

Data Analytics Final Project Report

Fall 2022

Dataset:

Taxi Trips in New York City

Target Variable:

Extra (Categorical)

Author: Niloufar Shokri

Prof. Adalbert F.X. Wilhelm

Date: 2022/12/10

Executive Summary

This report is provided for the data analytics project with the goal of performing a reasonable analysis of the data given. The general task of the project is to build a predictive model for the target variable Extra, from Taxi trips in New York City data set. The data in 28454 rows and 23 columns were collected from taxi trips which have taken place in the timespan of 1st to 29th February 2016. The target variable is categorical. Therefore, the algorithm used for machine learning model is the classification.

After loading the data and understanding each variable, I cleaned the data and removed the garbage variables. Then I transformed date/time related variables into datetime type and created 2 new variables and encoded the categorical, ordinal and Boolean variables into numbers so that I can use them in my model.

In the next step, I checked the linear correlation of the variables I performed simple data exploration using visualization for all the features.

After visualization the distributions and outliers, I chose my feature variables and split my data set into test data and train data. Then I selected Logistic Regression algorithm among 5 classification algorithms by performing cross validation on the training data. I trained my model with the training data and made a prediction on the test data using and evaluated the model using classification metrics.

At first, the accuracy of the model was 0.98 and the ROC-AUC was 0.97 but with improving the model using the l2 penalty (which is used in Ridge regression) and removing the variables with small coefficient in the model into 4 (including fare_amount, tip_amount, tolls_amount, total_amount), the model accuracy and ROC-AUC both increased into 1.

Table of Content

1	Introduction	4
2	Data Set: Taxi trips in New York City	4
3	Data Pre-Processing	6
3.1	Data Cleaning	6
3.1.1	Checking for missing values	6
3.1.2	Checking for garbage values	6
3.1.3	Datetime variable	7
3.1.4	Dropping unnecessary columns	7
3.2	Data Transformation.....	7
3.2.1	Transforming the categorical variables	7
4	Data Exploration	8
4.1	Correlation between different features	8
5	Visualization and checking the distribution of each variable	9
5.1.1	Continues Variables	9
5.1.2	Discrete / Categorical Variables.....	13
6	Modeling	16
6.1	Model Selection.....	17
7	Results and Conclusions.....	19
7.1	Fitting the model.....	19
7.2	Improving the model	21
8	References	23

1 Introduction

Data analytics is the collection, transformation, and organization of data in order to draw conclusions, make predictions, and drive informed decision making. In this project, the main goal is to build a predictive model for the target variable **Extra**, from **Taxi trips in New York City** data set. This variable is categorical. Therefore, the algorithm used for machine learning model is the classification type. Classification algorithms utilize input training data for the purpose of predicting the likelihood or probability that the data that follows will fall into one of the predetermined categories.

After loading the data and understanding each variable, I clean the data and perform data exploration using visualization. In the next step, I choose a classification algorithm by performing cross validation. Then I train my model on the training data and evaluated the model on the test data. The detail of each phase is explained in the report.

2 Data Set: Taxi trips in New York City

The data used in the attached datasets were collected and provided to the NYC Taxi and Limousine Commission (TLC) by technology providers authorized under the Taxicab & Livery Passenger Enhancement Programs (TPEP/LPEP). The data were collected for the taxi trips which have taken place in the timespan of 1st to 29th February 2016.

The dataset is provided in 'csv' type. After importing the data set, the first step is getting some information about shape of the data set and variables.

The shape of the dataset is (28454, 23) which essentially means that there are 28454 rows and 23 columns in the dataset.

The features of the raw data set with their corresponding descriptions are as below:

Features	Description	Type
Unnamed: 0	Index of the dataset	int64
VendorID	A code indicating the TPEP provider that provided the record which is integer value of 1 for Creative Mobile Technologies and 2 for VeriFone Inc.	int64
tpcp_pickup_datetime	The date and time when the meter was engaged.	object
tpcp_dropoff_datetime	The date and time when the meter was disengaged.	object
passenger_count	The number of passengers in the vehicle. This is a driver-entered value.	int64
trip_distance	The elapsed trip distance in miles reported by the taximeter.	float64
pickup_longitude	Longitude where the meter was engaged.	float64
pickup_latitude	Latitude where the meter was engaged.	float64
RatecodeID	The final rate code in effect at the end of the trip which is integer values of 1 to 5 where 1 is for Standard rate, 2 is for JFK, 3 is for Newark, 4 Nassau or Westchester and 5 is for Negotiated fare.	int64
store_and_fwd_flag	This flag indicates whether the trip record was held in vehicle memory before sending to the vendor, aka “store and forward,” as the vehicle did not have a connection to the server. Y= store and forward trip N= not a store and forward trip	object
dropoff_longitude	Longitude where the meter was disengaged.	float64
dropoff_latitude	Latitude where the meter was disengaged.	float64
payment_type	A numeric code (integers from 1 to 4) signifying how the passenger paid for the trip where 1 is for Credit card, 2 is for Cash, 3 is for No charge and 4 for Dispute	int64
fare_amount	The time-and-distance fare calculated by the meter	float64
extra	Miscellaneous extras and surcharges. Currently, this only includes the \$0.50 and \$1 rush hour and overnight charges.	float64
mta_tax	0.50 MTA tax that is automatically triggered based on the metered rate in use.	float64
tip_amount	This field is automatically populated for credit card tips. Cash tips are not included.	float64
tolls_amount	Total amount of all tolls paid in trip.	float64
improvement_surcharge	0.30 improvement surcharge assessed trips at the flag drop. the improvement surcharge began being levied in 2015.	float64
total_amount	The total amount charged to passengers. Does not include cash tips.	float64
GoodTip	Categorical variable indicating an above average tip	bool
Extra	An indicator for additional charges included.	bool
Cash	An indicator whether payment was made by cash or not	bool

3 Data Pre-Processing

3.1 Data Cleaning

The steps followed for the data set is given below:

3.1.1 Checking for missing values

In most of the cases, we do not get complete datasets. They either have some values missing from the rows and columns or they do not have standardized values.

So, before going ahead with the analysis, it is a good idea to check whether the dataset has any missing values. Fortunately, the data set didn't have any missing values.

3.1.2 Checking for garbage values

Garbage value is generally a term meaning that the value in a variable doesn't have some sort of planned meaning.

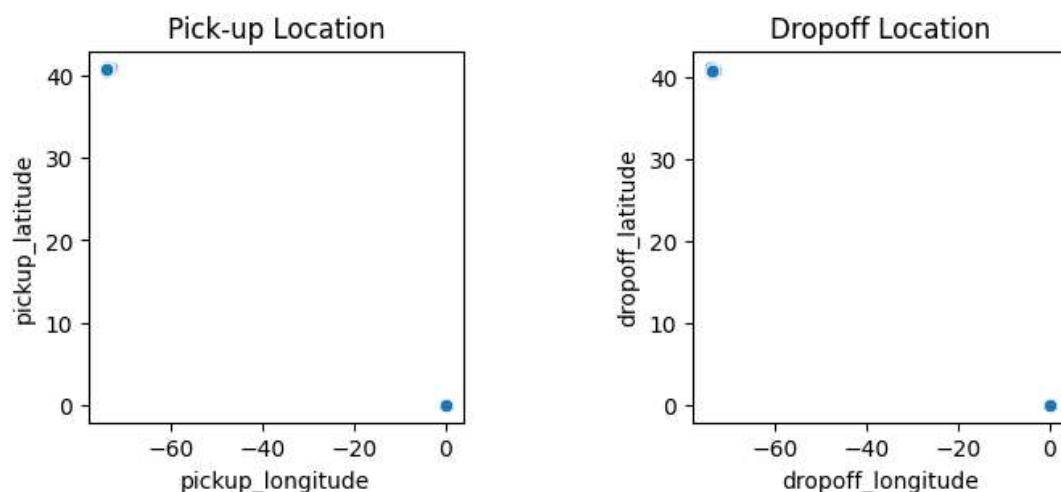
By checking the statistical information of the data, some variables have negative values, and some have 0 values which are not compatible with the definition (corresponding to the dataset).

