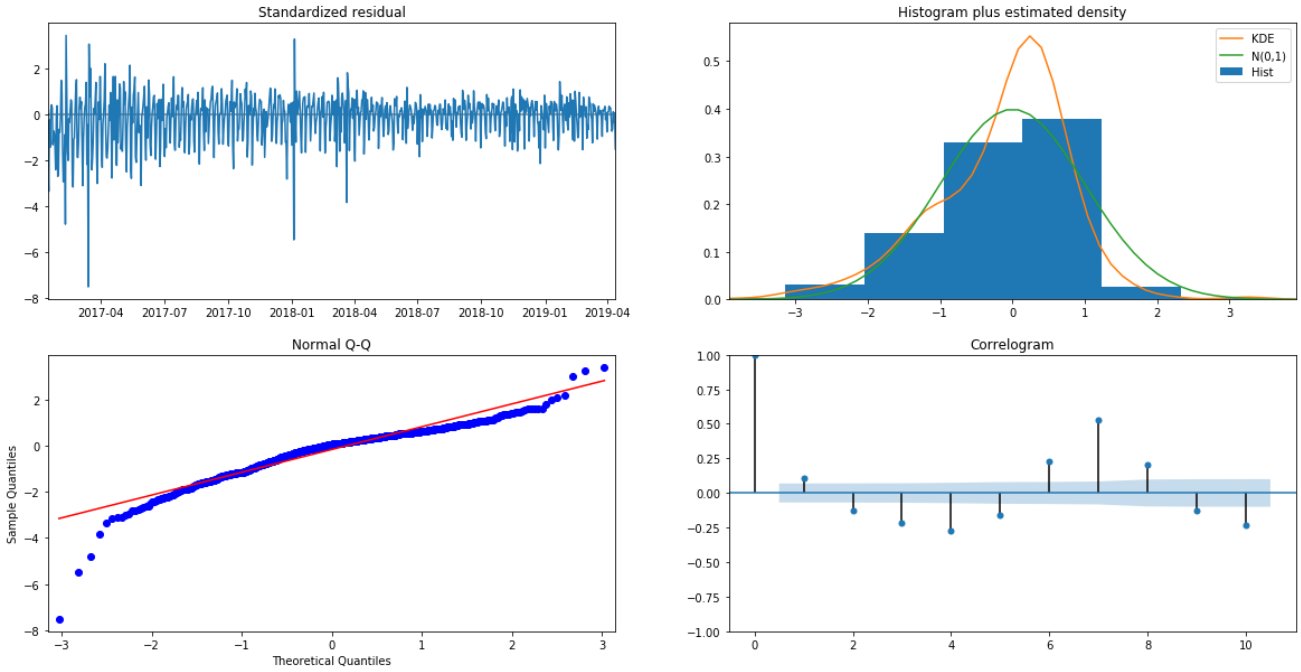
Some distinguishable patterns appear when we plot the data. The time series has seasonality pattern such as during holiday season such has Christmas and New Years there's a downward trend. One more such trend is seen in all the three years in the month of March end and beginning of April. We can see a distinct trend that the revenue for the year 2017 is more than for the year 2018 and 2019.

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html



2.2.Time Series Decomposition

We can also visualize our data using a method called time-series decomposition that allows us to decompose our time series into three distinct components: trend, seasonality, and noise. The plot above states that the total revenue for all the years 2017,2018 and 2019 are unstable.

