Ford GoBike system

Regression project

YEHUDA FINKELSHTIEN

DANIEL BITON

Introduction

This data set includes information about individual rides made in a bike-sharing system covering the greater San Francisco Bay area.
The dataset will require some data wrangling to make it tidy for analysis.

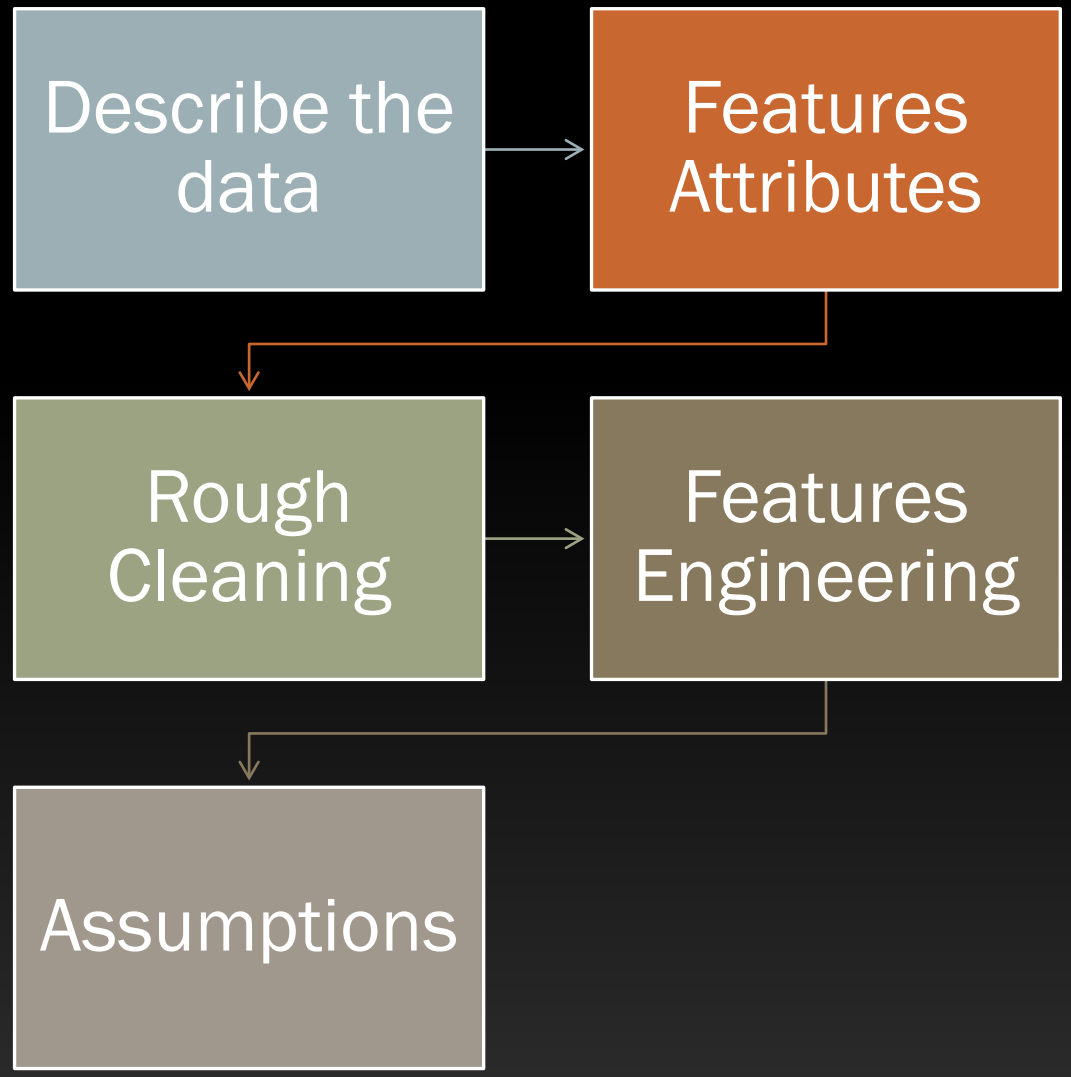
Data source: Kaggle

Prediction target definition

We would like to predict a **duration** of bike-sharing system usage based on the data which collect in the moment that ride is starting according to the location, gender, age and so forth.

This data should contribute the guidelines for the marketing division where to focus on the next campaign based on the attributes were examine.

Exploratory Data Analysis



Data .Info()

NULLs

duration_sec	0
start_time	0
end_time	0
start_station_id	197
start_station_name	197
start_station_latitude	0
start_station_longitude	0
end_station_id	197
end_station_name	197
end_station_latitude	0
end_station_longitude	0
bike_id	0
user_type	0
member_birth_year	8265
member_gender	8265
bike_share_for_all_trip	0