The detail of these values is given in the following tables:

Variable with negative value	Description
fare_amount	-3.5, 3 times / -4.5, 2 times / -3.0, 2 times / -2.5, 2 times / -60.0, 1 time
extra	-0.5, 1 time / -1.0, 1 time
mta_tax	-0.5, 10 times
improvement_surcharge	-0.3, 10 times
total_amount	-3.5, 3 times / -4.5, 2 times / -3.0, 2 times / -2.5, 2 times / -60.0, 1 time

Variable with 0 value	Description
passenger_count	1 time
trip_distance	154 times
fare_amount	6 times
pickup_latitude and pickup_longitude	477 times
dropoff_latitude and dropoff_longitude	444 times

The pickup and dropoff longitude and latitude equal to 0 belong to Null Island which is in international waters in the Atlantic Ocean. Therefore, these values should be filtered as well.



By removing the above-mentioned garbage values with filtering, the shape of our data set changes to (27755, 23).

3.1.3 Datetime variable

In the data set, we have two variables named “tpep_pickup_datetime” and “tpep_dropoff_datetime” which are date time variables, but their format is string. Therefore, I converted them into datetime format and created 2 new variables, “duration_permin” which is trip duration per minute and another one is the weekday.

By checking the statistical description of “duration_permin”, there are 2 observations with 0 duration. So, I removed them as they are considered as garbage values, and the number of observations reduced into 27753.

3.1.4 Dropping unnecessary columns

In this phase, first I checked the numeric variables with zero variance (threshold = 0), they do not have any contribution on the model. The near to zero variance variables are as follows:

- mta_tax
- improvement_surcharge

They can be removed from the data set. The variable “Unnamed: 0” is the dataset index which can be removed as well.

3.2 Data Transformation

3.2.1 Transforming the categorical variables

There are 5 variables in the data set which their data type is not number.

“store_and_fwd_flag” which its data type is string of Y or N (categorical). To use this variable in the model, I transformed it into 0 and 1.

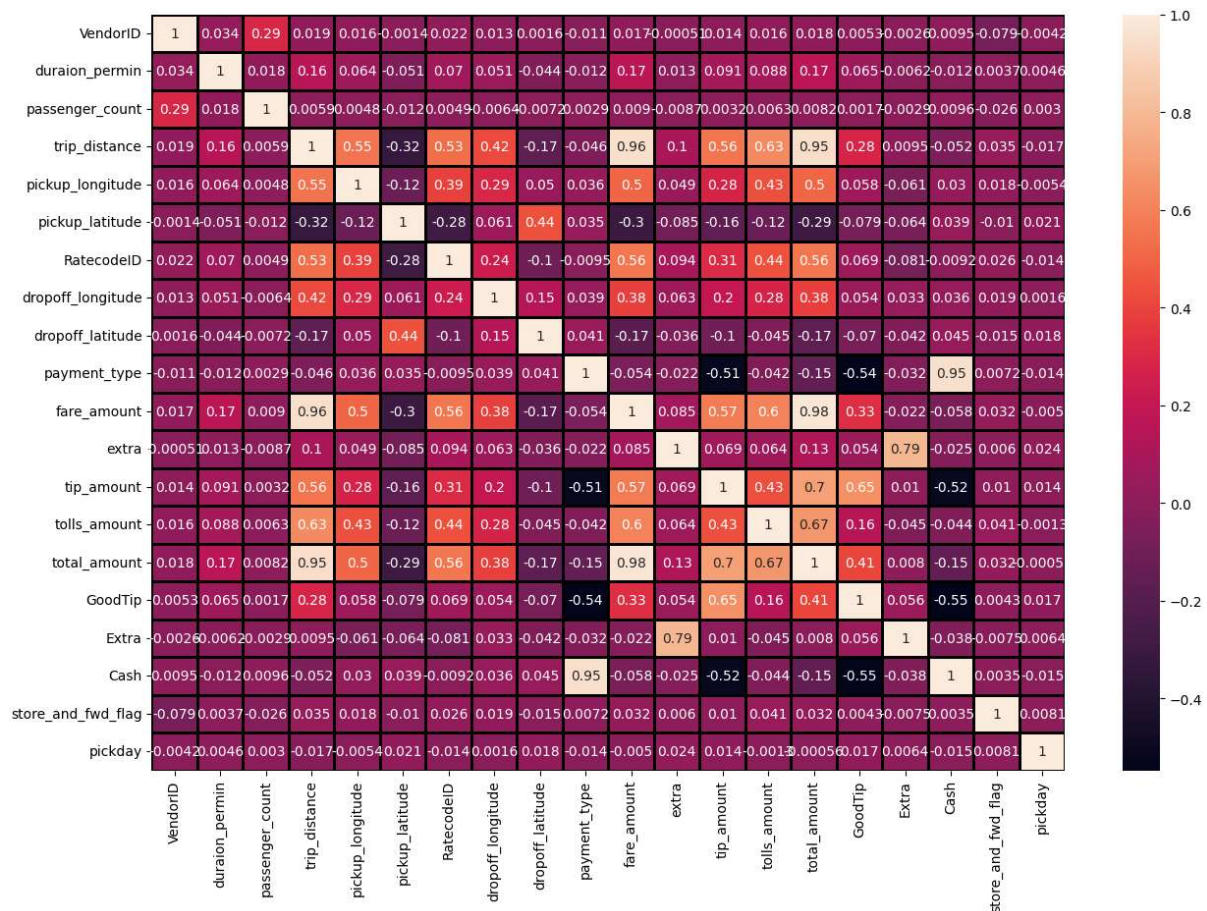
“pickday” which its data type is string of the days of the week (ordinal). To use this variable in my model, I transformed it to integer values in the range of 0, to 6.

“GoodTip”, “Extra” and “Cash” have the data type of Boolean. To use these variables in the model, I transformed all of them into 0 and 1.

4 Data Exploration

4.1 Correlation between different features

Correlation is the way of understanding the strength of the relationship between 2 variables or features in a dataset. Correlation coefficients determine this strength by indicating a value between $[-1,1]$ where -1 indicates a very strong negative relationship, 0 indicates no relationship and 1 indicates strong positive relationship. Pearson correlation is one of the most widely used correlation method and it indicates the linear relationship between 2 variables. The heatmap of correlation between all variables of the dataset is given below:



The sorted correlation matrix for the target variable is as follows:

Variable	Correlation with Extra
extra	0.788995
GoodTip	0.055842
dropoff_longitude	0.033074
tip_amount	0.010252
trip_distance	0.009459
total_amount	0.008006
pickday	0.006360
VendorID	-0.002636
passenger_count	-0.002936
duraion_permin	-0.006169
store_and_fwd_flag	-0.007480
fare_amount	-0.022386
payment_type	-0.032448
Cash	-0.037907
dropoff_latitude	-0.041754
tolls_amount	-0.044502
pickup_longitude	-0.061198
pickup_latitude	-0.064492
RatecodeID	-0.081118

Regarding the heatmap and our correlation matrix, our target variable “Extra” has the strongest linear correlation with “extra”.