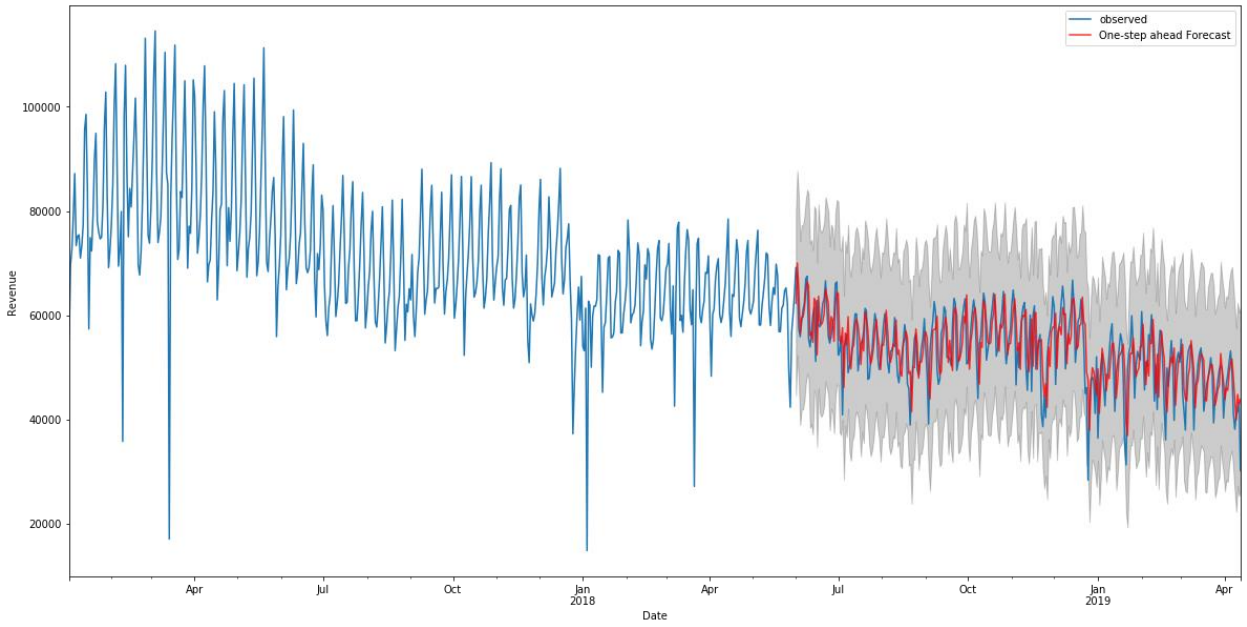


2.3. Revenue Diagostics

From the second plot (Histogram plus estimated density) we can conclude that the total amount seasonality is approximately normally distributed. The green curve signifies what the actual distribution of the data should be like. While the yellow curve signifies the true distribution which is close to the actual distribution which very few outliers.

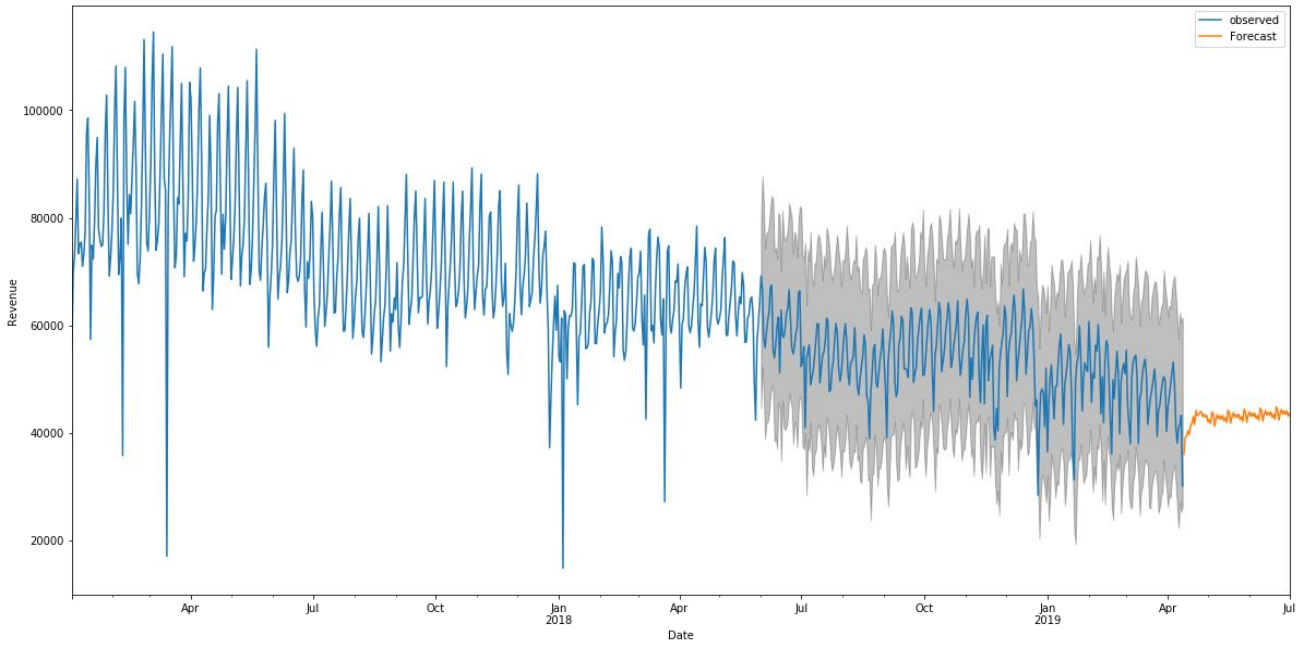
[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

The normal QQ plot is a linear line which clearly states that the data is normally distributed.