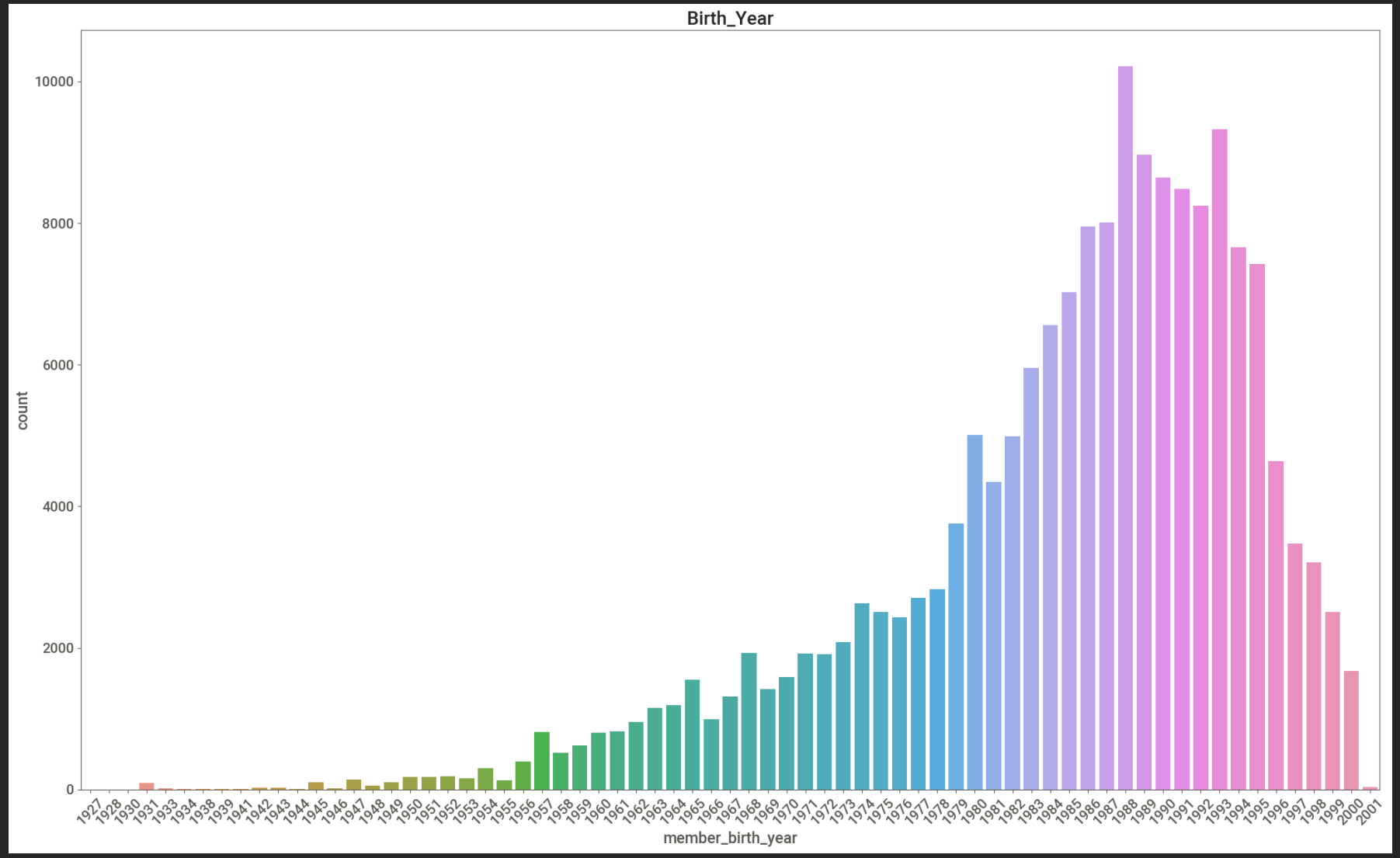
dtype: int64

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 183416 entries, 0 to 183415
Data columns (total 16 columns):
 #   Column                   Non-Null Count   Dtype
---  ------                   --------------   -----
 0   duration_sec             183416 non-null  int64
 1   start_time               183416 non-null  object
 2   end_time                 183416 non-null  object
 3   start_station_id         183219 non-null  float64
 4   start_station_name       183219 non-null  object
 5   start_station_latitude   183416 non-null  float64
 6   start_station_longitude  183416 non-null  float64
 7   end_station_id           183219 non-null  float64
 8   end_station_name         183219 non-null  object
 9   end_station_latitude     183416 non-null  float64
 10  end_station_longitude    183416 non-null  float64
 11  bike_id                  183416 non-null  int64
 12  user_type                183416 non-null  object
 13  member_birth_year        175151 non-null  float64
 14  member_gender            175151 non-null  object
 15  bike_share_for_all_trip  183416 non-null  object
dtypes: float64(7), int64(2), object(7)
memory usage: 22.4+ MB
```

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
duration_sec	183416.0	NaN	NaN	NaN	726.067311	1794.371831	61.0	325.0	514.0	796.0	85444.0
start_time	183416	35783	33:48.5	16	NaN	NaN	NaN	NaN	NaN	NaN	NaN
end_time	183416	35767	22:24.4	20	NaN	NaN	NaN	NaN	NaN	NaN	NaN
start_station_id	183219.0	NaN	NaN	NaN	138.590365	111.778938	3.0	47.0	104.0	239.0	398.0
start_station_name	183219	329	Market St at 10th St	3904	NaN	NaN	NaN	NaN	NaN	NaN	NaN
start_station_latitude	183416.0	NaN	NaN	NaN	37.771221	0.099586	37.317298	37.770083	37.78076	37.79728	37.880222
start_station_longitude	183416.0	NaN	NaN	NaN	-122.352661	0.1171	-122.453705	-122.412408	-122.398285	-122.286533	-121.874119
end_station_id	183219.0	NaN	NaN	NaN	136.249401	111.514916	3.0	44.0	100.0	235.0	398.0
end_station_name	183219	329	San Francisco Caltrain Station 2 (Townsend St...	4857	NaN	NaN	NaN	NaN	NaN	NaN	NaN