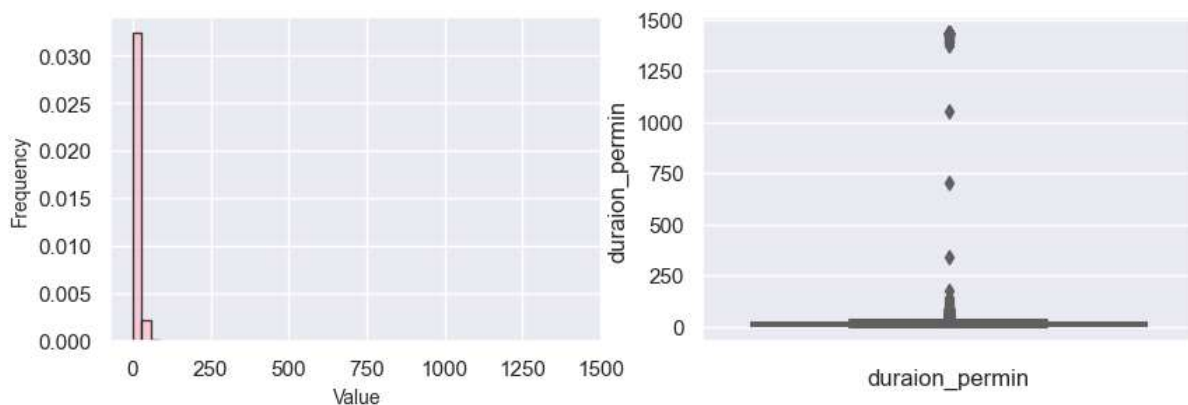
5 Visualization and checking the distribution of each variable

In this part of report, there are some visualizations to understand the distribution of the variables and to check if they have outliers or not.

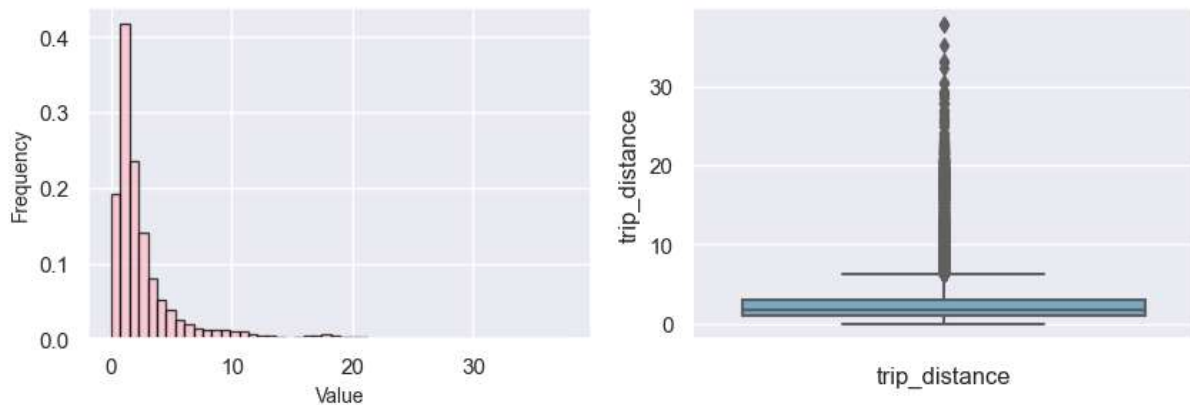
5.1.1 Continues Variables

For each continuous variable, I use boxplot and histogram for visualization. The plots are as bellows:

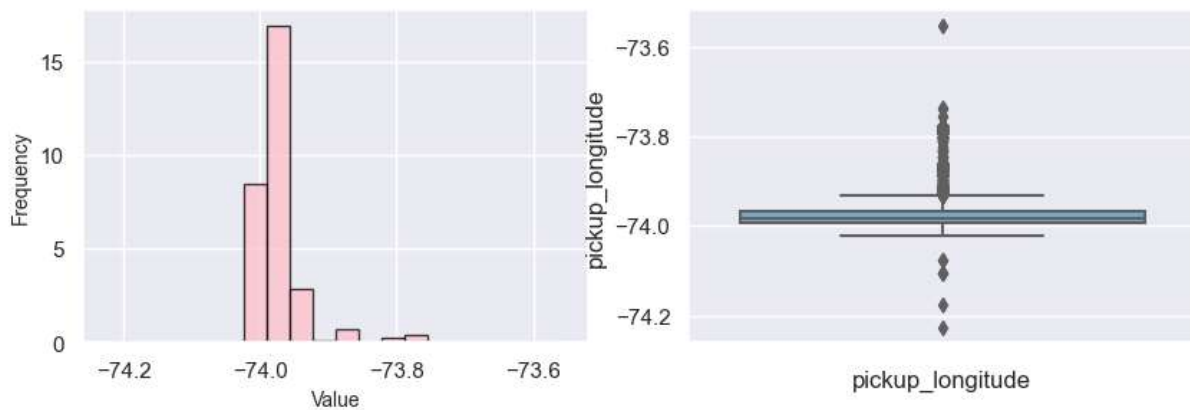
duration_permin is right skewed. There are 144 observations which their duration_permin is more than an hour.



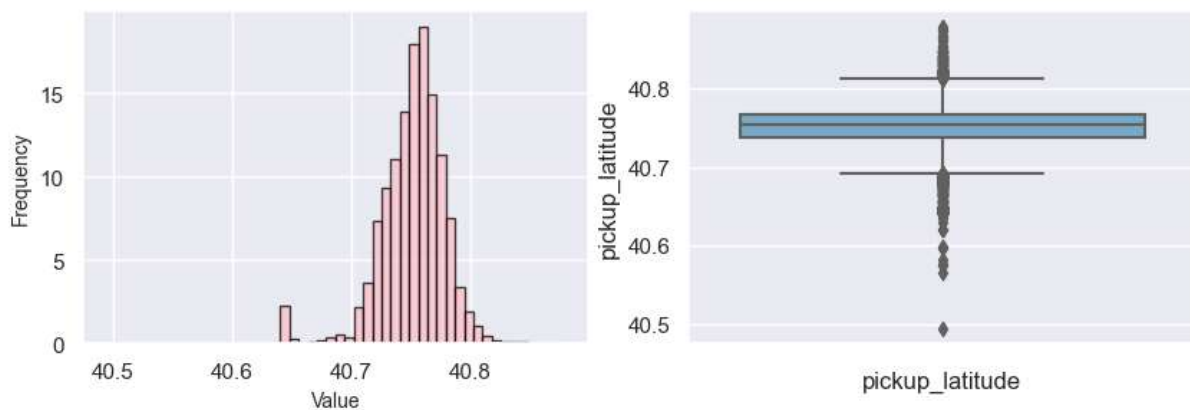
trip_distance is right skewed. There are 123 observations which their trip_distance is very larger than 20 miles.



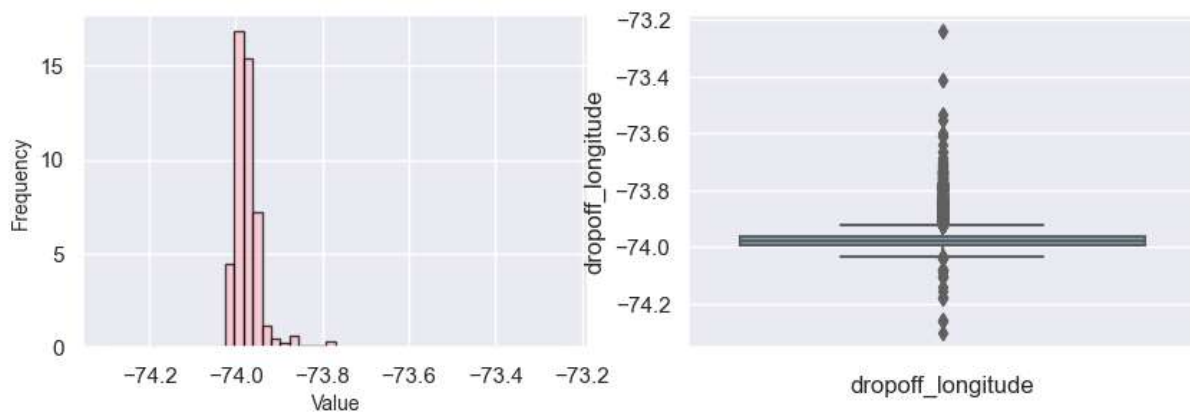
pickup_longitude is not skewed but there are still some outliers in both sides of the median.