2.4.Observed values v/s forecast predictions

The line plot is showing the observed values compared to the rolling forecast predictions. Overall, our forecasts align with the true values very well. The blue line shows the observed values of the data. The one step ahead values are the predicted values which are denoted by red. The grey area denotes the confidence interval of the forecasted values



[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

2.5. Visualizing Forecast

Our model clearly captured total amount (revenue) seasonality. As I forecast further out into the future, it is natural for us to become less confident in our values. This is reflected by the confidence intervals generated by our model, which grow larger as we move further out into the future.

3. Analyzing distribution of rides during the day according to trip distance

Data Preprocessing:
Data preprocessing was done similar to Data processing in research question 1.

Model used:
I have used Quantile-Regression model for this research question.
Linear Regression depicts the relationship between the dependent and independent variables, providing a mean estimate for the independent variable as a depiction of the strength of the model. While this has been a classic approach to understanding the relationship between the predictor and response, a mean estimate does not depict what is truly depict the what occurs in different ranges of data. Given the vast number of data points, dividing and analysing the data in different ranges was imperative. This is what quantile regression allows—we estimate coefficients of our model to estimate and conditional median.
Through this research question we explored the relation between the trip distance and the trip duration, where the basic assumption is that a greater trip duration and lesser distances indicates greater traffic in the area. We see how shorter and longer trip durations are affected by trip distance and total amount (cab fare).

Conclusions:
For initial analysis, the 25^{th}, 50^{th} and 75^{th} quantiles regression models were created.
(The categorical variables were not one hot-encoded, instead considered label encoded as that they did not prove to contribute heavily to the model, and would add too many features considering the large number of groups with each categorical feature.)

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

QuantReg Regression Results

Dep. Variable:	duration	Pseudo R-squared:	0.6171
Model:	QuantReg	Bandwidth:	6.785e-05
Method:	Least Squares	Sparsity:	0.01144
Date:	Sun, 01 Dec 2019	No. Observations:	19079450
Time:	23:04:46	Df Residuals:	19079438
		Df Model:	11

	coef	std err	t	P>|t|	[95.0% Conf.	Int.]
Intercept	0.0025	2.91e-05	87.352	0.000	0.002	0.003
VendorID	0.0001	2.99e-06	44.416	0.000	0.000	0.000
PULocationID	-6.822e-07	1.53e-08	-44.489	0.000	-7.12e-07	-6.52e-07
DOLocationID	4.569e-06	1.48e-08	307.769	0.000	4.54e-06	4.6e-06
RatecodeID	-0.0025	8.79e-06	-289.972	0.000	-0.003	-0.003
passenger_count	-1.84e-05	1.09e-06	-16.817	0.000	-2.05e-05	-1.63e-05
trip_distance	0.5137	0.000	1478.285	0.000	0.513	0.514
payment_type	0.0022	2.42e-06	891.619	0.000	0.002	0.002
trip_type	0.0076	3.56e-05	214.293	0.000	0.008	0.008
total_amount	12.0497	0.003	3592.564	0.000	12.043	12.056
speed	-0.0590	1.48e-05	-3998.262	0.000	-0.059	-0.059
holidays	0.0004	3.28e-06	117.538	0.000	0.000	0.000

3.1. 0.25 quantile summary for model

QuantReg Regression Results

Dep. Variable:	duration	Pseudo R-squared:	0.6974
Model:	QuantReg	Bandwidth:	6.537e-05
Method:	Least Squares	Sparsity:	0.008303
Date:	Sun, 01 Dec 2019	No. Observations:	19079450
Time:	22:25:20	Df Residuals:	19079438
		Df Model:	11

	coef	std err	t	P>|t|	[95.0% Conf.	Int.]
Intercept	0.0022	1.8e-05	121.586	0.000	0.002	0.002
VendorID	0.0002	2.5e-06	95.266	0.000	0.000	0.000
PULocationID	-1.146e-07	1.28e-08	-8.957	0.000	-1.4e-07	-8.95e-08
DOLocationID	2.911e-06	1.25e-08	233.317	0.000	2.89e-06	2.94e-06
RatecodeID	-0.0010	5.34e-06	-192.752	0.000	-0.001	-0.001
passenger_count	-2.056e-05	9.18e-07	-22.382	0.000	-2.24e-05	-1.88e-05
trip_distance	0.5754	0.000	2780.585	0.000	0.575	0.576
payment_type	0.0031	1.96e-06	1593.014	0.000	0.003	0.003
trip_type	0.0065	2.12e-05	304.001	0.000	0.006	0.006
total_amount	14.0476	0.002	7148.205	0.000	14.044	14.051
speed	-0.0663	1.16e-05	-5720.916	0.000	-0.066	-0.066
holidays	0.0002	2.75e-06	71.055	0.000	0.000	0.000