end_station_latitude	183416.0	NaN	NaN	NaN	37.771425	0.099494	37.317298	37.770407	37.78101	37.79732	37.880222
end_station_longitude	183416.0	NaN	NaN	NaN	-122.352247	0.116677	-122.453705	-122.411726	-122.398279	-122.288045	-121.874119
bike_id	183416.0	NaN	NaN	NaN	4472.89026	1664.386068	11.0	3777.0	4958.0	5502.0	6645.0
user_type	183416	2	Subscriber	163548	NaN	NaN	NaN	NaN	NaN	NaN	NaN
member_birth_year	175151.0	NaN	NaN	NaN	1984.806504	10.116589	1878.0	1980.0	1987.0	1993.0	2001.0
member_gender	175151	3	Male	130655	NaN	NaN	NaN	NaN	NaN	NaN	NaN
bike_share_for_all_trip	183416	2	No	166056	NaN	NaN	NaN	NaN	NaN	NaN	NaN

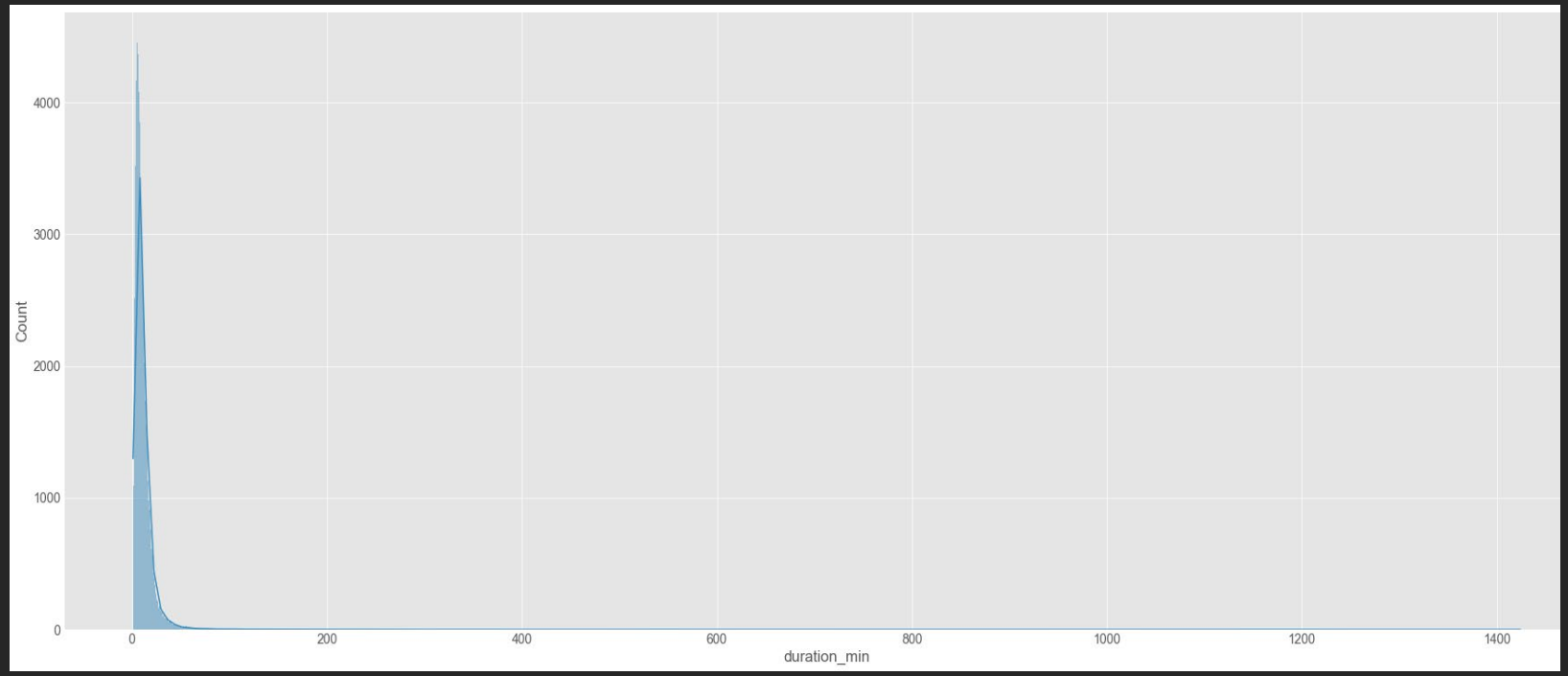
Age Issues

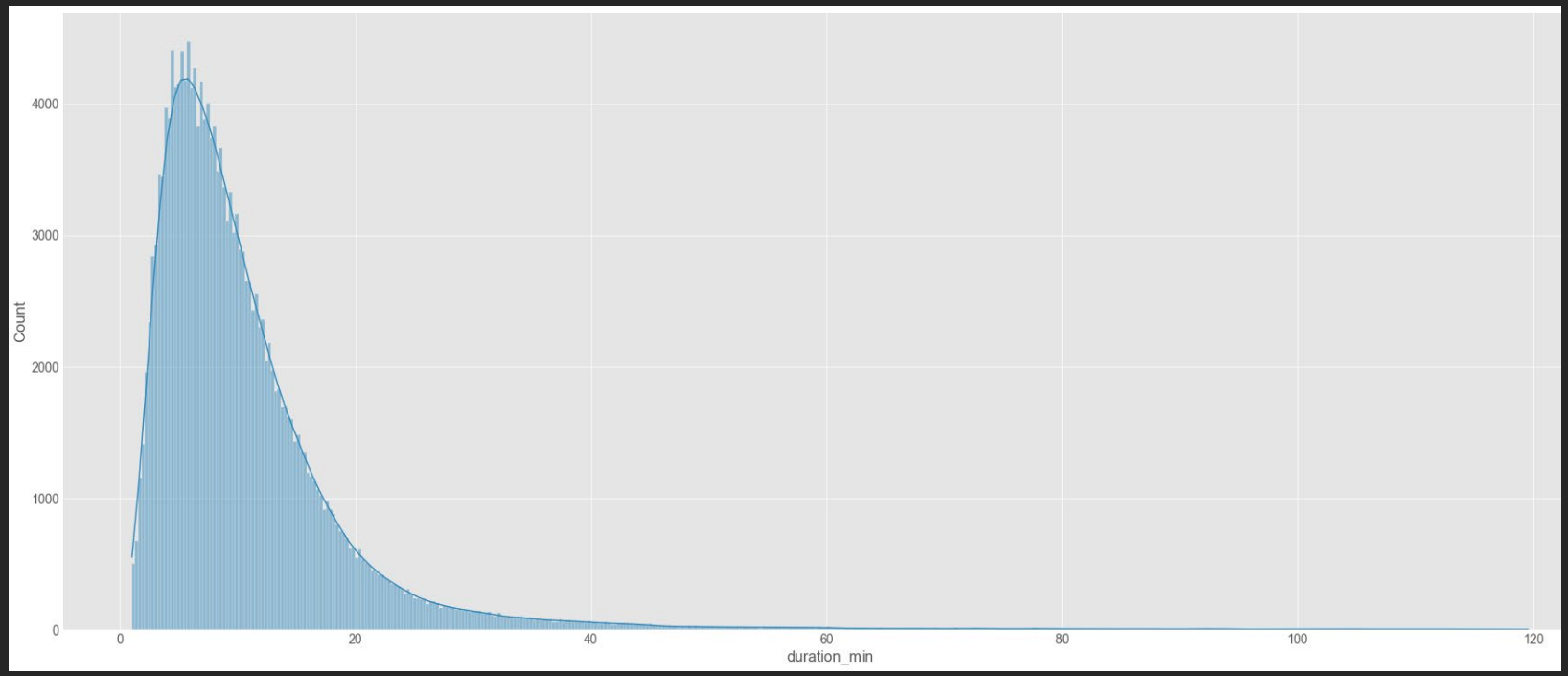
- Cleaning to range of 20-100 years old
- Imputing missing values by average



Duration distribution

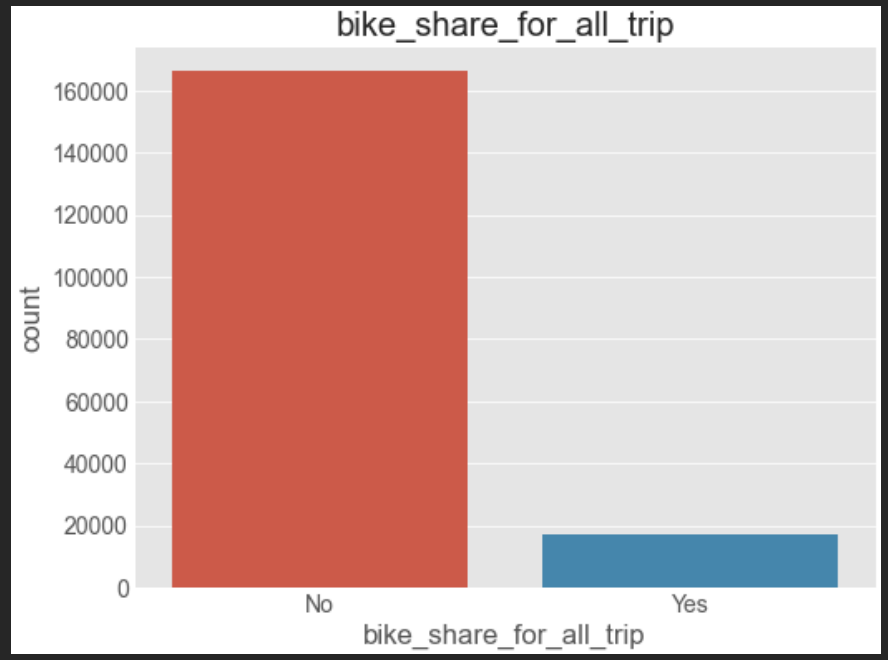
- Cleaning out of range of 120 min

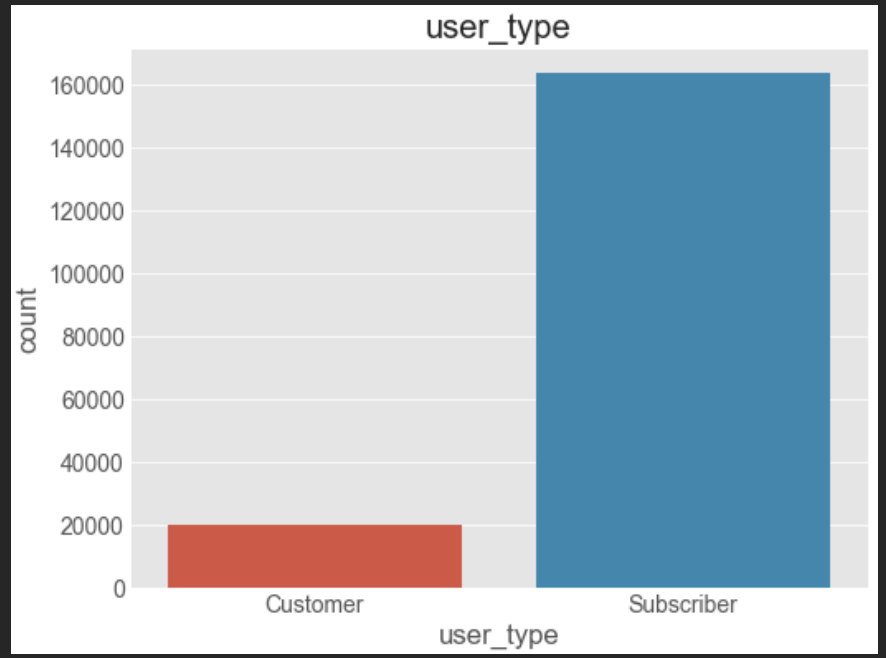




Categorical Features Distribution

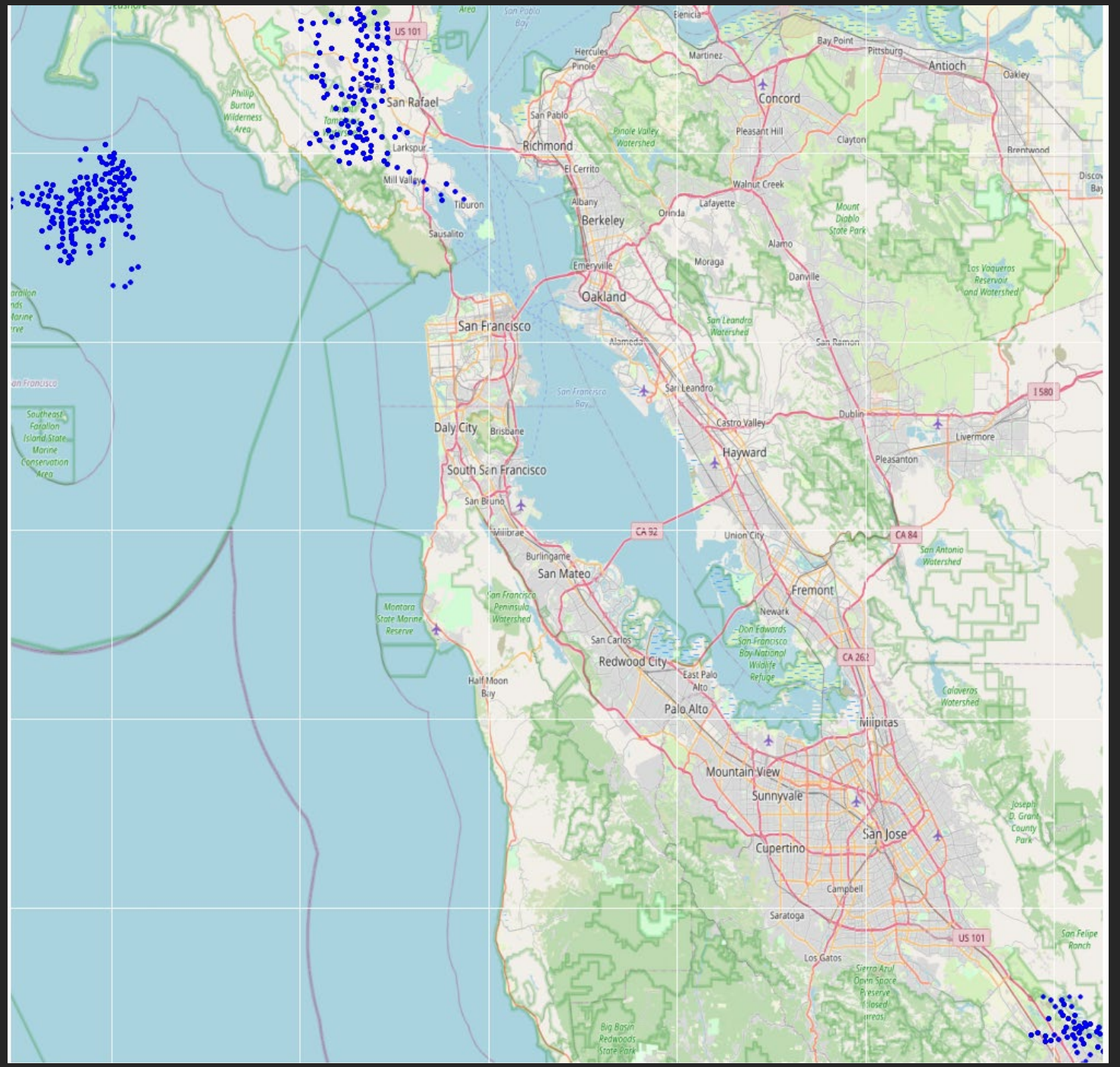
"All_Trip" organizing tours in San Francisco Bay area.

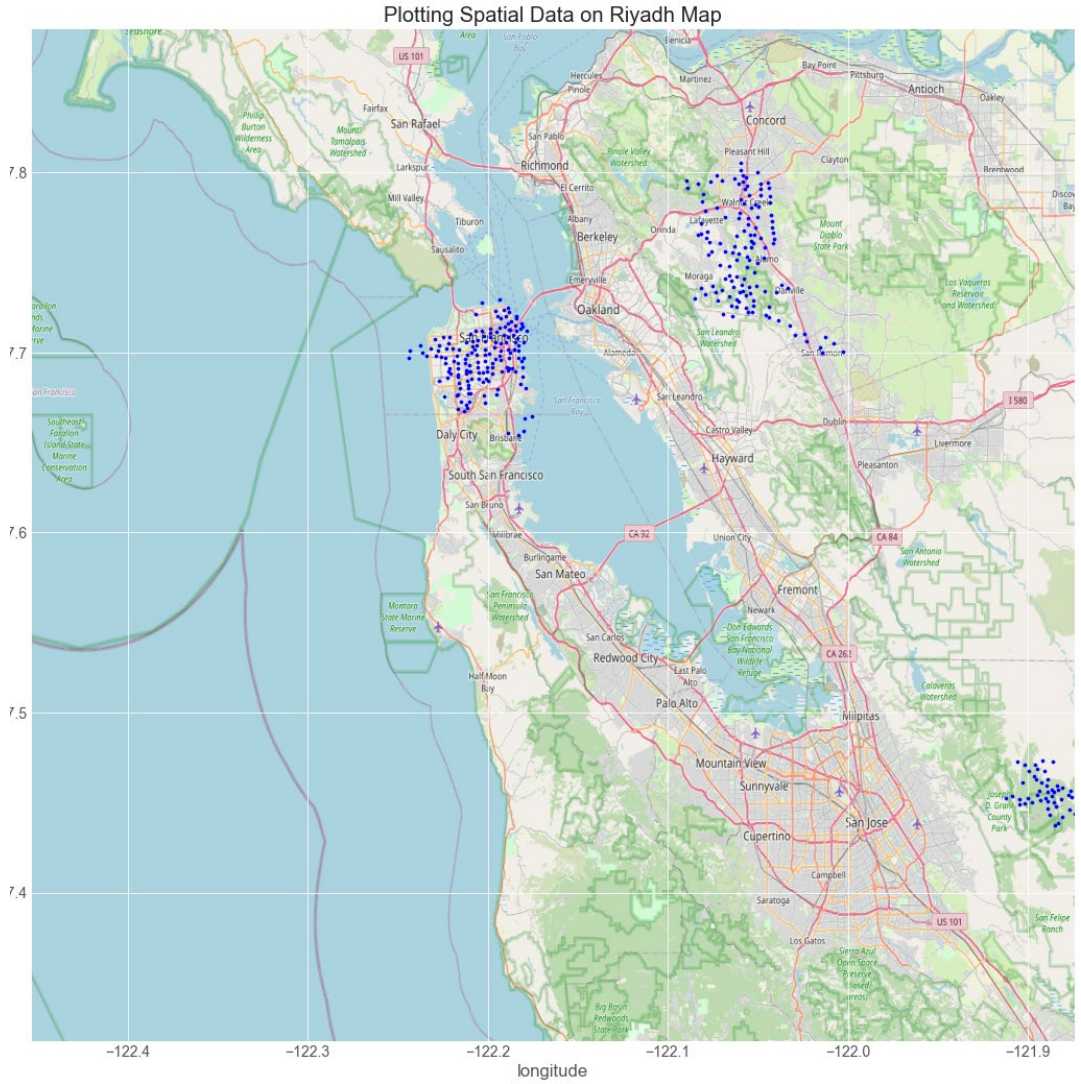


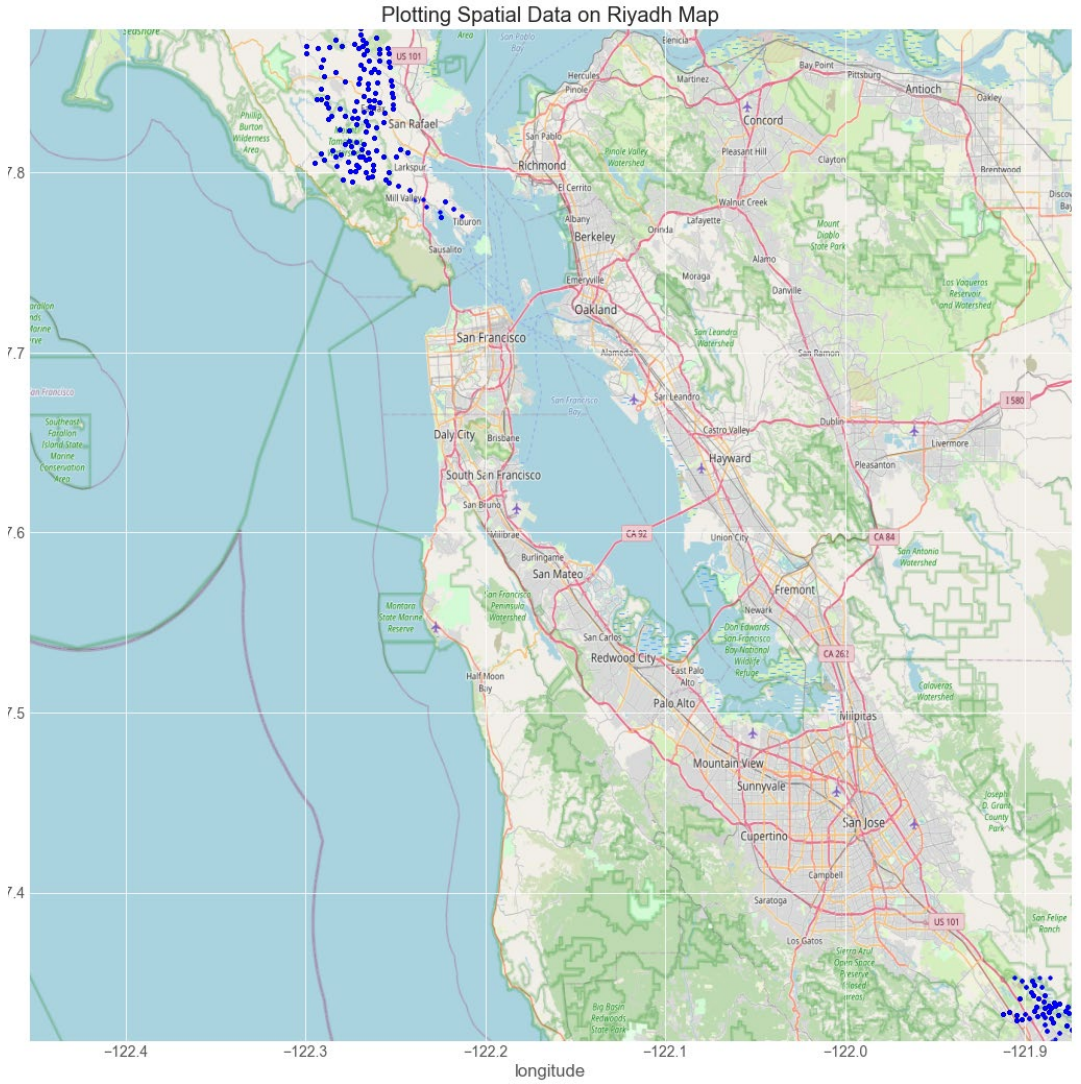


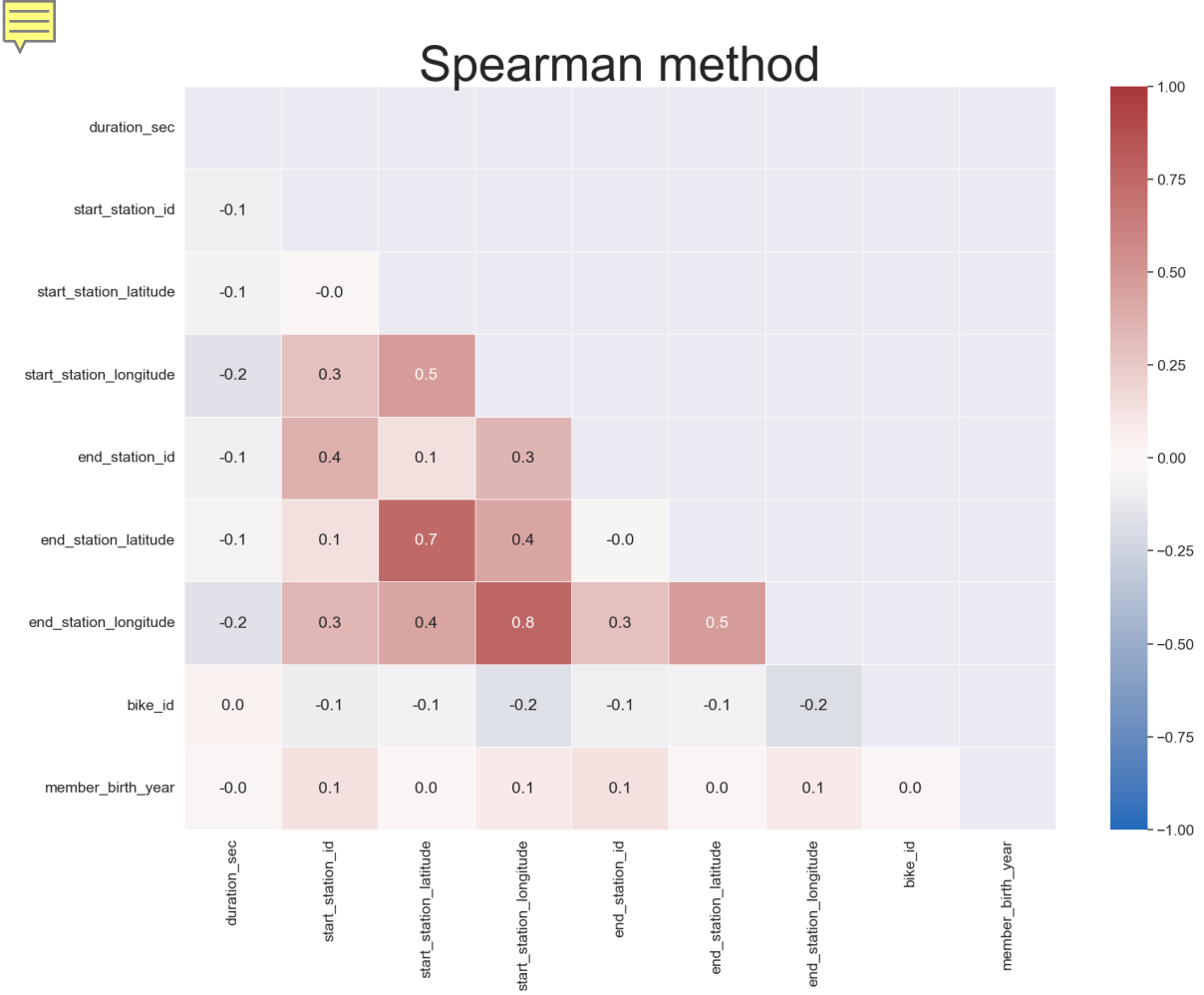
Mapping stations

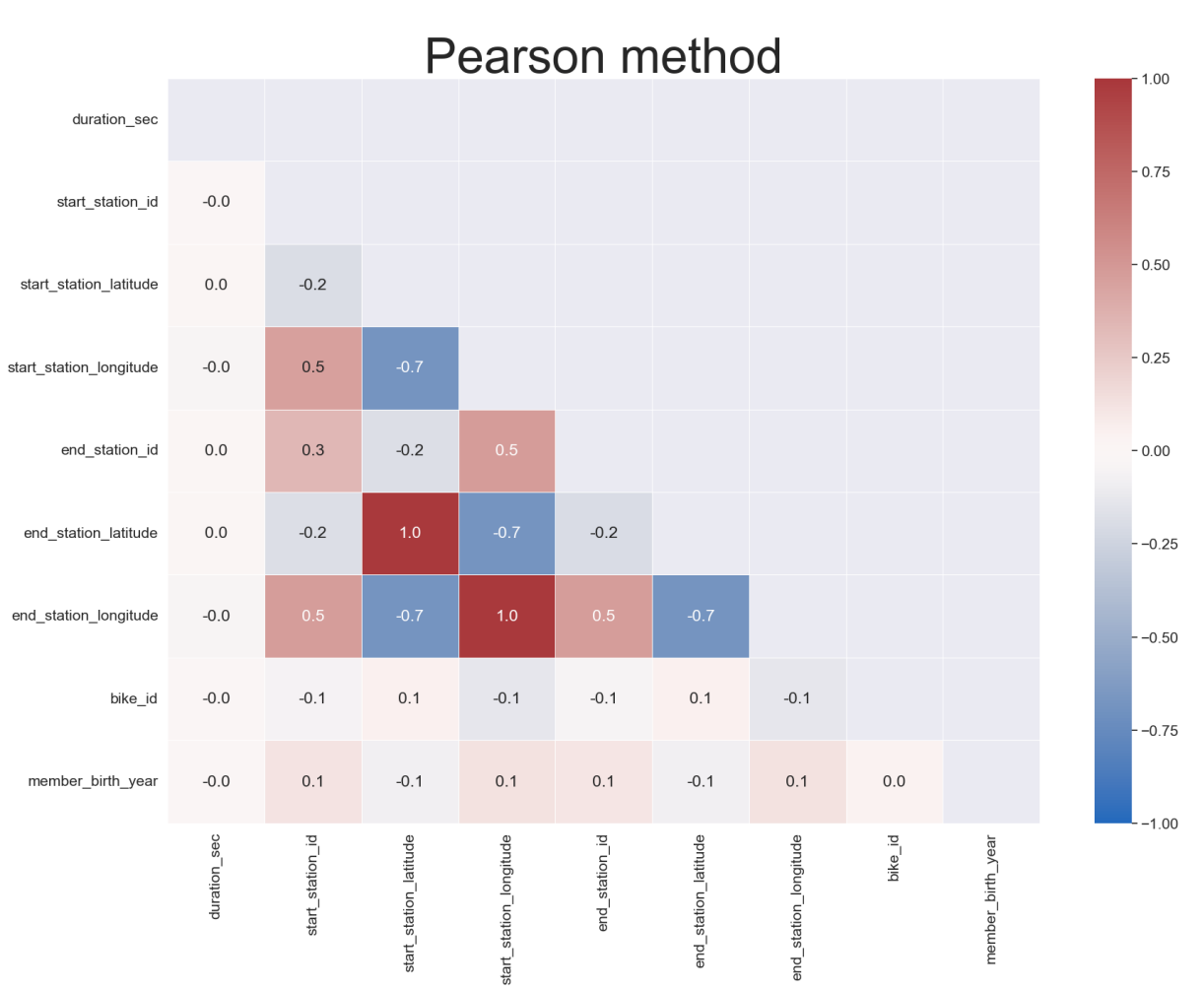
THERE ARE SOME STATIONS WERE ARE LOCATED ON THE SEA




Plotting Spatial Data on Riyadh Map
7.8
7.7
7.6
7.5
7.4
−122.4
−122.3
−122.2
−122.1
−122.0
−121.9
longitude

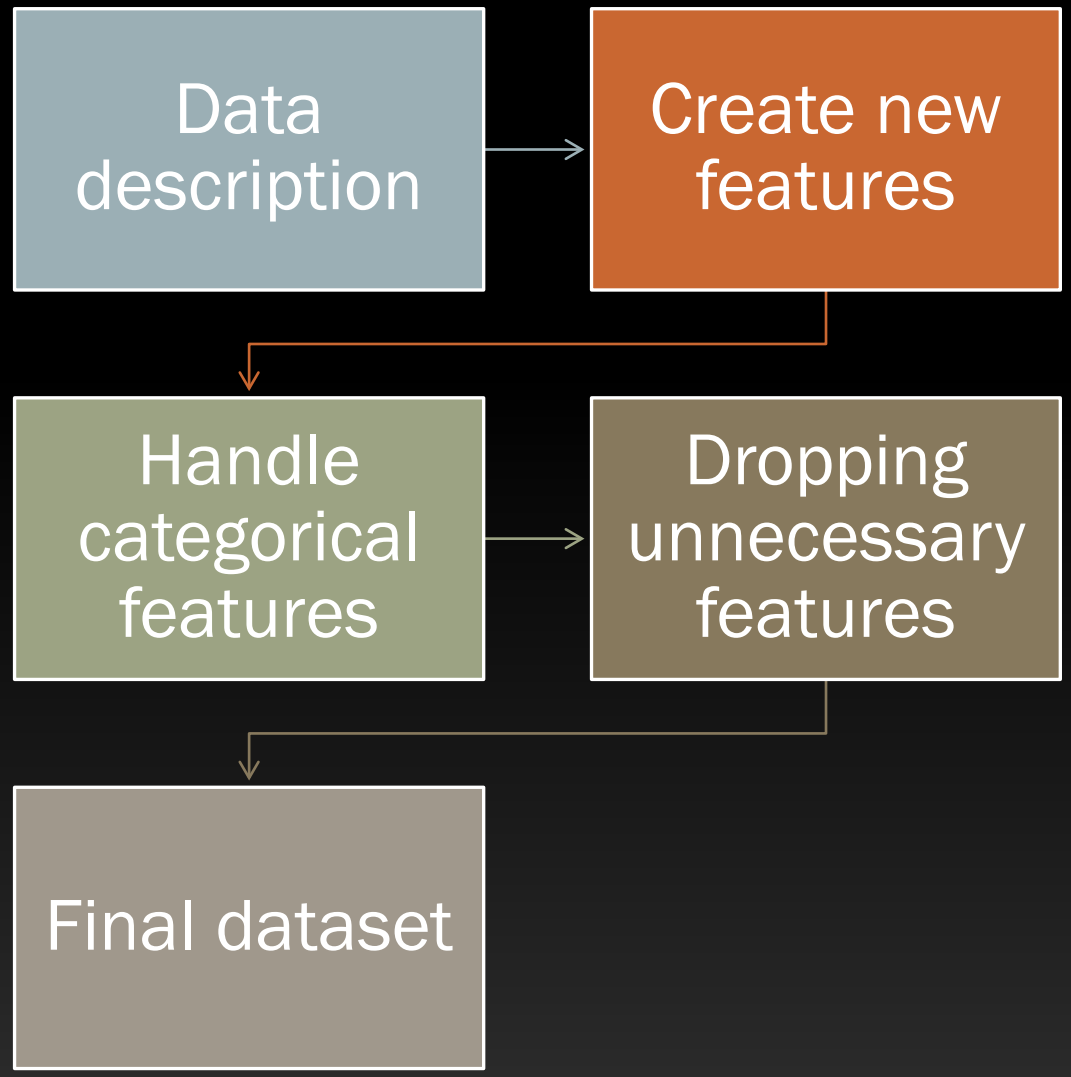

Plotting Spatial Data on Riyadh Map
7.8
7.7
7.6
7.5
7.4
−122.4
−122.3
−122.2
−122.1
−122.0
−121.9
longitude





Correlations matrix

Preprocessing



Create new features

Calculate the distance between two locations based on **Haversine formula.**

$$d = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

Calculate average speed

$$v = \frac{\Delta x}{\Delta t} = \frac{x_2 - x_1}{t_2 - t_1}$$

Calculate altitude

$$E = \arctan\left[\frac{\cos(G)\cos(L) - .1512}{\sqrt{1 - \cos^2(G)\cos^2(L)}}\right]$$

E = elevation of antenna in degrees
S = satellite longitude in degrees
N = site longitude in degrees
L = site latitude in degrees
G = S – N

Handle categorical Features