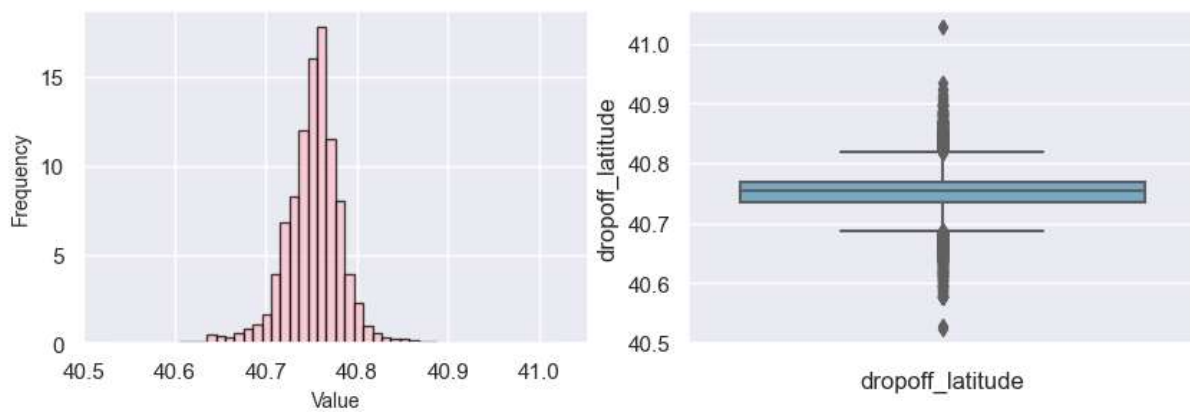
pickup_latitude is not skewed but there are still some outliers in both sides of the median.



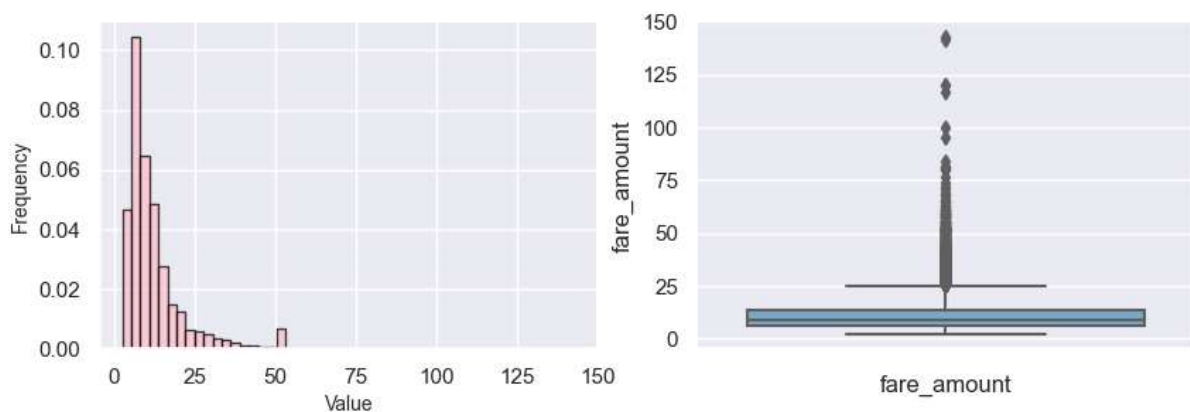
dropoff_latitude is a bit right skewed but there are some outliers in both sides of the median.



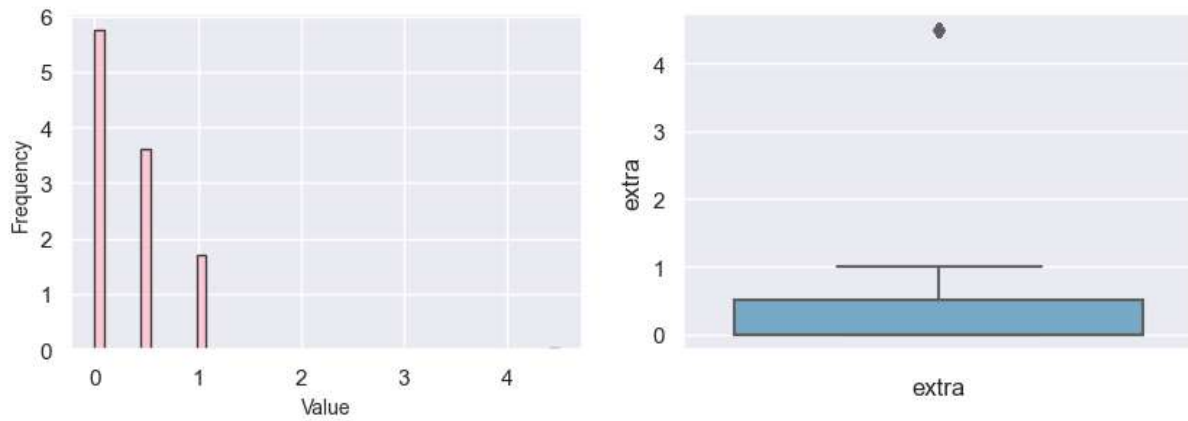
dropoff_longitude is not skewed but there are still some outliers in both sides of the median.



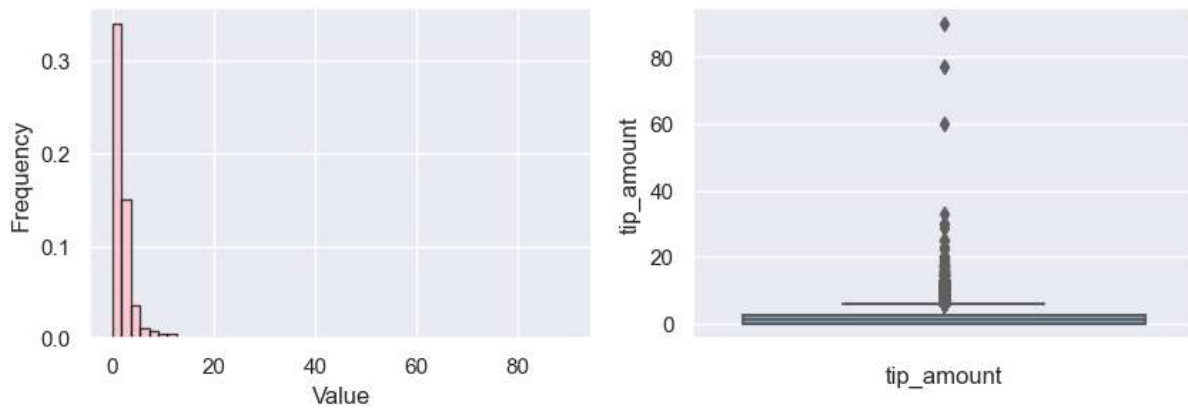
fare_amount is right skewed. There are 90 observations which their fare_amount is very larger than 52 dollars. The outliers are larger than median.