3.2. 0.50 quantile summary for model

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

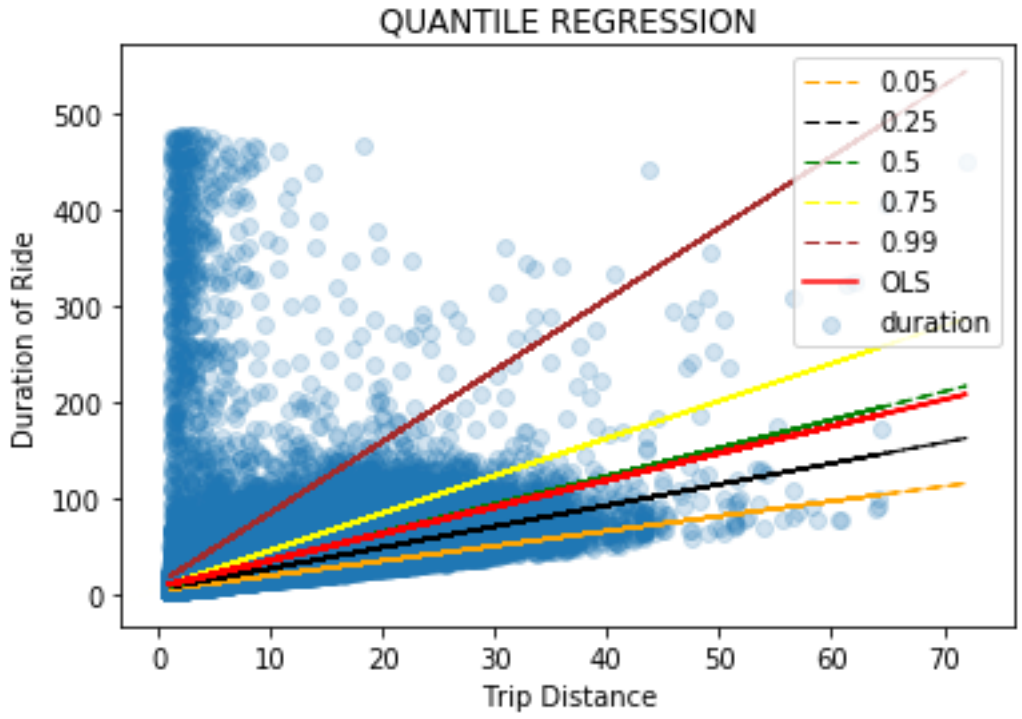
QuantReg Regression Results

Dep. Variable:	duration	Pseudo R-squared:	0.7422
Model:	QuantReg	Bandwidth:	5.598e-05
Method:	Least Squares	Sparsity:	0.01224
Date:	Sun, 01 Dec 2019	No. Observations:	19079450
Time:	23:46:56	Df Residuals:	19079438
		Df Model:	11

	coef	std err	t	P>|t|	[95.0% Conf.	Int.]
Intercept	-0.0005	1.99e-05	-22.772	0.000	-0.000	-0.000
VendorID	0.0006	3.19e-06	186.731	0.000	0.001	0.001
PULocationID	-5.039e-07	1.61e-08	-31.279	0.000	-5.35e-07	-4.72e-07
DOLocationID	4.917e-07	1.58e-08	31.053	0.000	4.61e-07	5.23e-07
RatecodeID	0.0008	5.81e-06	144.530	0.000	0.001	0.001
passenger_count	-3.267e-05	1.18e-06	-27.751	0.000	-3.5e-05	-3.04e-05
trip_distance	0.3920	0.000	1581.525	0.000	0.392	0.393
payment_type	0.0030	2.56e-06	1165.935	0.000	0.003	0.003
trip_type	0.0058	2.26e-05	256.992	0.000	0.006	0.006
total_amount	17.9033	0.002	7569.945	0.000	17.899	17.908
speed	-0.0638	1.76e-05	-3626.954	0.000	-0.064	-0.064
holidays	-6.993e-05	3.5e-06	-19.964	0.000	-7.68e-05	-6.31e-05

3.3. 0.75 quantile summary for model

Considering the fluctuating values of the coefficient of trip_distance and most instinctive relationship between trip distance and duration, we further analysed the relationship between trip distance and different ranges in duration of ride. We compare the quantile regressions with the linear regression to draw conclusions.



[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html

We also see that the slopes low incline because of the majority of the trip durations are between 0-150. The analysis indicates that the estimated mean and median (0.5 quantile) coincide. The different slopes with respect their quantiles indicate that different ranges of trip distances affect the trip distances differently. However we may conclude that for this data, linear regression maybe a suitable model in this case considering the coinciding of mean and median.

[https://slundberg.github.io/shap/notebooks/plots/dependence_plot.html]
https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b
https://towardsdatascience.com/deep-quantile-regression-c85481548b5a
http://inversionlabs.com/2016/02/07/using-quantile-regression.html