```
 >>> Data info:
<class 'pandas.core.frame.DataFrame'>
Int64Index: 182429 entries, 4 to 183411
Data columns (total 26 columns):
 #   Column                        Non-Null Count   Dtype
---  ------                        --------------   -----
 0   duration_sec                  182429 non-null  int64
 1   start_time                    182429 non-null  object
 2   end_time                      182429 non-null  object
 3   start_station_id              182429 non-null  float64
 4   start_station_name            182429 non-null  object
 5   start_station_latitude        182429 non-null  float64
 6   start_station_longitude       182429 non-null  float64
 7   end_station_id                182429 non-null  float64
 8   end_station_name              182429 non-null  object
 9   end_station_latitude          182429 non-null  float64
 10  end_station_longitude         182429 non-null  float64
 11  bike_id                       182429 non-null  int64
 12  user_type                     182429 non-null  category
 13  member_birth_year             182429 non-null  float64
 14  member_gender                 182429 non-null  category
 15  bike_share_for_all_trip       182429 non-null  category
 16  age                           182429 non-null  float64
 17  duration_min                  182429 non-null  float64
 18  start_station_latitude_radians  182429 non-null  float64
 19  start_station_longitude_radians  182429 non-null  float64
 20  end_station_latitude_radians  182429 non-null  float64
 21  end_station_longitude_radians  182429 non-null  float64
 22  latitude_diff                 182429 non-null  float64
 23  longitude_diff                182429 non-null  float64
 24  trip_distance                 182429 non-null  float64
 25  avg_trip_speed                182429 non-null  float64
dtypes: category(3), float64(17), int64(2), object(4)
memory usage: 38.0+ MB
None

-------------****---------------
```

Dropping unnecessary features & Order columns

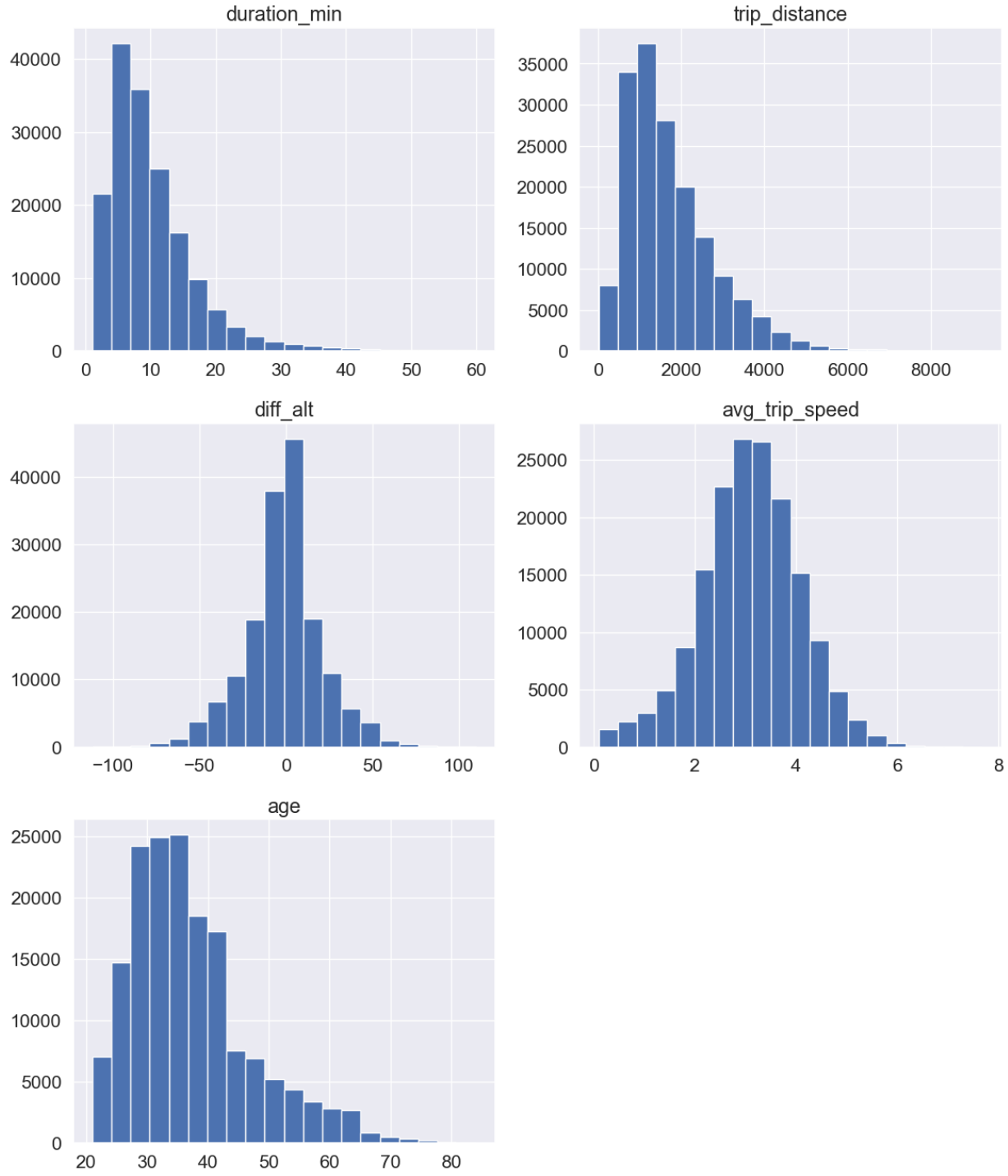
```
order_columns = [
    'bike_id',
    'start_station_id',
    'start_station_name',
    'end_station_id',
    'end_station_name',
    'trip_distance',
    'diff_alt',
    'duration_min',
    'avg_trip_speed',
    'counts_start',
    'counts_end',
    'user_type',
    'member_gender',
    'age',
    'bike_share_for_all_trip'
    ]

fdf = fdf[order_columns]
fdf.to_csv(r'fdf.csv', index=False)
fdf.columns
```

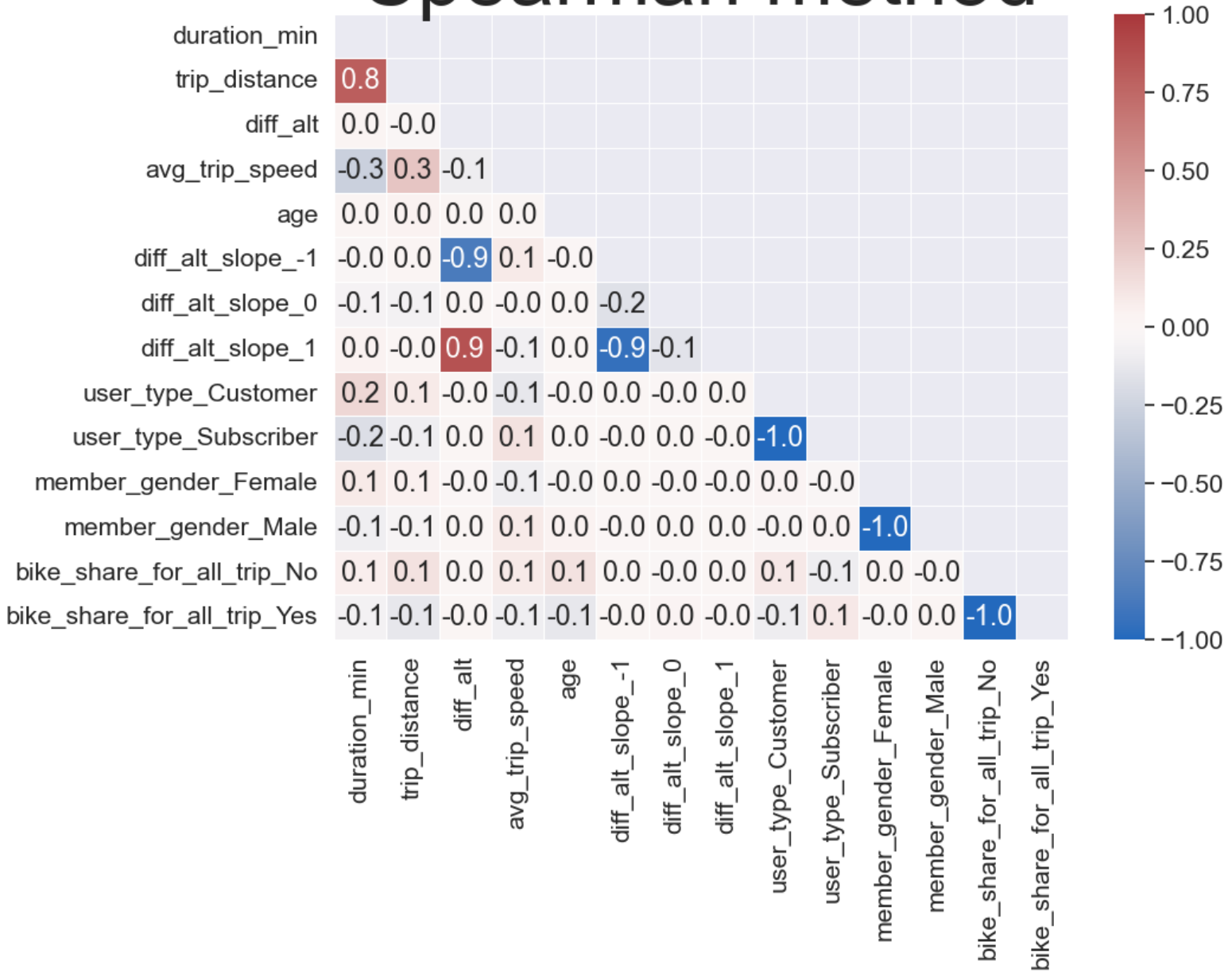
```
drop_columns_list  = [
        'duration_sec',
        'start_time',
        'end_time',
          # 'start_station_id',
          # 'start_station_name',
        'start_station_latitude',
        'start_station_longitude',
          # 'end_station_id',
          # 'end_station_name',
        'end_station_latitude',
        'end_station_longitude',
          # 'bike_id',
          # 'user_type',
        'member_birth_year',
          # 'member_gender',
          # 'bike_share_for_all_trip',
          # 'age',
          # 'duration_min',
        'start_station_latitude_radians',
        'start_station_longitude_radians',
        'end_station_latitude_radians',
        'end_station_longitude_radians',
        'latitude_diff',
        'longitude_diff',
          # 'trip_distance',
          # 'avg_trip_speed',
        'station_latitude_start',
        'station_longitude_start',
          # 'counts_start',
        'station_altitude_start',
        'station_latitude_end',
        'station_longitude_end',
        # ' counts_end',
        'station_altitude_end',
          # 'diff_alt'
      ]
```