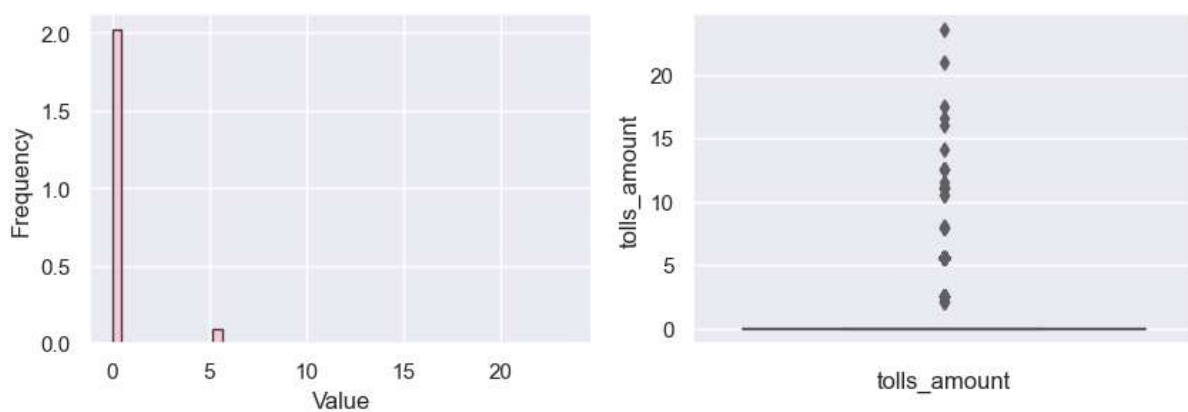
extra variable has 4 values, but there are 85 observations that have the value of 4.5.



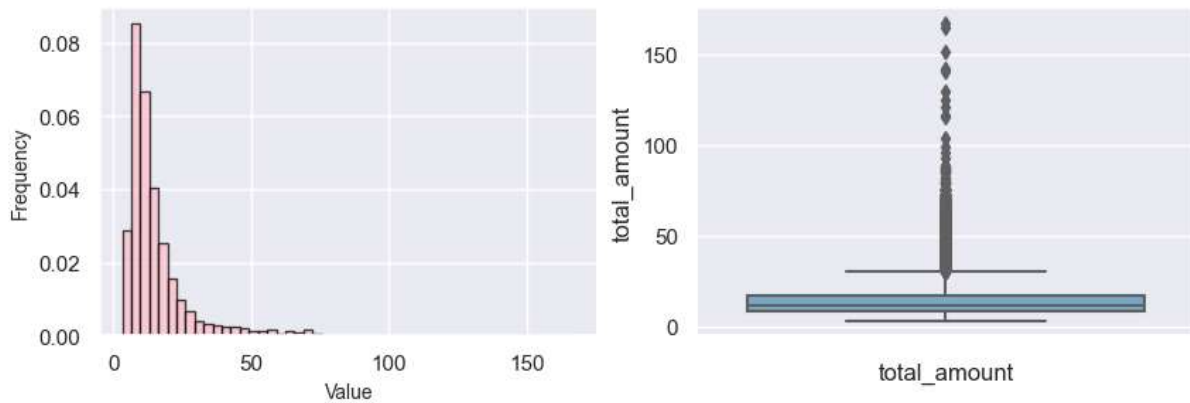
tip_amount is right skewed. There are 40 observations which their tip_amount is larger than 15 dollars.



tolls_amount is right skewed. There are 24 observations which their tolls_amount is larger than 5.54 dollars.

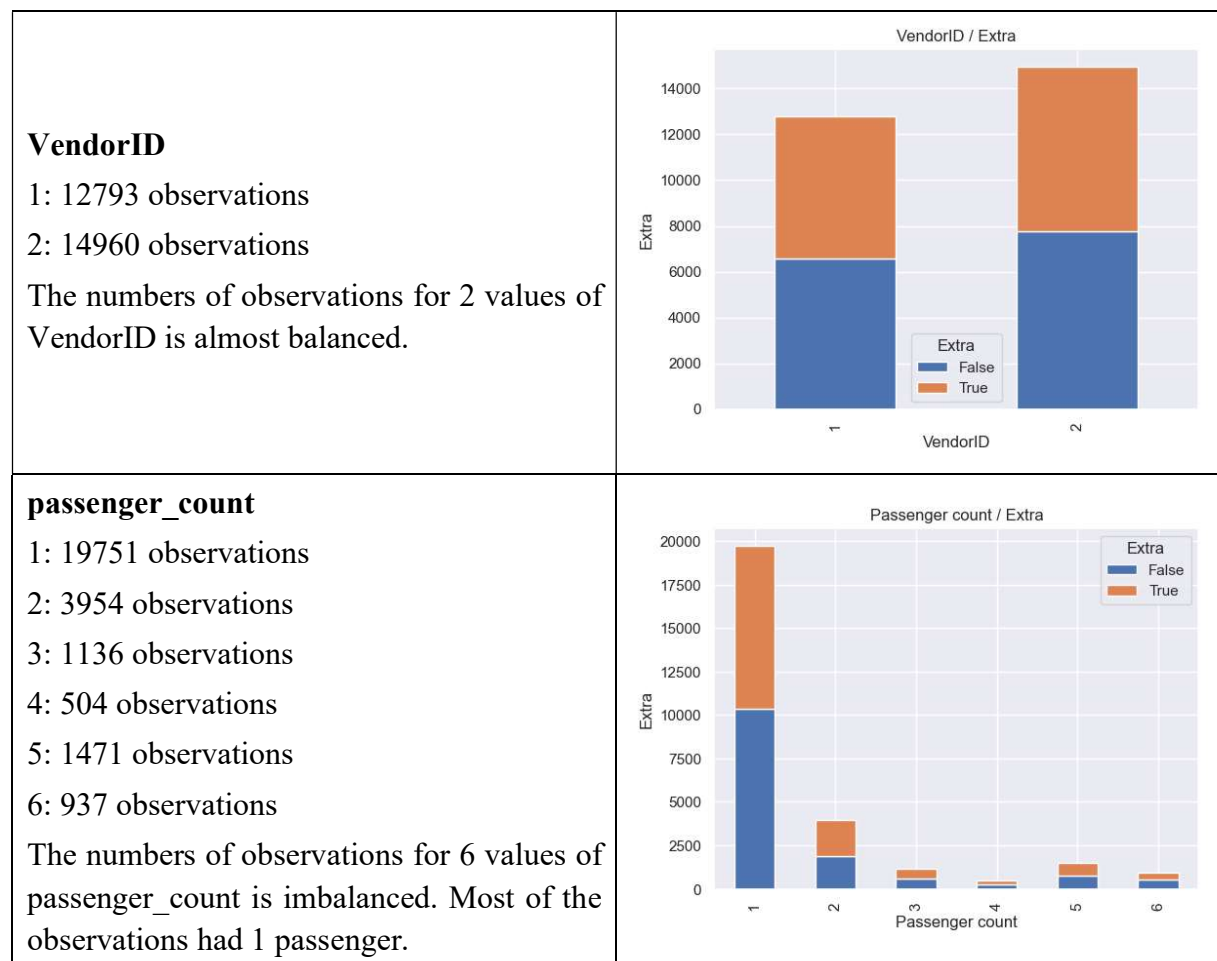


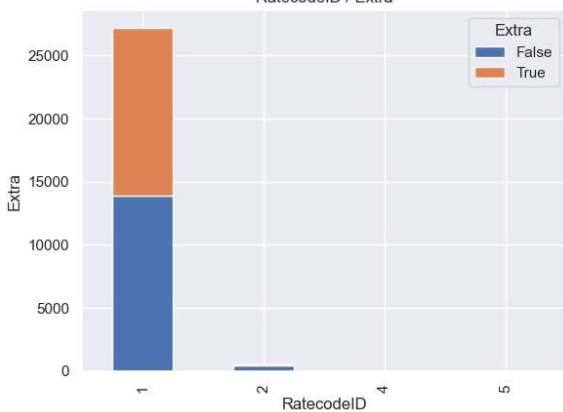
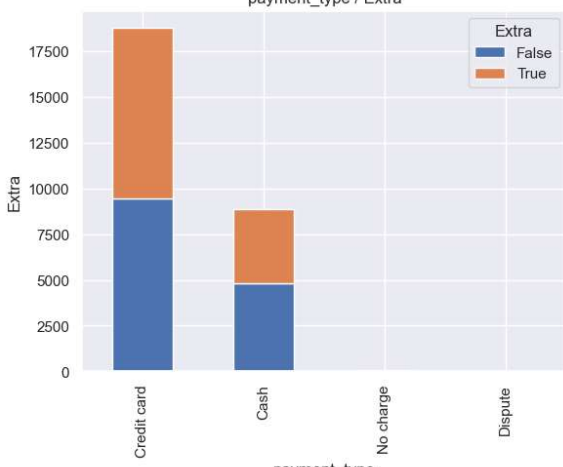
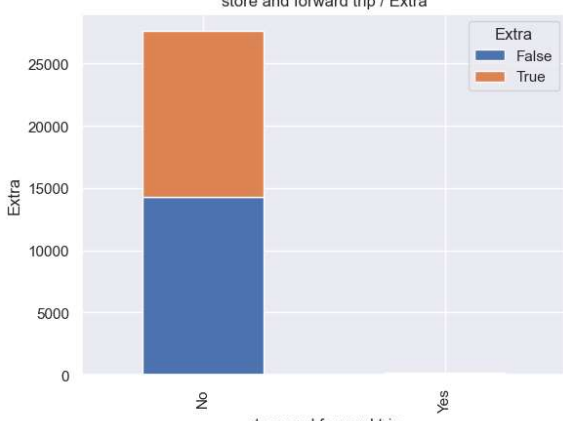
total_amount is right skewed. There are a few observations which their total_amount is larger than 50 dollars.

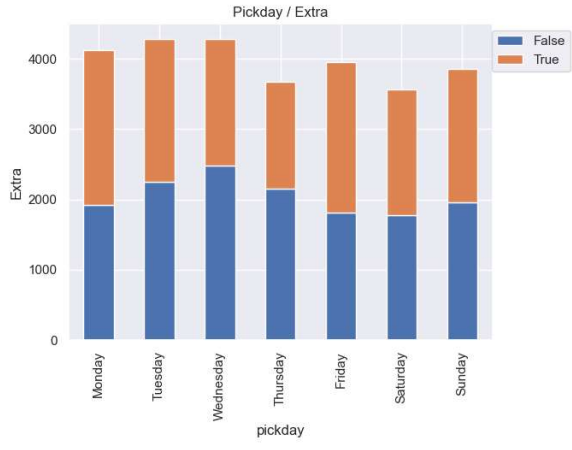
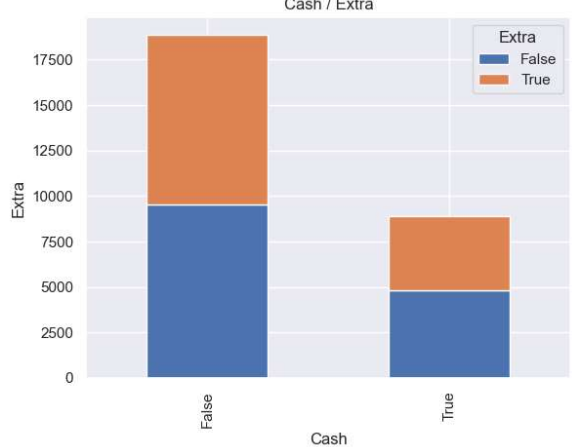
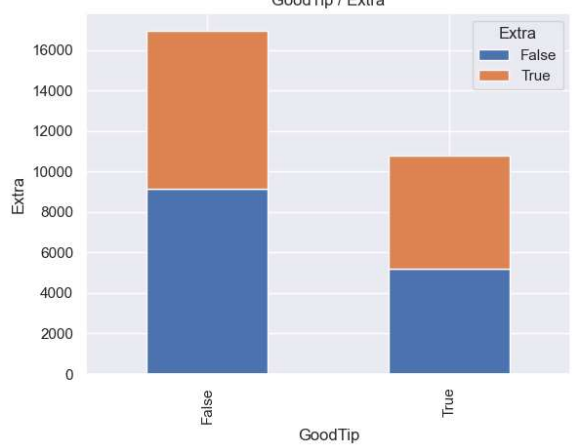


5.1.2 Discrete / Categorical Variables

For each discrete variable (including Booleans), I used bar charts highlighting the target variable for each value. The plots are as bellows:



<p>RatecodeID</p> <p>1: 27223 observations</p> <p>2: 506 observations</p> <p>4: 10 observations</p> <p>5:14 observations</p> <p>The numbers of observations for 4 values of RatecodeID is imbalanced. Most of the observations had RatecodeID of 1 and there was not any RatecodeID for Newark.</p>	 <table><thead><tr><th>RatecodeID</th><th>Extra = False</th><th>Extra = True</th></tr></thead><tbody><tr><td>1</td><td>13800</td><td>13423</td></tr><tr><td>2</td><td>200</td><td>306</td></tr><tr><td>4</td><td>10</td><td>0</td></tr><tr><td>5</td><td>10</td><td>4</td></tr></tbody></table>	RatecodeID	Extra = False	Extra = True	1	13800	13423	2	200	306	4	10	0	5	10	4
RatecodeID	Extra = False	Extra = True														
1	13800	13423														
2	200	306														
4	10	0														
5	10	4														
<p>payment_type</p> <p>Credit card: 18770 observations</p> <p>Cash: 8878 observations</p> <p>No charge: 68 observations</p> <p>Dispute: 37 observations</p> <p>The numbers of observations for 4 values of payment_type is imbalanced. Most of the observations had payment type of credit card or cash.</p>	 <table><thead><tr><th>payment_type</th><th>Extra = False</th><th>Extra = True</th></tr></thead><tbody><tr><td>Credit card</td><td>9500</td><td>9270</td></tr><tr><td>Cash</td><td>4800</td><td>4078</td></tr><tr><td>No charge</td><td>0</td><td>68</td></tr><tr><td>Dispute</td><td>0</td><td>37</td></tr></tbody></table>	payment_type	Extra = False	Extra = True	Credit card	9500	9270	Cash	4800	4078	No charge	0	68	Dispute	0	37
payment_type	Extra = False	Extra = True														
Credit card	9500	9270														
Cash	4800	4078														
No charge	0	68														
Dispute	0	37														
<p>store_and_fwd_flag</p> <p>No: 27607 observations</p> <p>Yes: 146 observations</p> <p>The numbers of observations for 2 values of store_and_fwd_flag is highly imbalanced. Almost all of the observations had store_and_fwd_flag of No.</p>	 <table><thead><tr><th>store and forward trip</th><th>Extra = False</th><th>Extra = True</th></tr></thead><tbody><tr><td>No</td><td>14200</td><td>13407</td></tr><tr><td>Yes</td><td>0</td><td>146</td></tr></tbody></table>	store and forward trip	Extra = False	Extra = True	No	14200	13407	Yes	0	146						
store and forward trip	Extra = False	Extra = True														
No	14200	13407														
Yes	0	146														