```
[115]  cdf = pdf.drop(drop_columns_list, axis=1)
       cdf
```

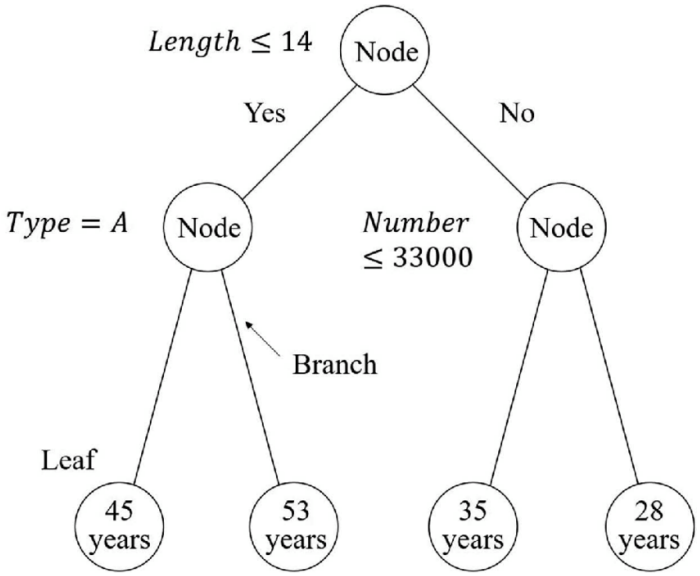
Features Distributions

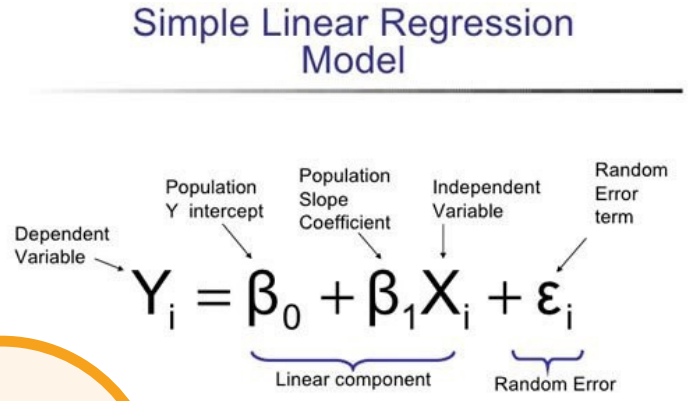


Spearman method



Regression models





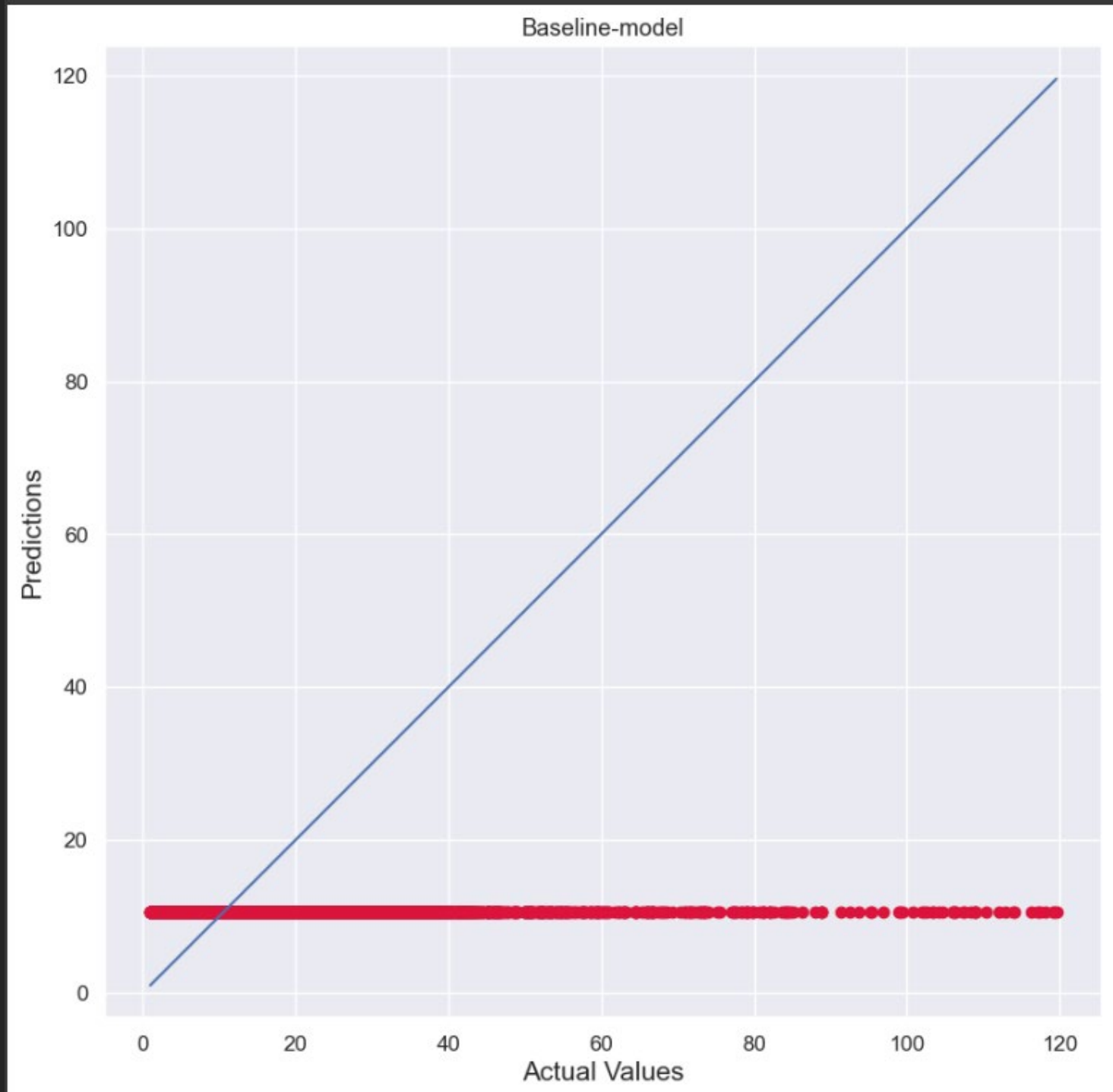
Train: 70%

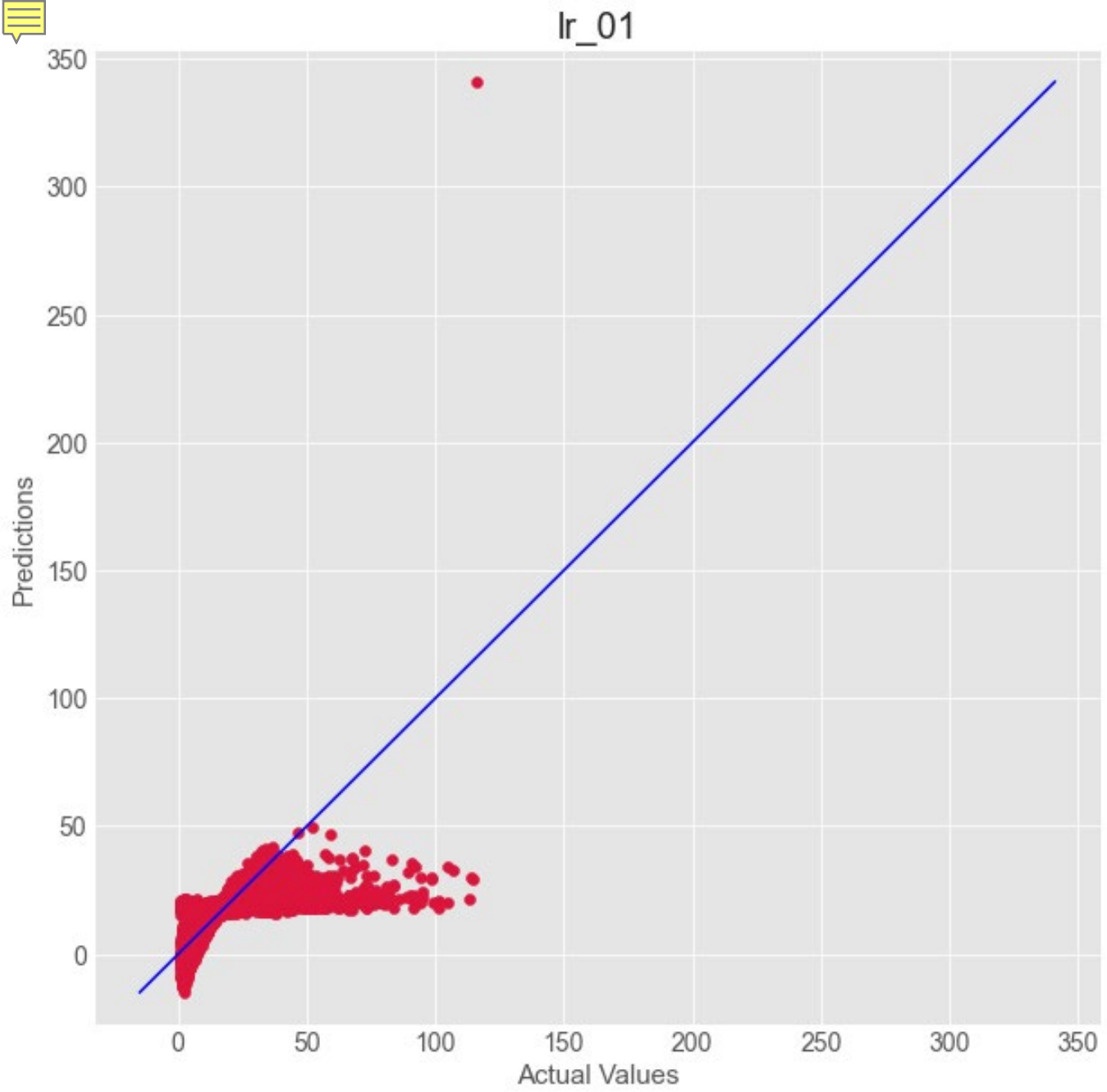
Valid: 15%

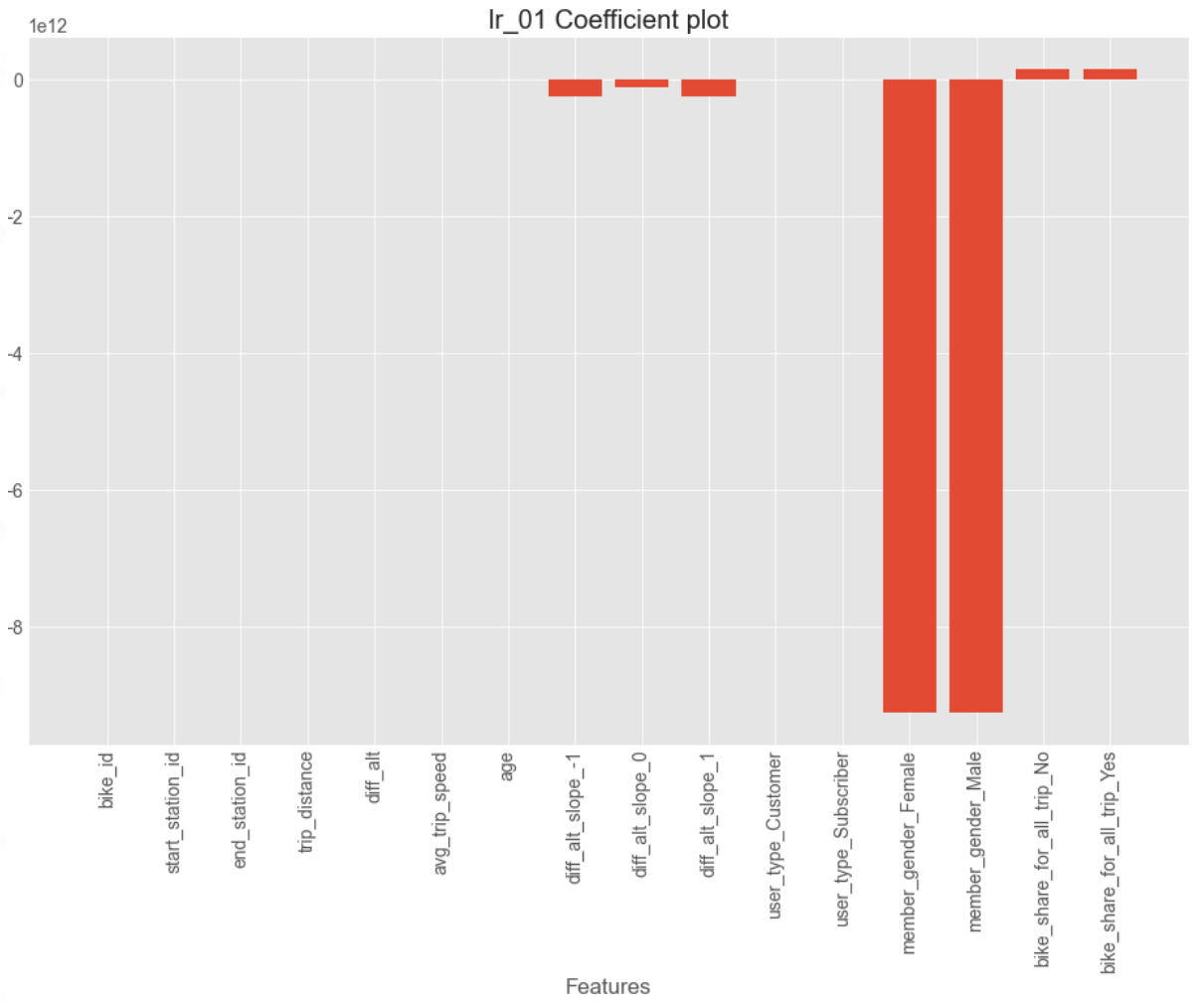
Test: 15%

Baseline model based on .mean()

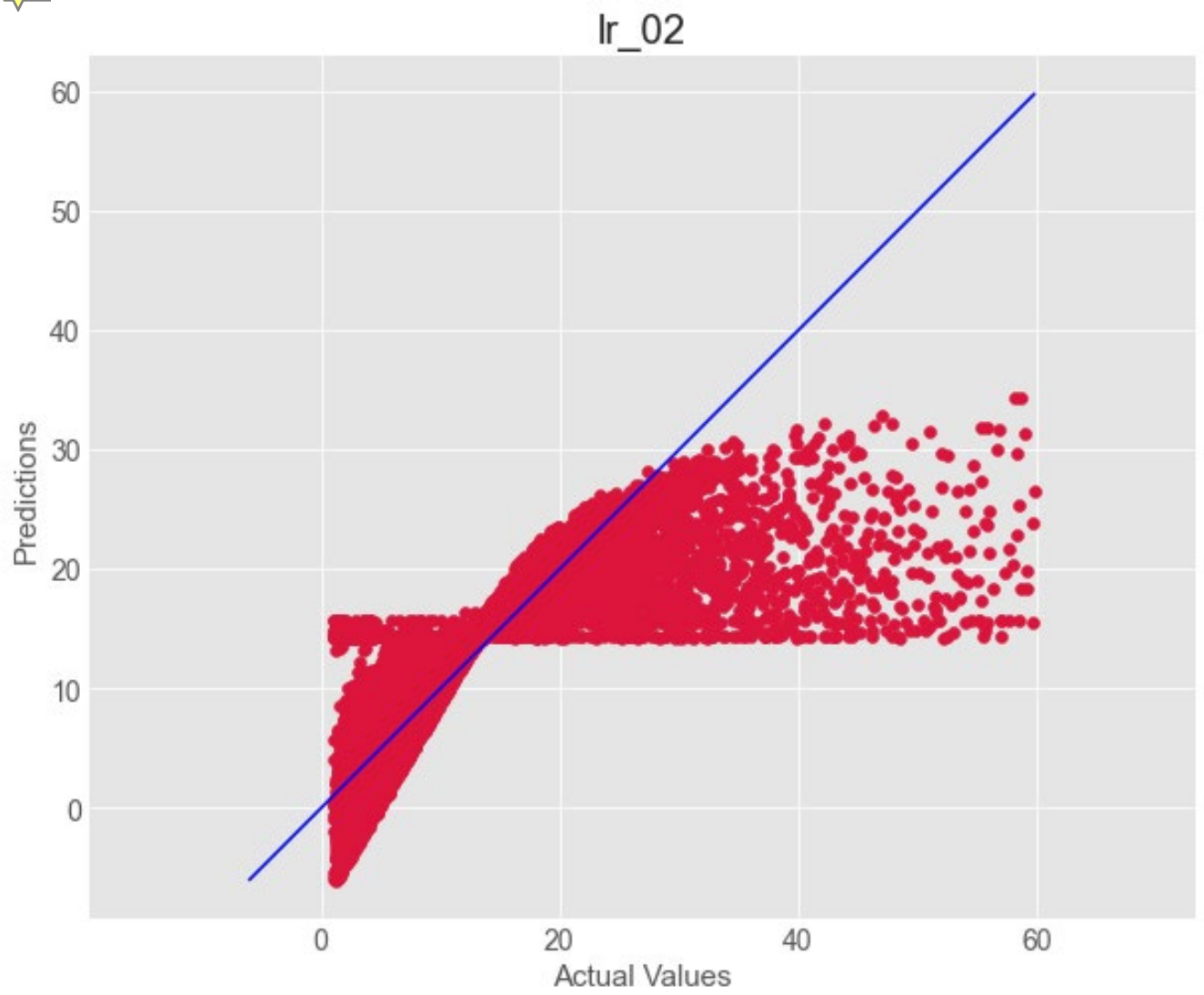
```
MAE: 5.654298977198291
MSE: 82.18859531479465
RMSE: 9.06579259164882
R2: 0
```

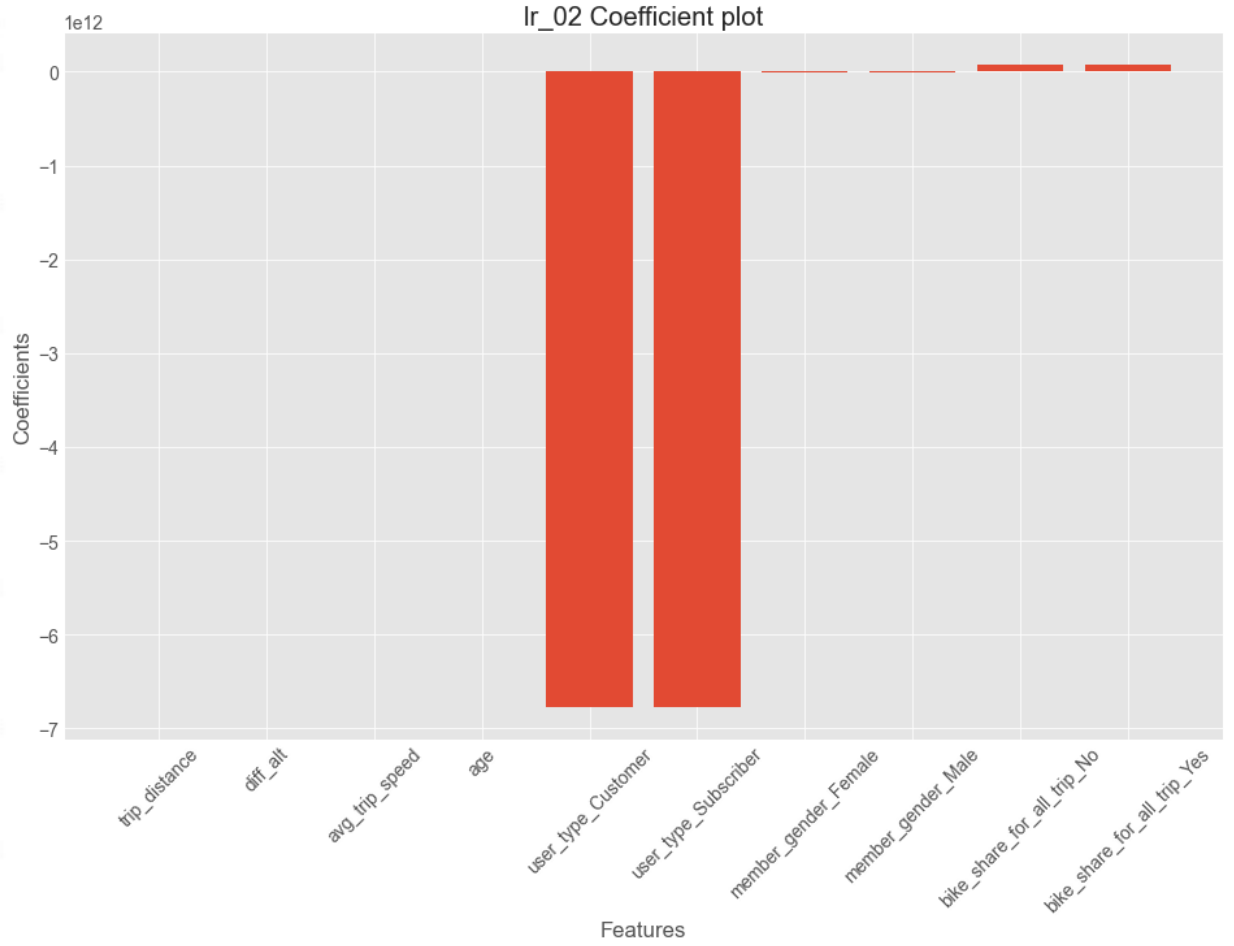




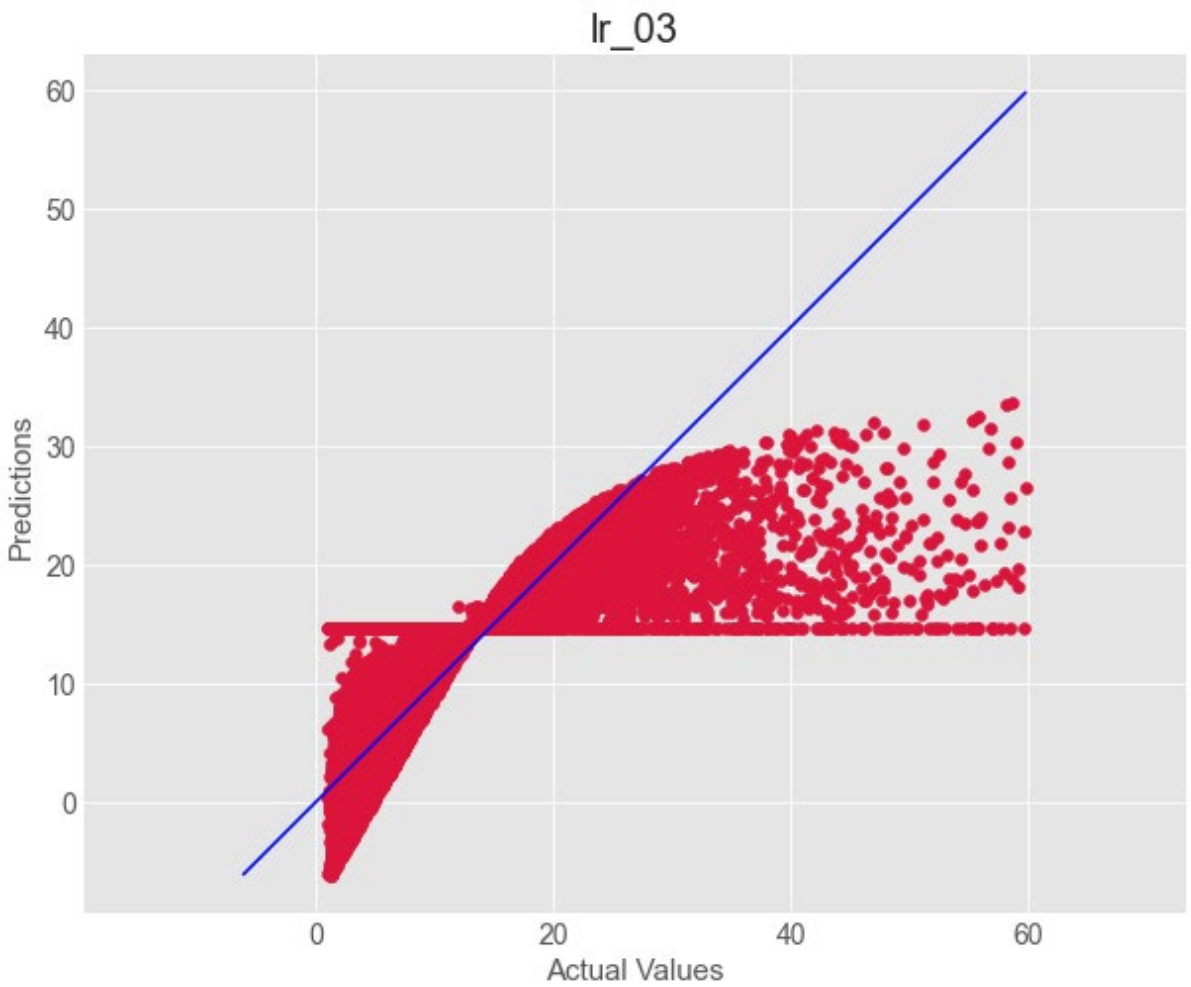


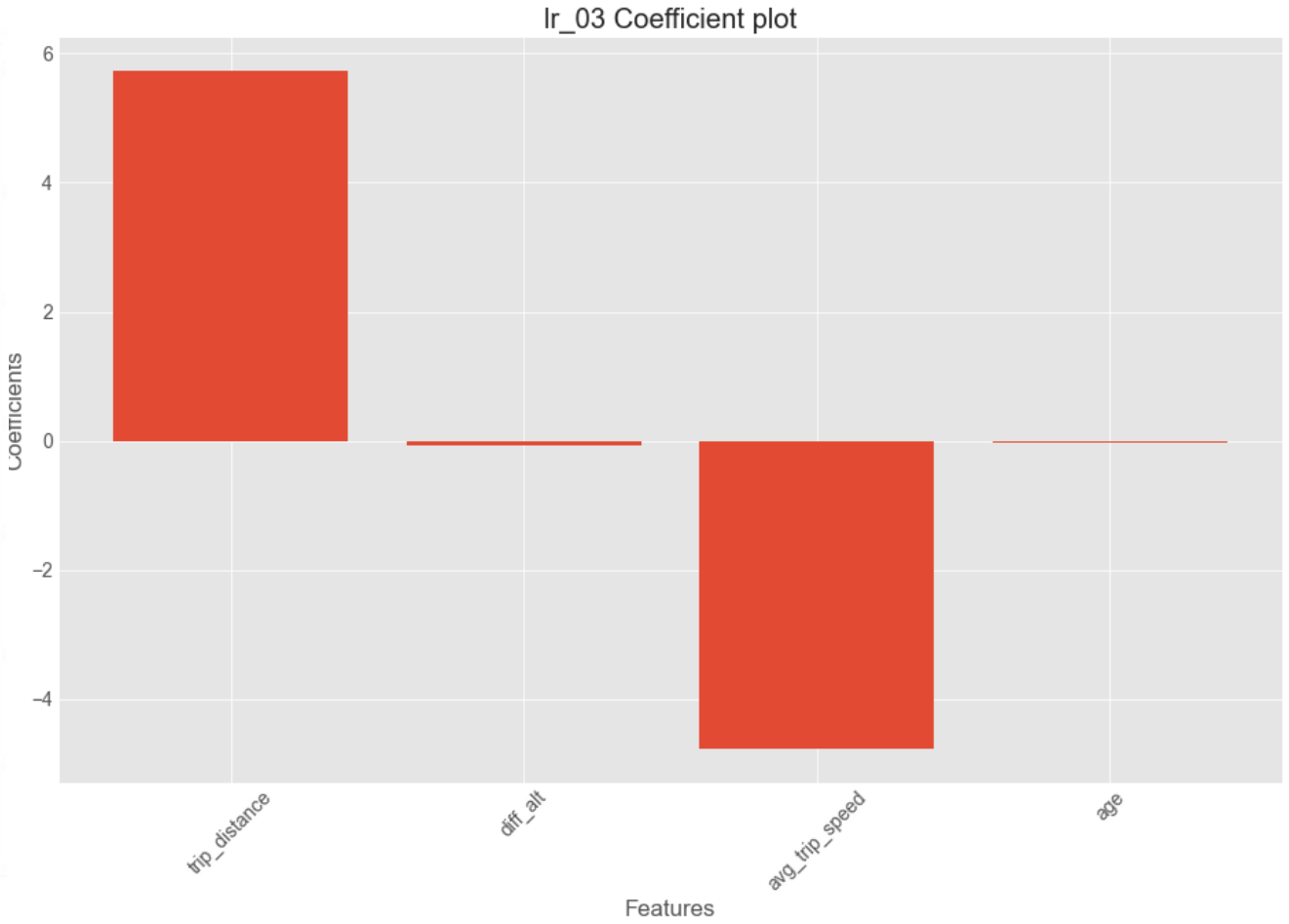
$MAE = 2.70,$ $MSE = 31.59,$ $RMSE = 5.62,$ $R^2 = 0.573$



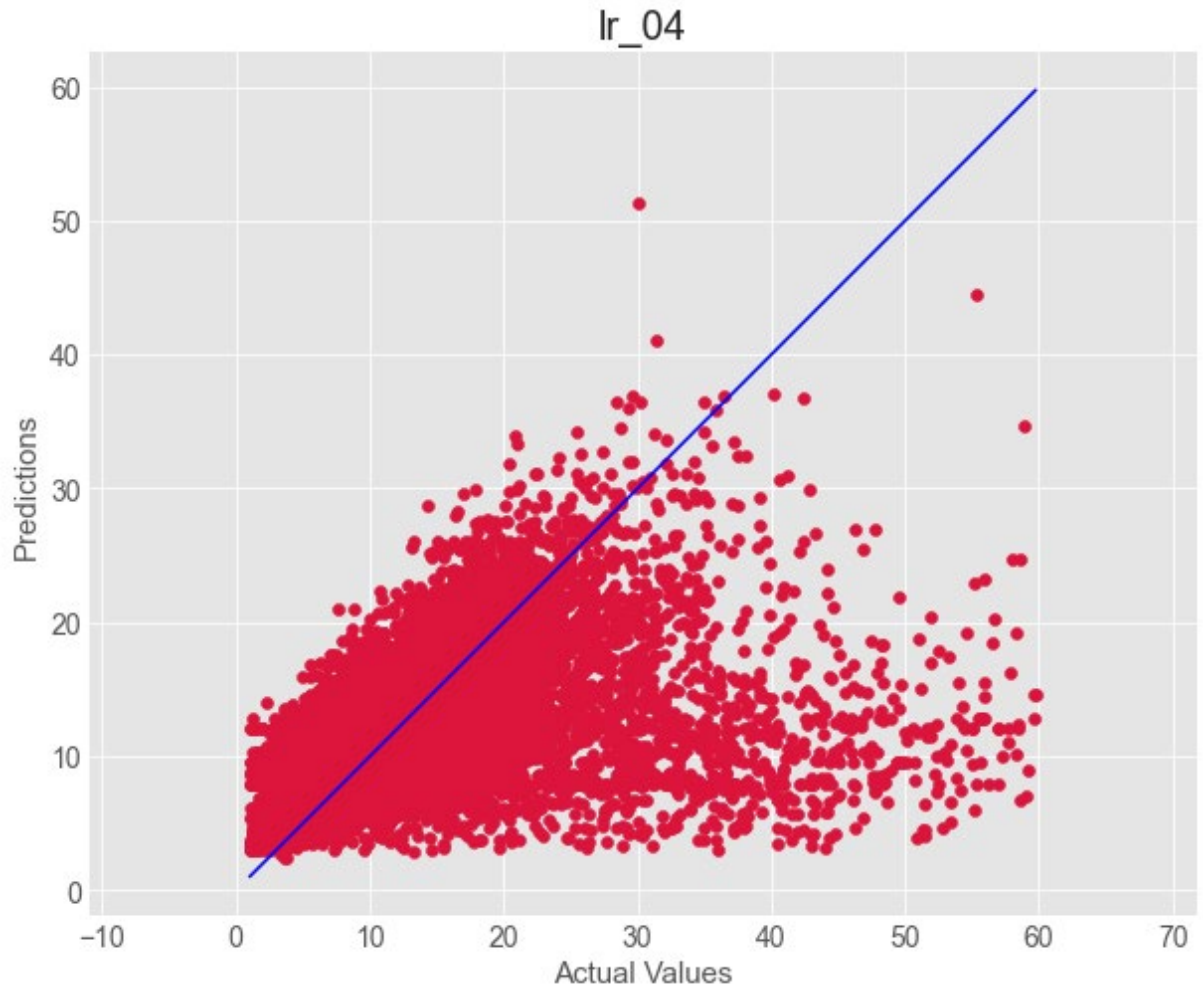


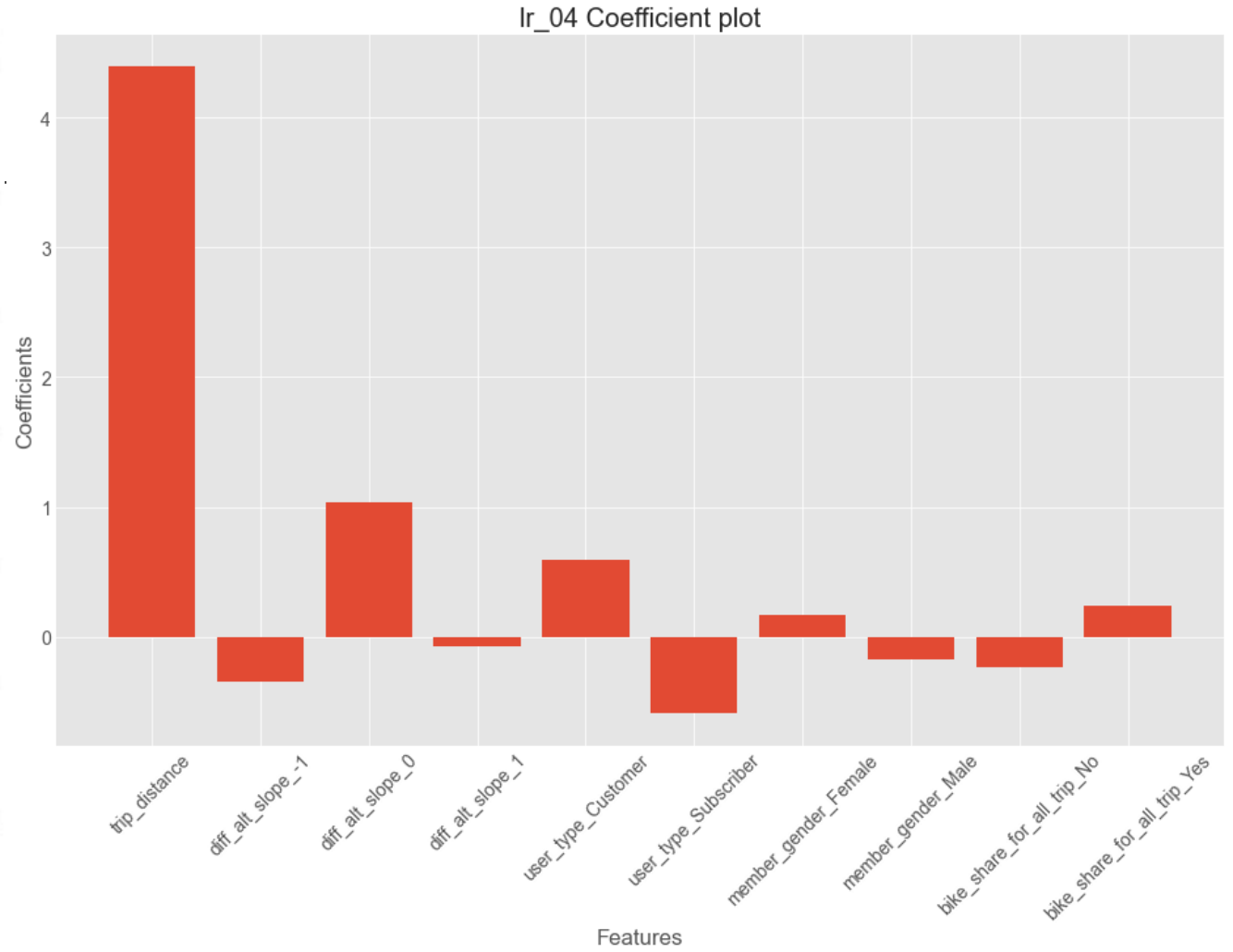
$MAE = 2.27,$ $MSE = 17.19,$ $RMSE = 4.14,$ $R^2 = 0.69$





$Validation$: $MAE = 2.890$, $MSE = 17.381$, $RMSE = 4.169$, $R^2 = 0.688$
$Test$: $MAE = 2.256$, $MSE = 16.877$, $RMSE = 4.108$, $R^2 = 0.691$





$Validation$: $MAE = 3.295$, $MSE = 34.803$, $RMSE = 5.899$, $R^2 = 0.377$
$Test$: $MAE = 3.266$, $MSE = 33.864$, $RMSE = 5.819$, $R^2 = 0.381$

Decision-Tree Modeling

This phase split up into two parts:

1. A dataset with clipping into the limits (clip_df)

2. A dataset without unruly data (cut_df)

Both were examined parallel with the same model attributes, same hyper-parameters and so forth.

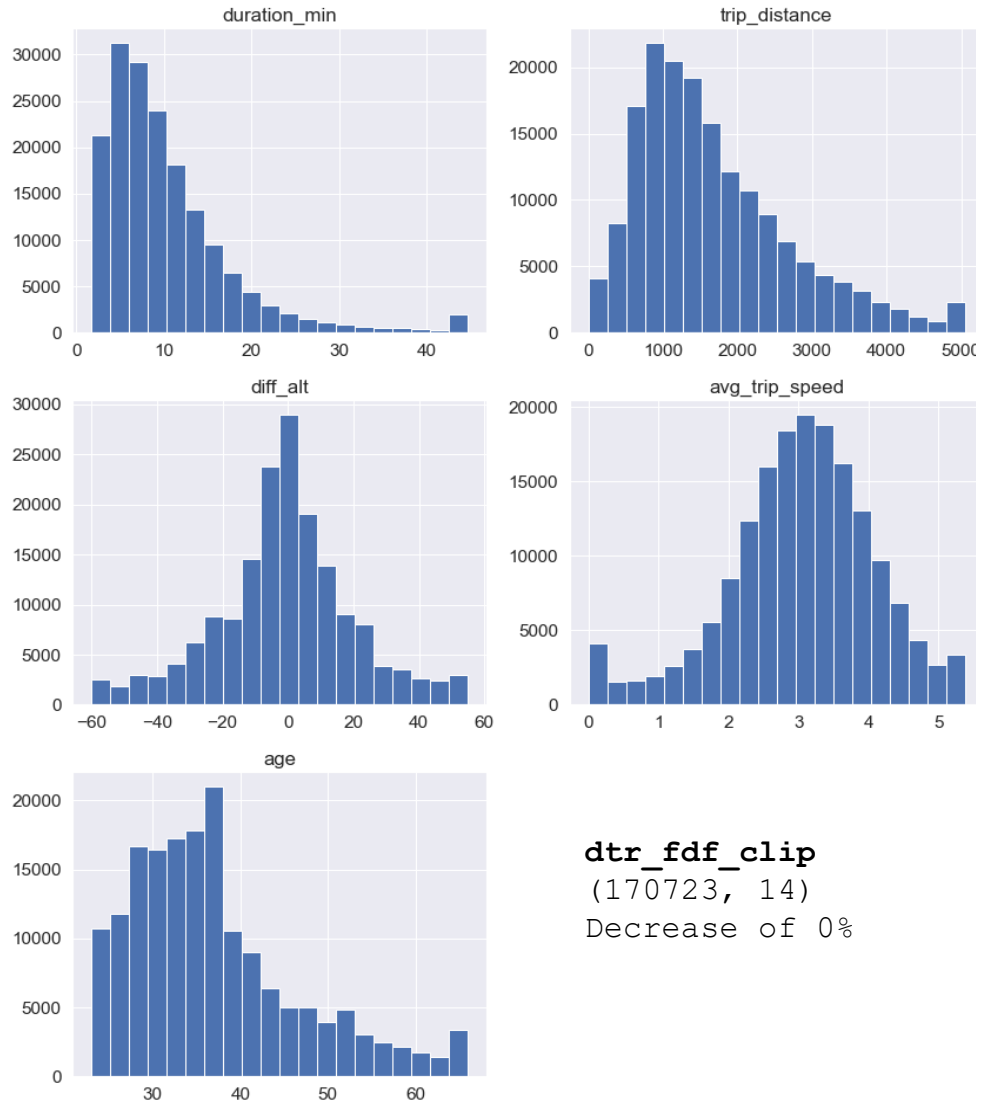
Methods we had used:

Handle with categorical features via `$get.dummies()$`

Control the attributes of decision regressor: [`$random\_state$`,`$tree\_.n\_leaves$`]

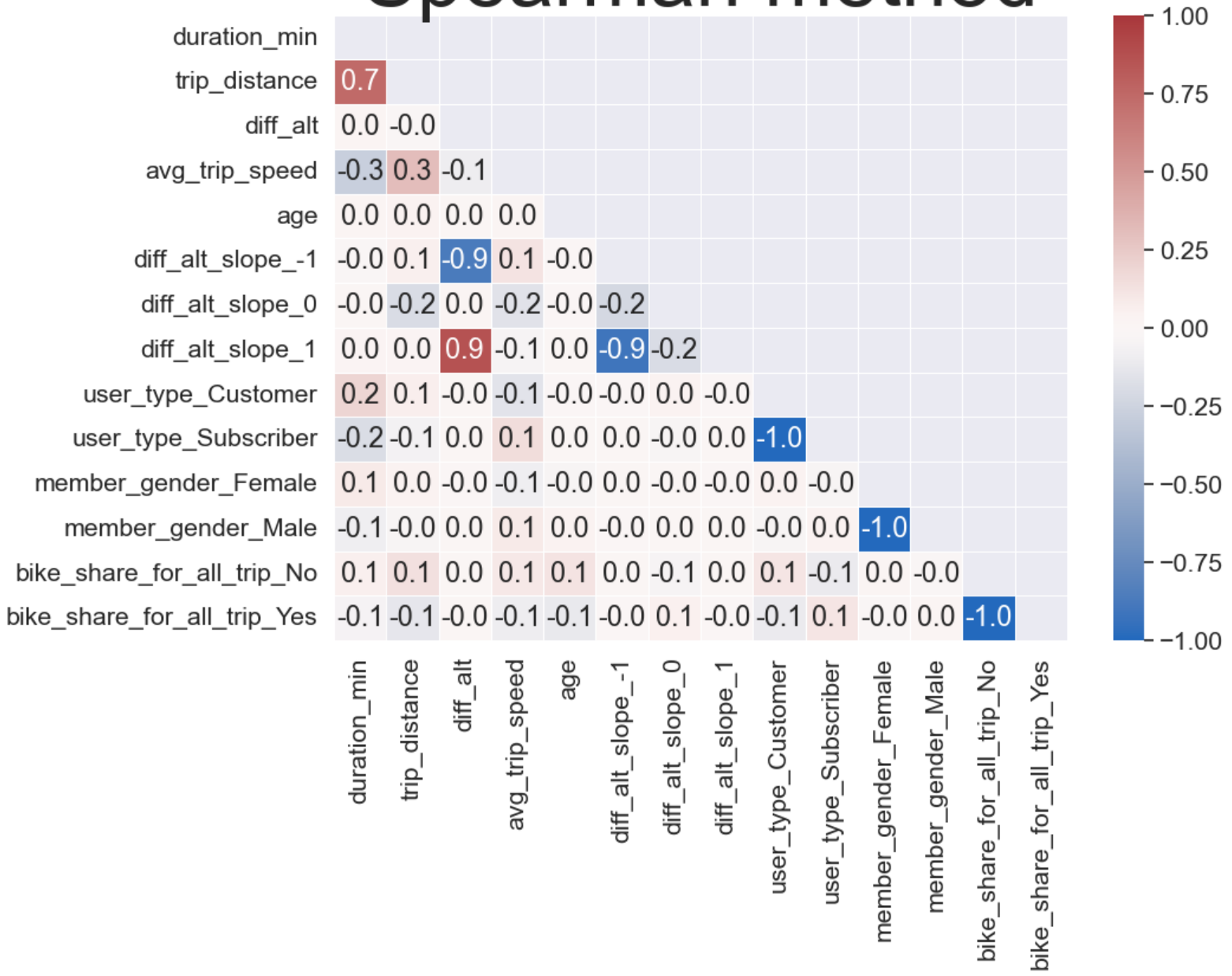
Tuning Hyper-parameters: [`max_depth`, `min_samples_split`, `min_samples_leaf`, `ccp_alpha`]

Clip_df



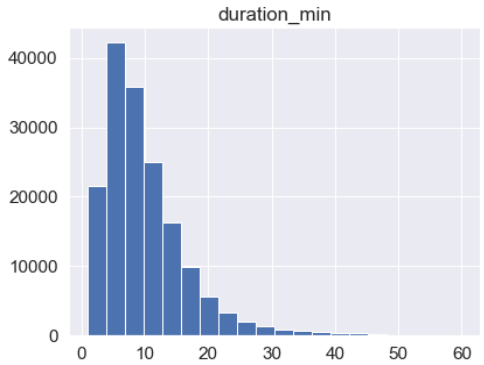
dtr_fdf_clip
(170723, 14)
Decrease of 0%

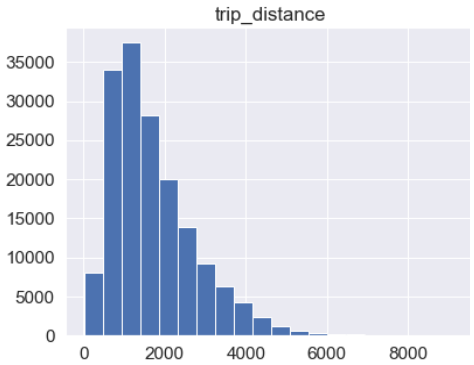
Spearman method

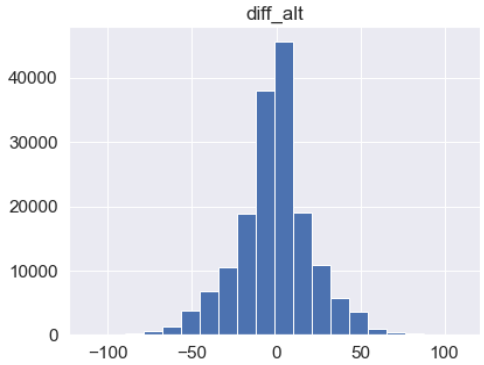


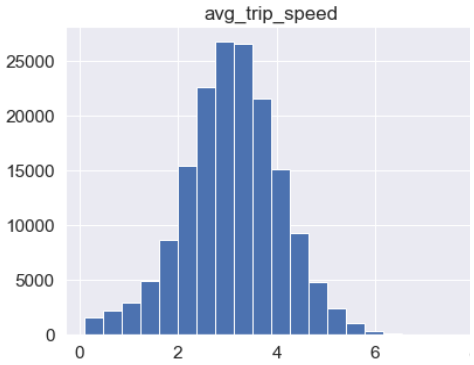
clip_train(119506, 14), *clip_valid* (25608 , 14), *clip_test*(25609 , 14)

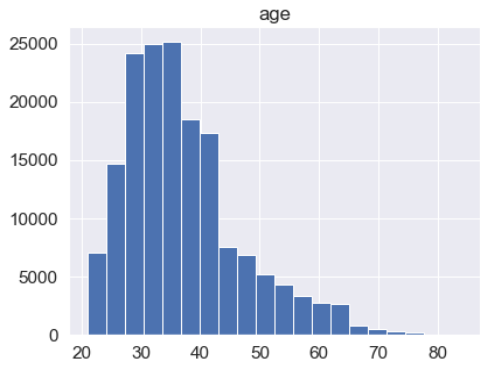
Cut_df





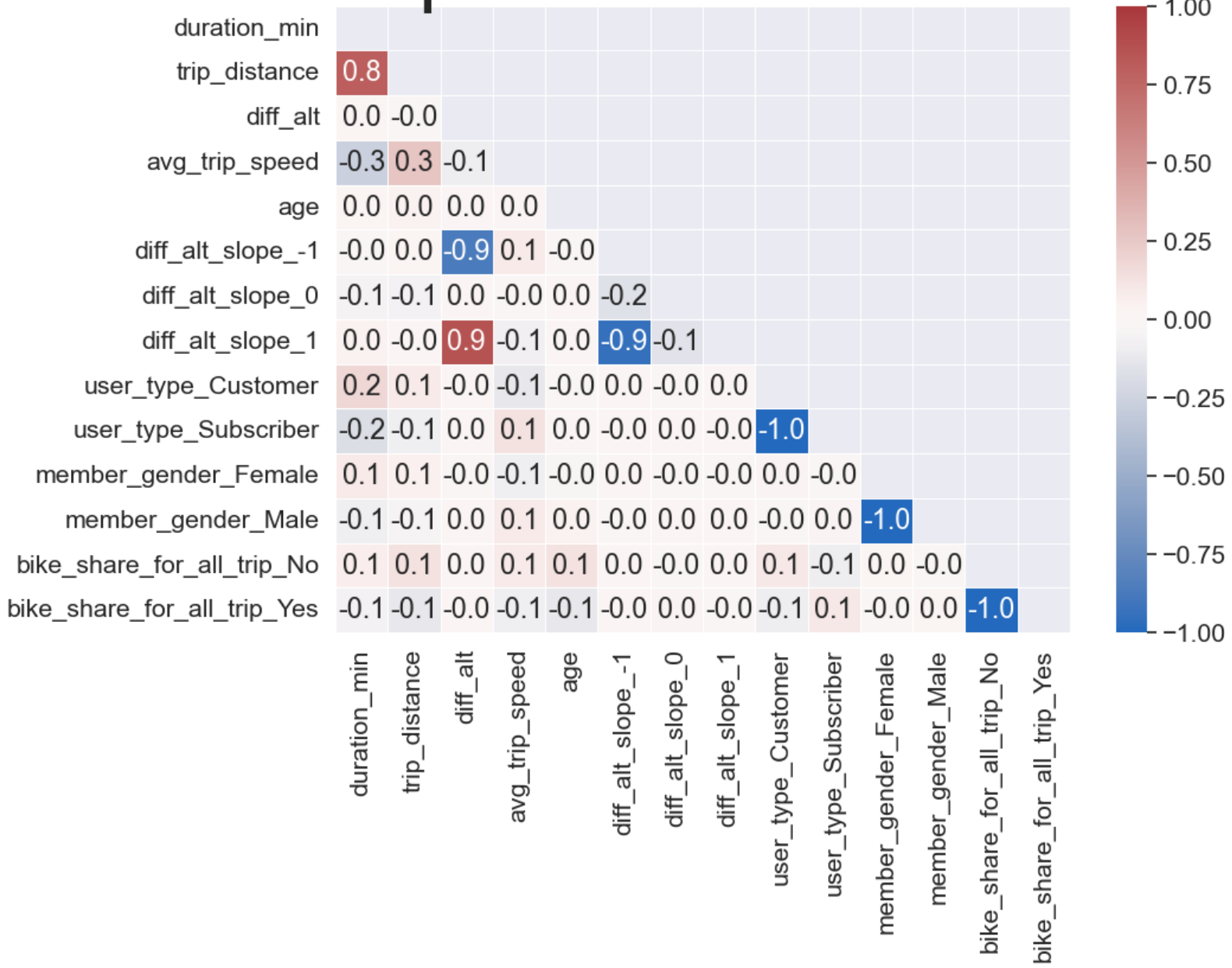






dtr_fdf_cut
(166679, 14)
Decrease of 2.37%

Spearman method

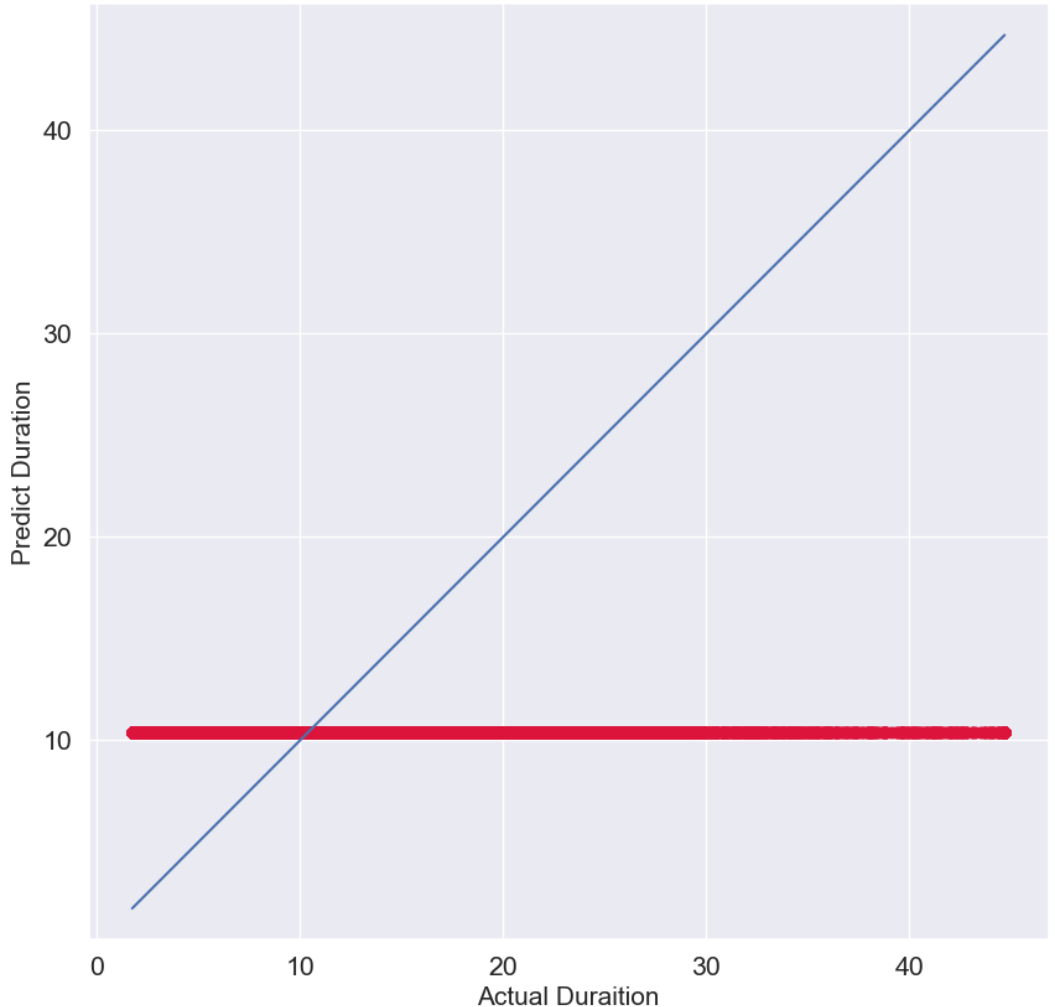


$cut\_train(116675, 14)$, $cut\_valid\ (25002\ , 14)$, $cut\_test(25002\ , 14)$

dtr_00
Baseline model based on .mean()

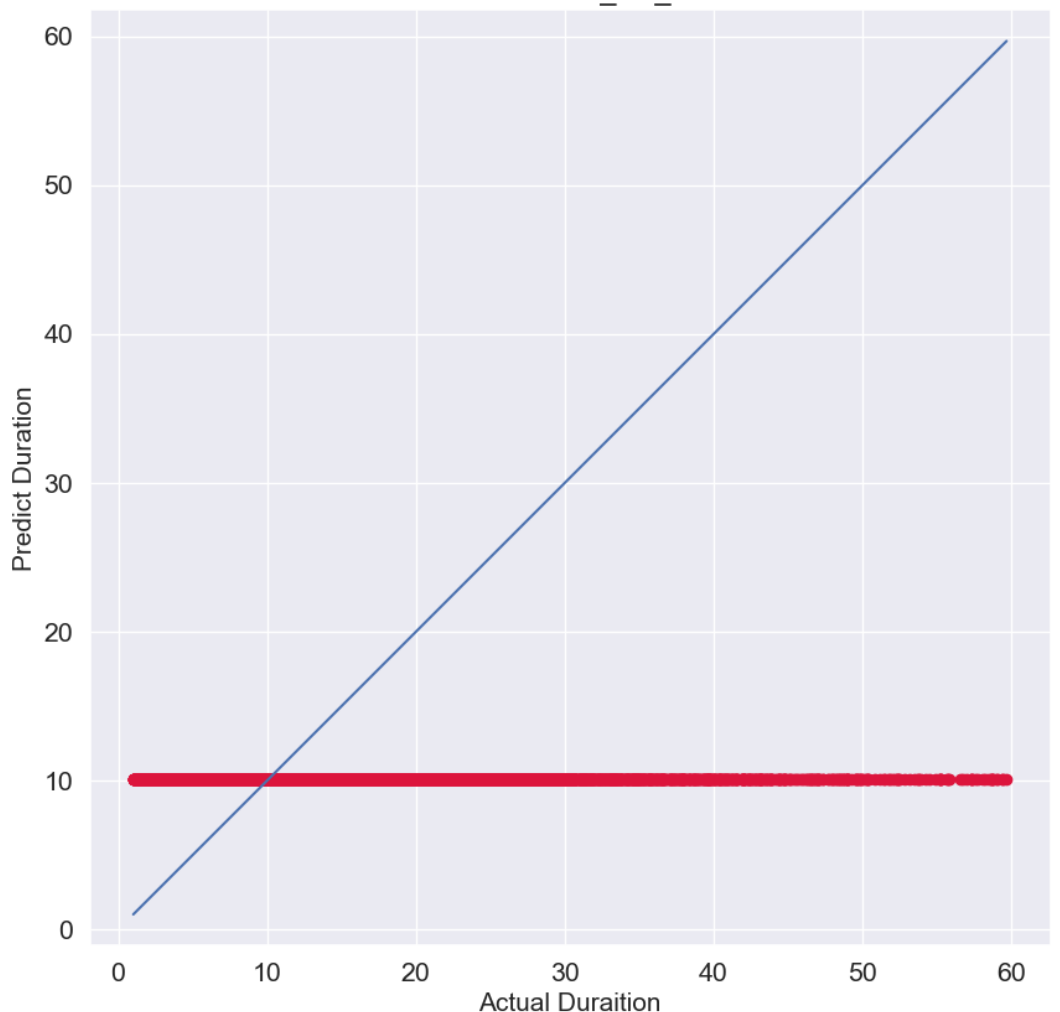