<p>pickday</p> <p>Monday: 4127 observations</p> <p>Tuesday: 4283 observations</p> <p>Wednesday: 4288 observations</p> <p>Thursday: 3675 observations</p> <p>Friday: 3952 observations</p> <p>Saturday: 3566 observations</p> <p>Sunday: 3862 observations</p> <p>The numbers of observations for 7 values of pickday is almost balanced.</p>	 <table><caption>Pickday / Extra</caption><thead><tr><th>Pickday</th><th>False</th><th>True</th></tr></thead><tbody><tr><td>Monday</td><td>1900</td><td>2200</td></tr><tr><td>Tuesday</td><td>2200</td><td>2100</td></tr><tr><td>Wednesday</td><td>2400</td><td>1900</td></tr><tr><td>Thursday</td><td>2100</td><td>1600</td></tr><tr><td>Friday</td><td>1800</td><td>2100</td></tr><tr><td>Saturday</td><td>1700</td><td>1900</td></tr><tr><td>Sunday</td><td>1900</td><td>1900</td></tr></tbody></table>	Pickday	False	True	Monday	1900	2200	Tuesday	2200	2100	Wednesday	2400	1900	Thursday	2100	1600	Friday	1800	2100	Saturday	1700	1900	Sunday	1900	1900
Pickday	False	True																							
Monday	1900	2200																							
Tuesday	2200	2100																							
Wednesday	2400	1900																							
Thursday	2100	1600																							
Friday	1800	2100																							
Saturday	1700	1900																							
Sunday	1900	1900																							
<p>Cash</p> <p>False: 18875 observations</p> <p>True: 8878 observations</p> <p>The numbers of observations for True value are almost half of False Value.</p>	 <table><caption>Cash / Extra</caption><thead><tr><th>Cash</th><th>False</th><th>True</th></tr></thead><tbody><tr><td>False</td><td>9500</td><td>9000</td></tr><tr><td>True</td><td>4800</td><td>4100</td></tr></tbody></table>	Cash	False	True	False	9500	9000	True	4800	4100															
Cash	False	True																							
False	9500	9000																							
True	4800	4100																							
<p>GoodTip</p> <p>False: 16959 observations</p> <p>True: 10794 observations</p> <p>The numbers of observations for True value are almost two thirds of False Value.</p>	 <table><caption>GoodTip / Extra</caption><thead><tr><th>GoodTip</th><th>False</th><th>True</th></tr></thead><tbody><tr><td>False</td><td>9000</td><td>7500</td></tr><tr><td>True</td><td>5000</td><td>5800</td></tr></tbody></table>	GoodTip	False	True	False	9000	7500	True	5000	5800															
GoodTip	False	True																							
False	9000	7500																							
True	5000	5800																							

Regarding the above bar charts, the variables such as “RatecodeID” and “store_and_fwd_flag” are near to zero variance variables and can be removed from our features of model.

For other variables, the variable “Extra” is almost balanced distributed in each value.

6 Modeling

Now that the data is clean and the values of our target variable is balanced (True: 13402, False: 14351), it is time to choose our classifier. A summary of each algorithm is described below.

Logistic Regression is a classification method used when the Response column is categorical with only two possible values. The probability of the possible outcomes is modeled with a logistic transformation as a weighted sum of the Predictor columns. The weights or regression coefficients are selected to maximize the likelihood of the observed data.

Linear Discriminant Analysis or Normal Discriminant Analysis or Discriminant Function Analysis is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e. separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space. Linear discriminant analysis is popular when we have more than two response classes, because it also provides low-dimensional views of the data

K-Nearest Neighbors algorithm, also known as KNN or k-NN, is a non-parametric algorithm (which means it does not make any assumption on underlying data), supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. a class label is assigned based on a majority vote.

Decision Tree is a type of supervised machine learning used to categorize or make predictions based on how a previous set of questions were answered. The model is a form of supervised learning, meaning that the model is trained and tested on a set of data that contains the desired categorization. The tree can be explained by two entities, namely decision nodes and leaves.

Random Forest is a collection (a.k.a. ensemble) of many decision trees. A decision tree is a flow chart which separates data based on some condition. If a condition is true, you move on a path otherwise, you move on to another path.

At the first step of modeling, I decided to select 17 independent variables to put in the model. The variables are as follows:

VendorID	payment_type
duraion_permin	fare_amount
passenger_count	extra
trip_distance	tip_amount
pickup_longitude	tolls_amount
pickup_latitude	total_amount
RatecodeID	pickday
dropoff_longitude	Cash
dropoff_latitude	GoodTip

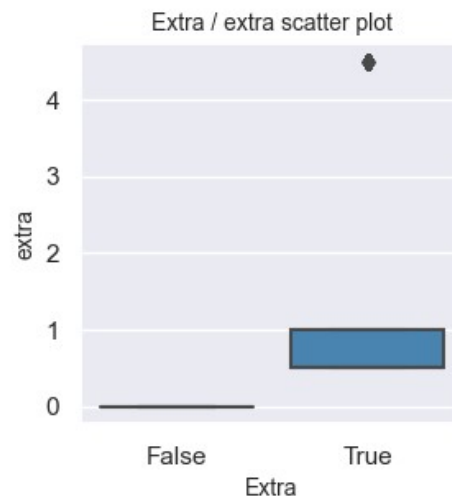
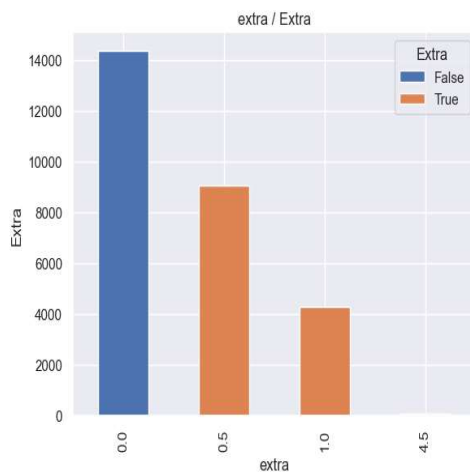
6.1 Model Selection

In order to select my classifier, I performed a 10-fold cross validation algorithm on the above-mentioned classification models and calculated the accuracy (average of all 10 folds) of each model. The result of the cross validation is as follows:

Algorithm	Model Accuracy (Average of 10 folds)	Standard Deviation (Of 10 folds)
Logistic Regression	1.000000	0.000000
Linear Discriminant Analysis	0.996892	0.000911
K-Nearest Neighbors	0.831142	0.008900
Decision Tree	1.000000	0.000000
Random Forest	1.000000	0.000000

The results shows that all the classification algorithms except Linear Discriminant Analysis and K-Nearest Neighbors have the accuracy of 100%. The result is perfect, but it is not normal that almost all the classifiers are giving a perfect result at the first step and without tuning the hyperparameters and feature engineering.

Therefore, the data must be surveyed precisely. Our target variable had the strongest linear correlation with variable “extra”. With a closer look at below bar and box plots, I found out that **Extra** is a Boolean variable driven from **extra**, with the rule that if $\text{extra} = 0$, $\text{Extra} = \text{False}$, if $\text{extra} > 0$, $\text{Extra} = \text{True}$.



To check this more precisely, I removed **extra** from my feature variables and performed a Random Forest Classifier model. I split my data set into train (80% of the observation) and test (20% of the observation), fitted the model on train data and performed a prediction on my test data. The classification report is as follows; the accuracy reduced into 80%.

	precision	recall	f1-score	support
False	0.79	0.85	0.82	2902
True	0.82	0.75	0.78	2649
accuracy			0.80	5551
macro avg	0.80	0.80	0.80	5551
weighted avg	0.80	0.80	0.80	5551

Therefore, I will remove “extra” from the model to have a more real and correct model.

After removing **extra** variable from the features, I performed a 10-fold cross validation algorithm for the second time on the classification models and calculated the accuracy (average of all 10 folds) of each model. The result of the cross validation is as follows:

Algorithm	Model Accuracy (Average of 10 folds)	Standard Deviation (Of 10 folds)
Logistic Regression	0.999685	0.000288
Linear Discriminant Analysis	0.996802	0.000792
K-Nearest Neighbors	0.724124	0.010515
Decision Tree	0.756868	0.023278
Random Forest	0.809296	0.003650

Regarding the accuracy score, I chose **Logistic Regression** as my classifier as its accuracy score surpassed Random Forest Classifier and Decision Tree.

7 Results and Conclusions

7.1 Fitting the model

In this part of the report, I explain the steps I took for the algorithm:

Step 1: After deciding what variables to choose, I split my data into train (80% of observations) and test (20% of observations) dataset.

Step 2: I selected following hyper parameters for my model:

Solver = 'lbfgs', Stands for Limited-memory Broyden–Fletcher–Goldfarb–Shanno. It approximates the second derivative matrix updates with gradient evaluations. It stores only the last few updates, so it saves memory. Algorithm to use in the optimization problem. 'lbfgs' is the default solver.

C = 0.3, Inverse of regularization strength which must be positive float. Smaller values specify stronger regularization.

Penalty = l2, l2 penalty function uses the sum of the squares of the parameters and Ridge Regression encourages this sum to be small. l1 penalty function uses the sum of the absolute values of the parameters and Lasso encourages this sum to be small.

n_jobs = -1, Number of CPU cores used when parallelizing over classes. -1 means using all processors

Step 3: After fitting the model on my train data set, I used the 10-fold cross validation to evaluate the accuracy of my model. The result of cross validation is as follows:

```
Cross Validation Scores: [0.99954975 1.          0.99954955 1.          0.99864865 0.99864865
 1.          1.          1.          0.99909991 ]
Average CV Score: 0.9995495698309748
```

The average cross validation score for accuracy metric is high, which shows that the model is not overfitted and will fit the test data very well.

Step 4: I performed a prediction on my test data and used classification metrics such as confusion matrix, classification report and ROC AUC to evaluate the performance of the model.

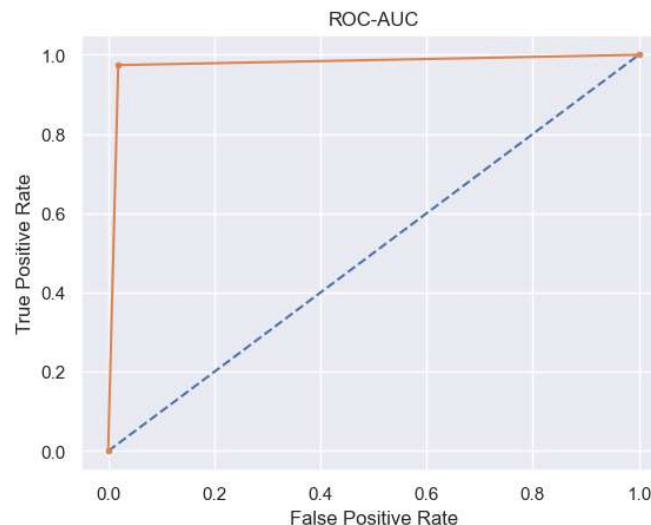
Based on the results of the model prediction, the confusion matrix and classification report are as follows:

Confusion Matrix	Predicted as False	Predicted as True
Actual False	2847 (TP)	55 (FN)
Actual True	69 (FP)	2580 (TN)

The confusion matrix depicts that the number of actual False values that the model truly predicted False as (TP) is 2847, the number of actual False values that the model wrongly predicted as True (FN) is 55, the number of actual True values that the model truly predicted as True (TN) is 2580 and the number of actual True values that the model wrongly predicted as False (FP) is 69.

	precision	recall	f1-score	support
0	0.98	0.98	0.98	2902
1	0.98	0.97	0.98	2649
accuracy			0.98	5551
macro avg	0.98	0.98	0.98	5551
weighted avg	0.98	0.98	0.98	5551

Receiver operating characteristic (ROC) curve plots true positive rate (sensitivity) vs. false positive rate. The Area Under the ROC curve (AUC) is a measure of how well a parameter can distinguish between two diagnostic groups. Each point on the ROC curve represents a sensitivity/ (1 - specificity) pair corresponding to a particular decision threshold and is used as a performance metric in classification algorithms. The default threshold for interpreting probabilities to class labels is 0.5. The area under the curve of the model is 0.9774999941462638.



7.2 Improving the model

Regarding the classification metrics, the model is performing very well. But let's see if the performance can be improved or not.

In the model hyperparameters, I tuned the penalty λ . λ regularization adds an λ penalty equal to the square of the magnitude of coefficients. By using λ , all coefficients are shrunk, but none are eliminated. This is the method which is used in Ridge regression. So, I removed the features with small coefficient to check the performance of the model. The table of model coefficient is as follows:

Row	Dependent Variable	Coefficient
1	VendorID	-0.187343
2	duraion_permin	-0.000361
3	passenger_count	0.049829
4	trip_distance	1.621235
5	pickup_longitude	-0.360051
6	pickup_latitude	-0.280732
7	RatecodeID	-2.03707
8	dropoff_longitude	0.170097
9	dropoff_latitude	-0.258859
10	payment_type	0.325254
11	fare_amount	-11.620426
12	tip_amount	-10.96205
13	tolls_amount	-12.620044
14	total_amount	11.092745
15	pickday	-0.389677
16	Cash	-0.011924
17	GoodTip	1.952415

By looking at the coefficients of the model, I keep below model and check the model performance with new features.

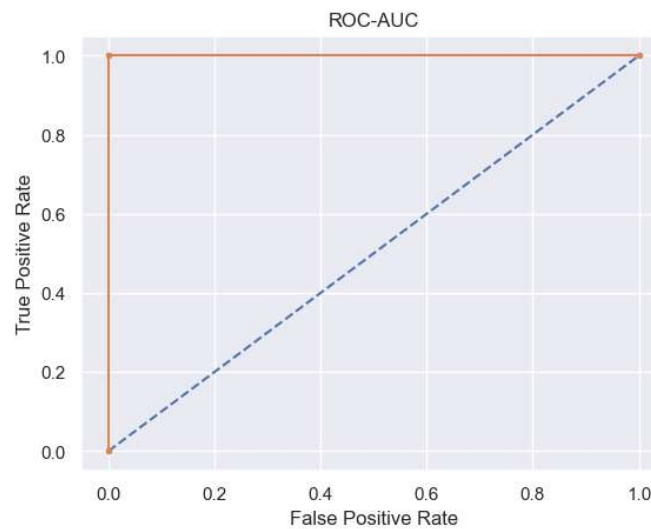
- fare_amount
- tip_amount
- tolls_amount
- total_amount

I performed the 4 steps again and the result are as follow:

Confusion Matrix	Predicted as False	Predicted as True
Actual False	2902 (TP)	0 (FN)
Actual True	0 (FP)	2649 (TN)

	precision	recall	f1-score	support
False	1.00	1.00	1.00	2902
True	1.00	1.00	1.00	2649
accuracy			1.00	5551
macro avg	1.00	1.00	1.00	5551
weighted avg	1.00	1.00	1.00	5551

The area under the curve of the model is 1 which shows the model predicts very well.



The new coefficients of the model are as follows:

Row	Dependent Variable	Coefficient
1	fare amount	-12.948226
2	tip amount	-12.915728
3	tolls amount	-12.940069
4	total amount	12.942052

By eliminating each of these features the accuracy of the model reduces dramatically based on table below:

Eliminating the variable	accuracy
fare amount	0.54
tip amount	0.6
tolls amount	0.81
total amount	0.54

The prediction model with 4 features performs perfectly on the test data and all the classification metrics are 100%.

However, I again used 10-fold class validation on the train data set again and compared other classification algorithms.

Algorithm	Model Accuracy (Average of 10 folds)	Standard Deviation (Of 10 folds)
Logistic Regression	1.000000	0.000000
Linear Discriminant Analysis	0.989731	0.001504
K-Nearest Neighbors	0.961355	0.004084
Decision Tree	0.970183	0.003239
Random Forest	0.970769	0.001652

The results show that the accuracy metric for all other algorithms (except Linear Discriminant Analysis) increased as well. But for Logistic Regression, the accuracy metric is 1.

8 References

Codes, dataset, question paper: <https://github.com/Niillooff/DA-Project.git>