Train:	$MAE = 5.341,$	$MSE = 56.623,$	$RMSE = 7.524,$	$R^2 = 0.0$
Validation:	$MAE = 5.358,$	$MSE = 56.675,$	$RMSE = 7.528,$	$R^2 = 0.0$
Test:	$MAE = 5.366,$	$MSE = 56.893,$	$RMSE = 7.542,$	$R^2 = 0.0$

validation dtr_clip_00



Train:	$MAE = 5.026,$	$MSE = 49.197,$	$RMSE = 7.014,$	$R^2 = 0.0$
Validation:	$MAE = 5.084,$	$MSE = 50.731,$	$RMSE = 7.122,$	$R^2 = 0.0$
Test:	$MAE = 5.052,$	$MSE = 50.804,$	$RMSE = 7.127,$	$R^2 = 0.0$

validation dtr_cut_00



dtr_01

FEATURES:

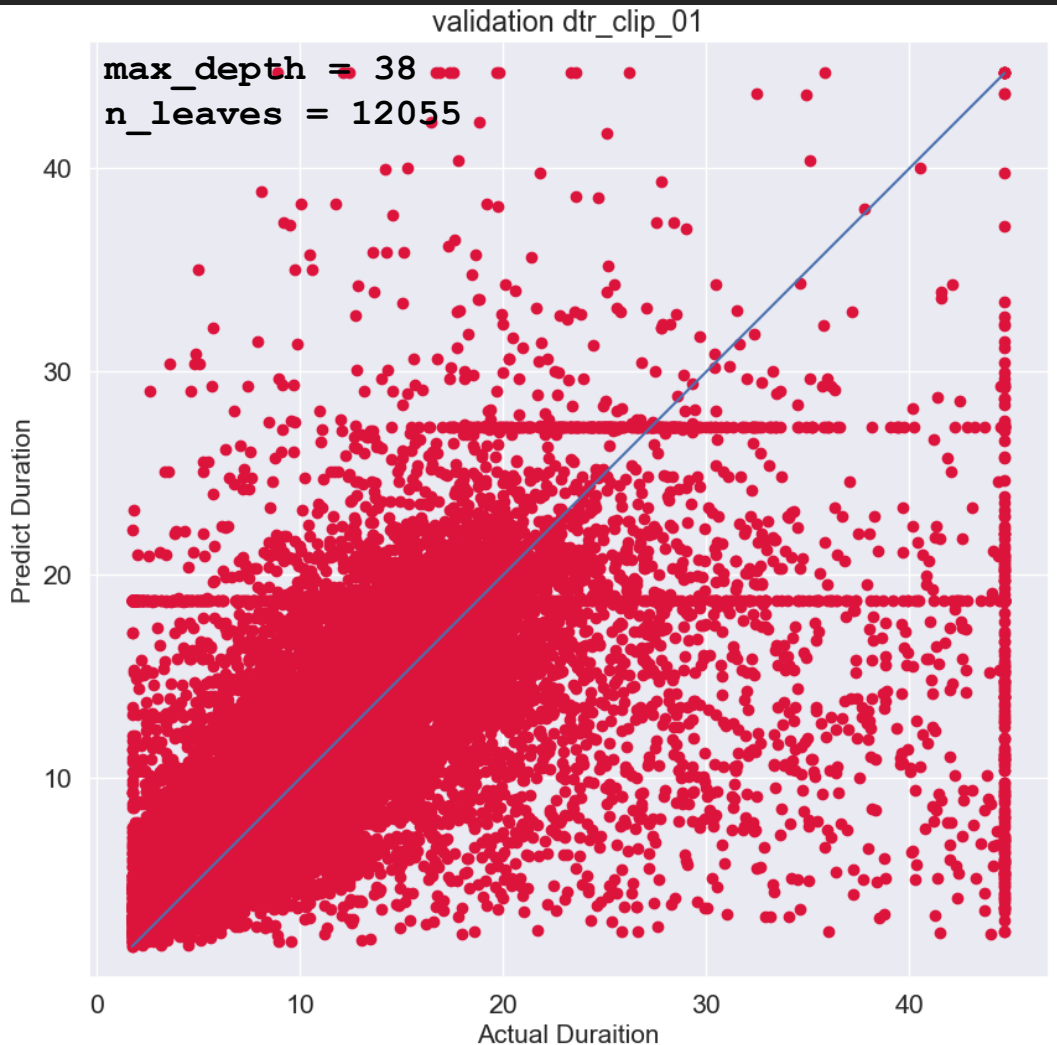
['trip_distance']

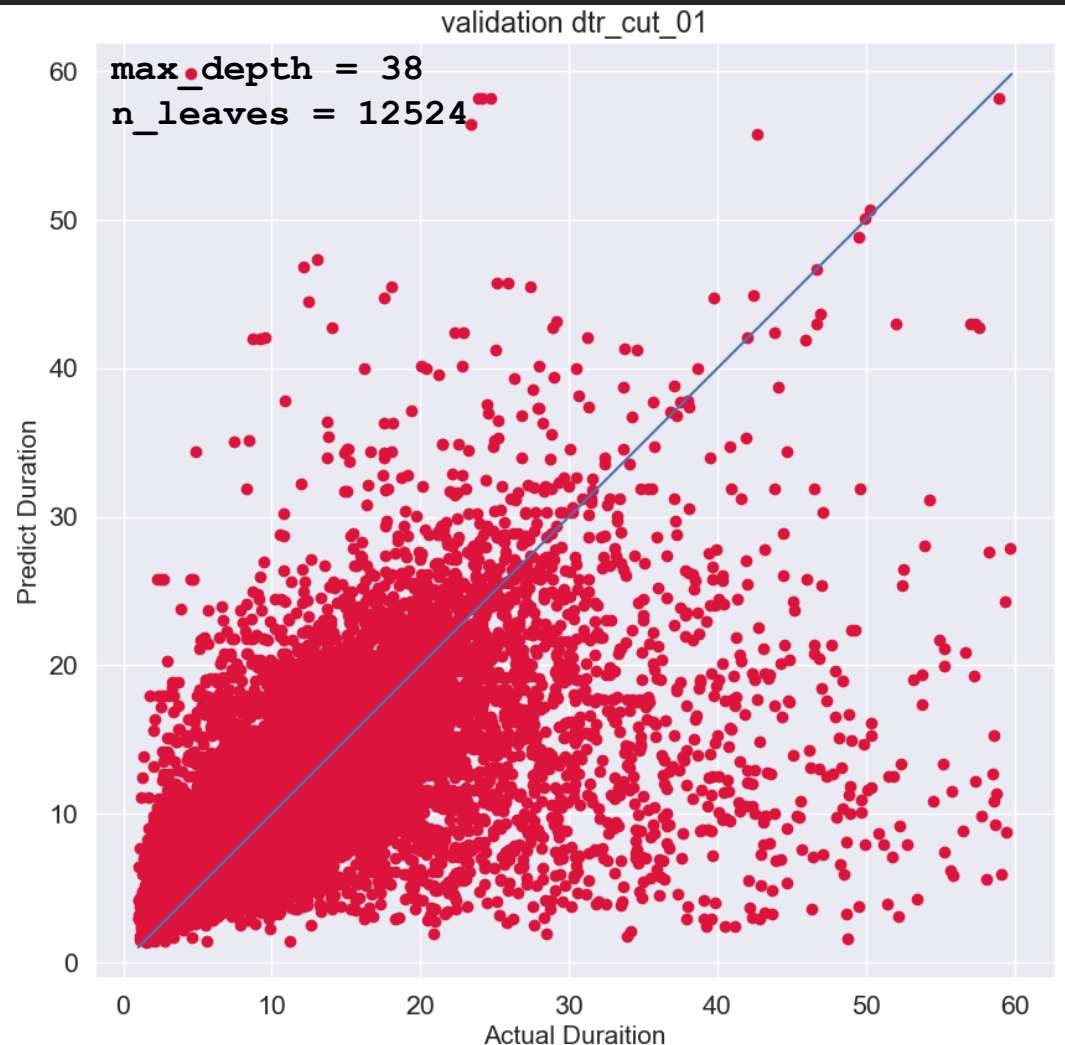
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 2.774$, $MSE = 26.087$, $RMSE = 5.107$, $R^2 = 0.539$
$Validation$: $MAE = 3.179$, $MSE = 33.634$, $RMSE = 7.528$, $R^2 = 0.406$

$Train$: $MAE = 2.405$, $MSE = 20.088$, $RMSE = 4.482$, $R^2 = 0.597$
$Validation$: $MAE = 2.844$, $MSE = 27.532$, $RMSE = 5.247$, $R^2 = 0.457$





dtr_02

FEATURES:

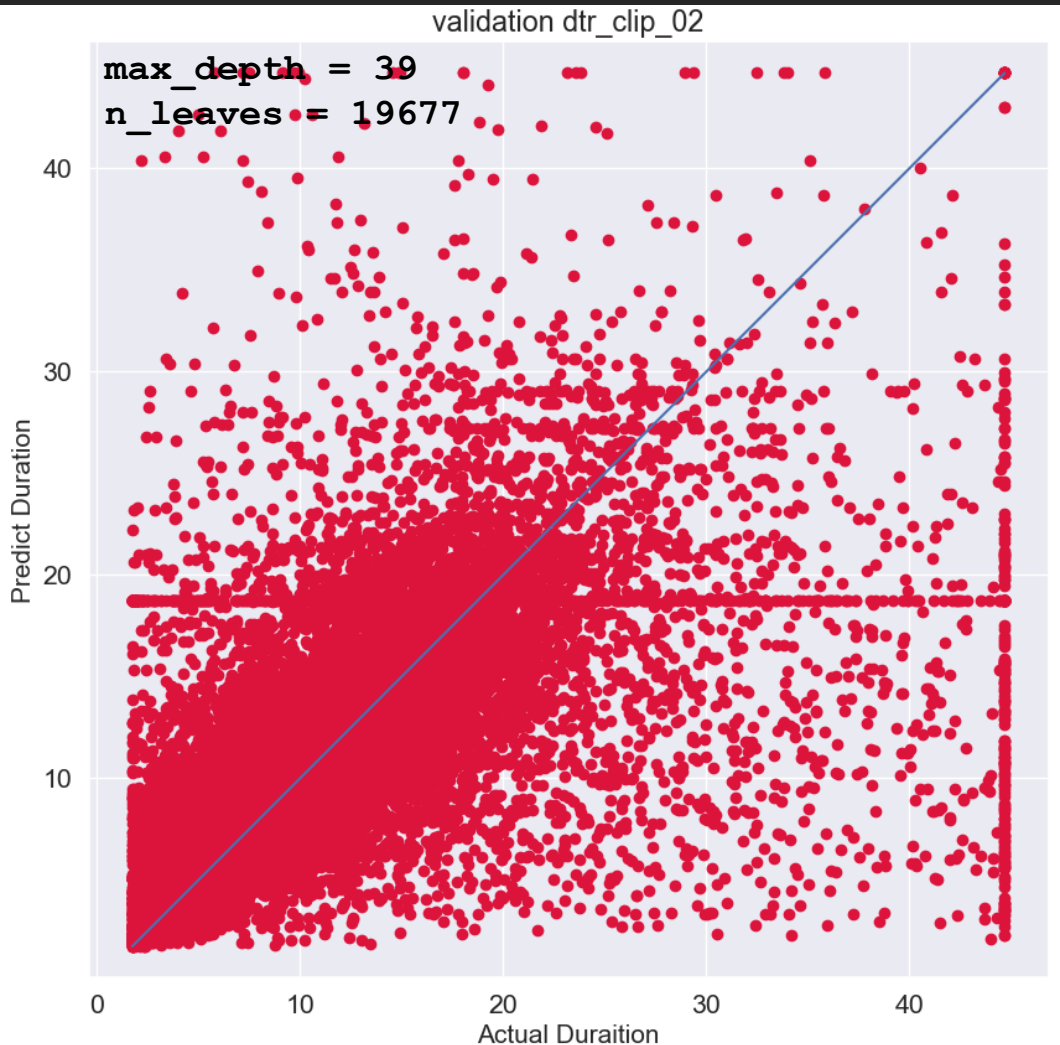
['trip_distance', 'diff_alt']

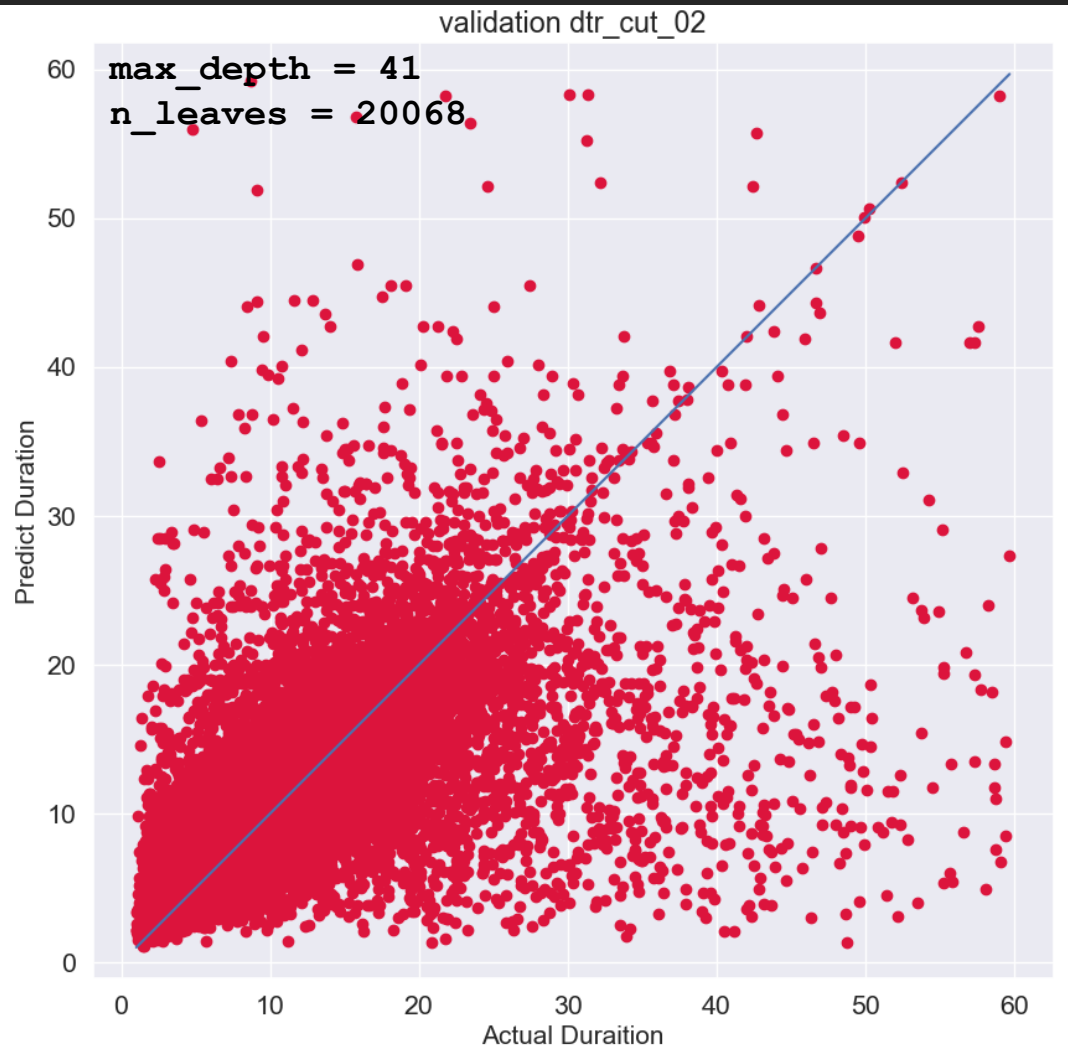
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 2.517$, $MSE = 23.461$, $RMSE = 4.843$, $R^2 = 0.585$
$Validation$: $MAE = 3.205$, $MSE = 35.314$, $RMSE = 5.942$, $R^2 = 0.376$

$Train$: $MAE = 2.177$, $MSE = 17.583$, $RMSE = 4.193$, $R^2 = 0.642$
$Validation$: $MAE = 2.901$, $MSE = 29.413$, $RMSE = 5.423$, $R^2 = 0.421$





dtr_03

FEATURES:

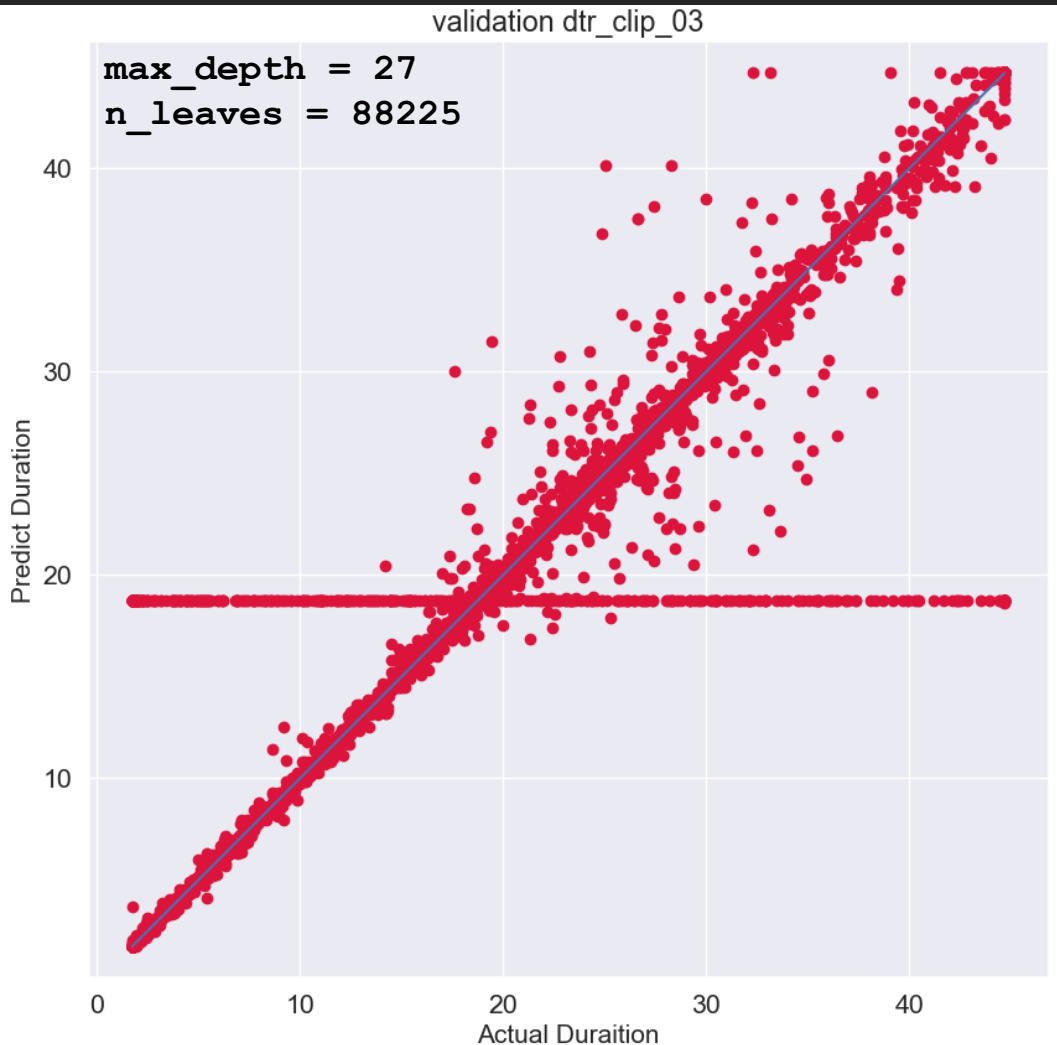
['trip_distance', 'diff_alt', 'avg_trip_speed']

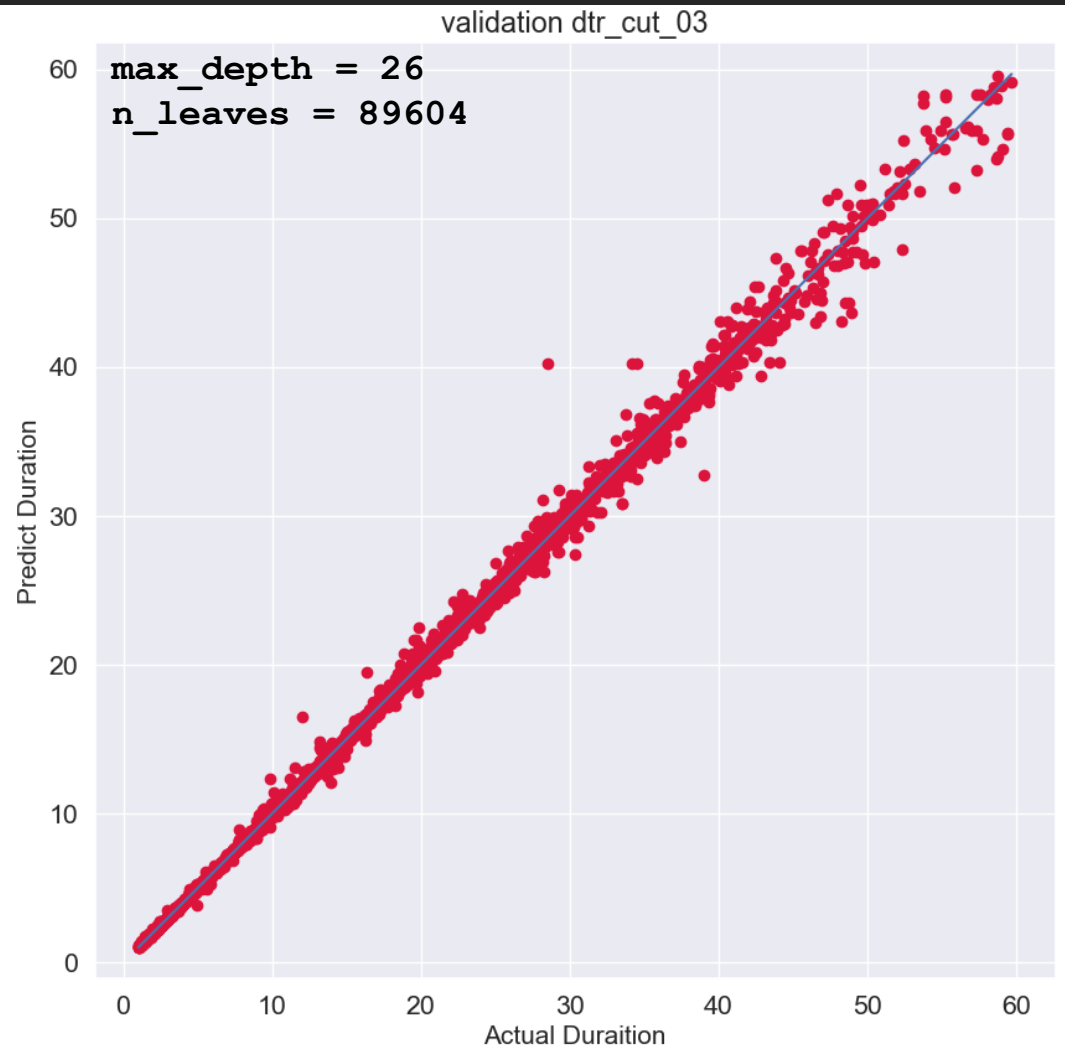
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 0.257$, $MSE = 4.412$, $RMSE = 2.101$, $R^2 = 0.922$
$Validation$: $MAE = 0.325$, $MSE = 4.223$, $RMSE = 2.055$, $R^2 = 0.925$

$Train$: $MAE = 1.387$, $MSE = 1.495$, $RMSE = 1.222$, $R^2 = 1$
$Validation$: $MAE = 0.071$, $MSE = 0.061$, $RMSE = 0.248$, $R^2 = 0.998$





dtr_04

FEATURES:

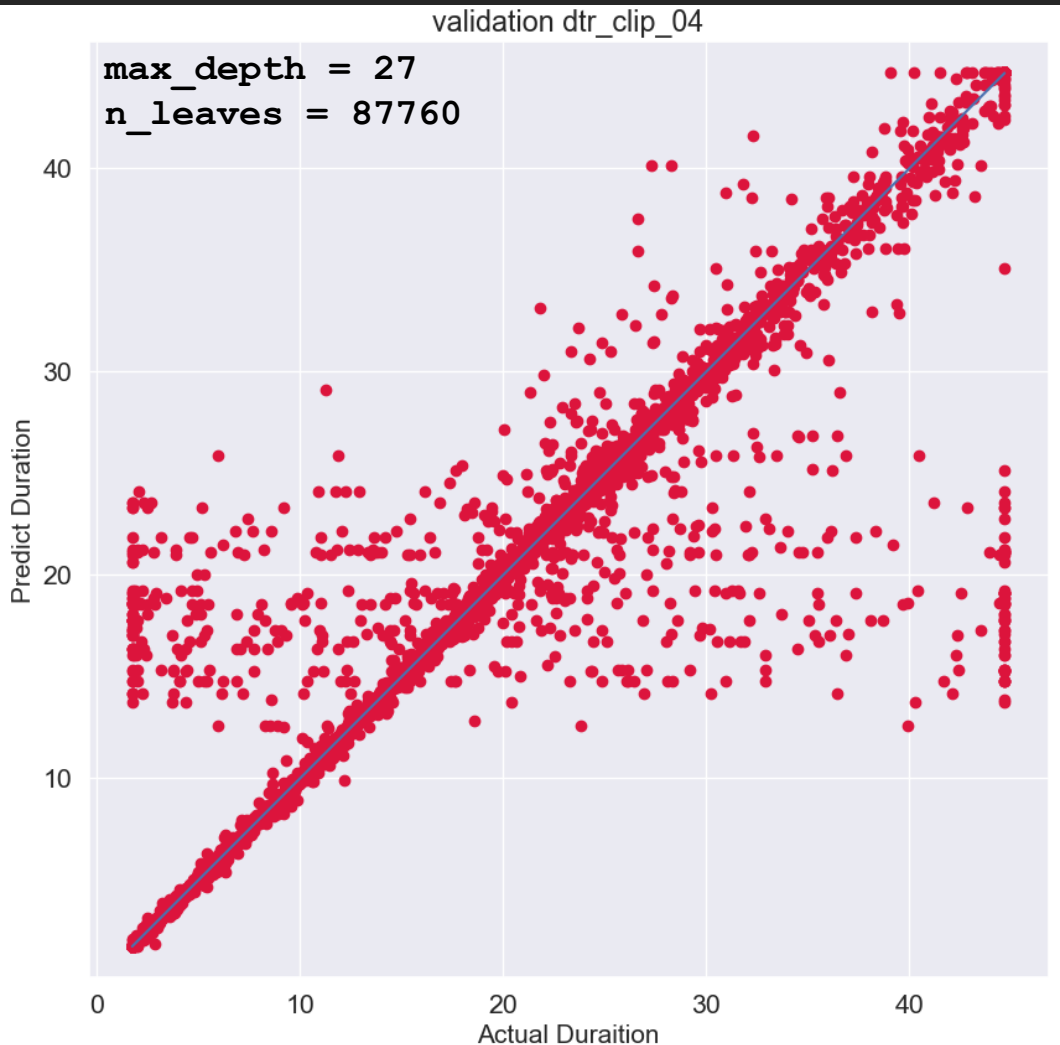
['trip_distance', 'diff_alt', 'avg_trip_speed', 'age']

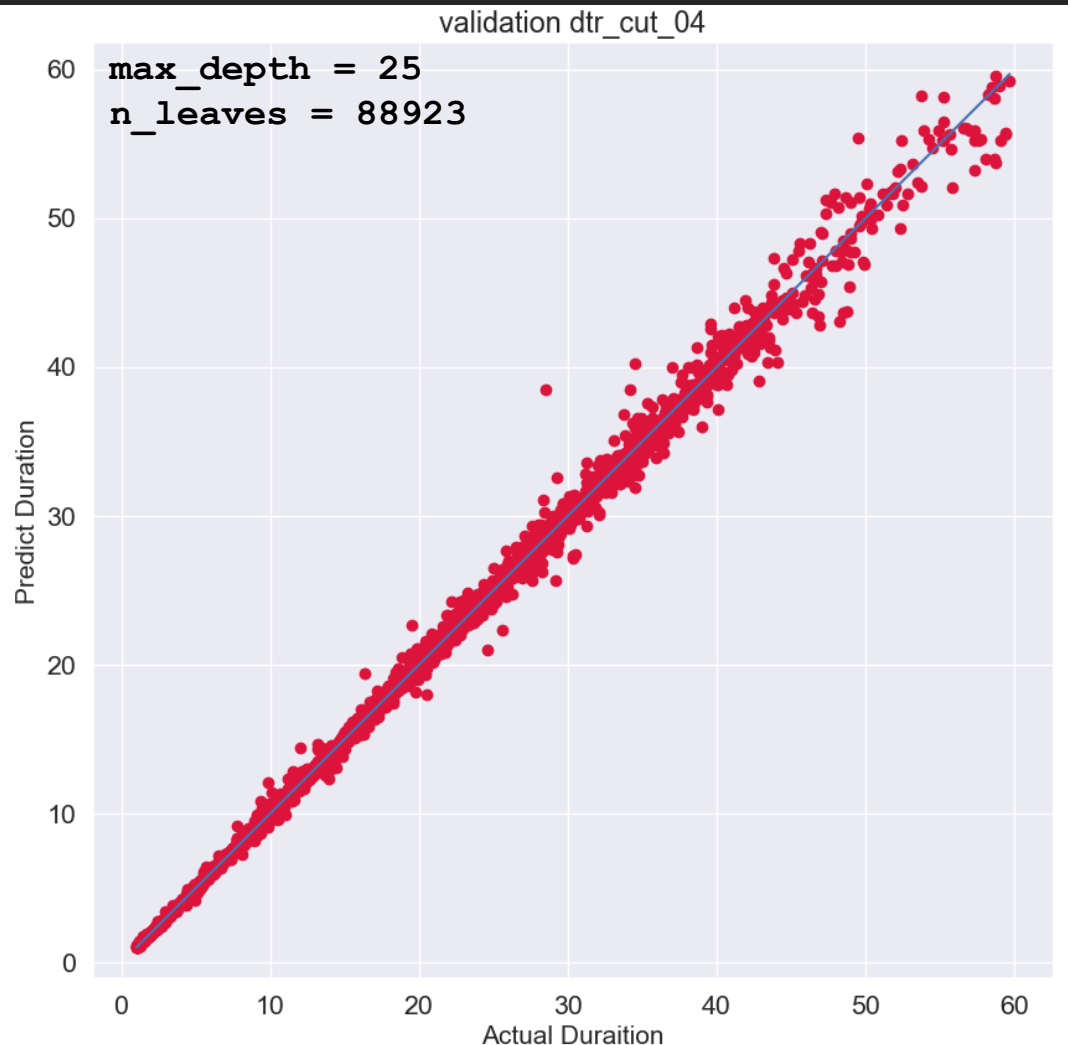
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 0.248$, $MSE = 4.219$, $RMSE = 2.054$, $R^2 = 0.925$
$Validation$: $MAE = 0.328$, $MSE = 4.251$, $RMSE = 2.061$, $R^2 = 0.924$

$Train$: $MAE = 1.517$, $MSE = 1.691$, $RMSE = 1.301$, $R^2 = 1$
$Validation$: $MAE = 0.075$, $MSE = 0.065$, $RMSE = 0.255$, $R^2 = 0.998$





dtr_05

FEATURES:

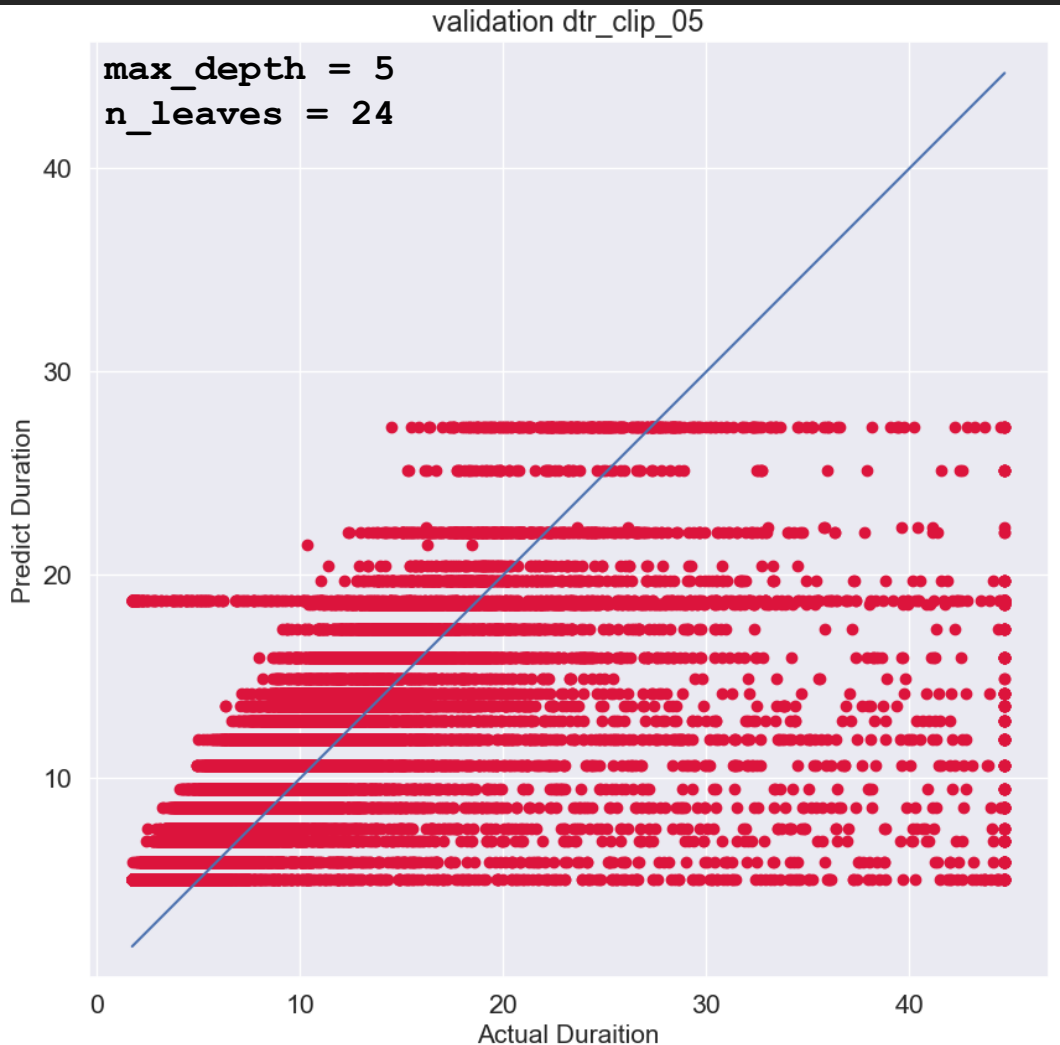
['trip_distance']

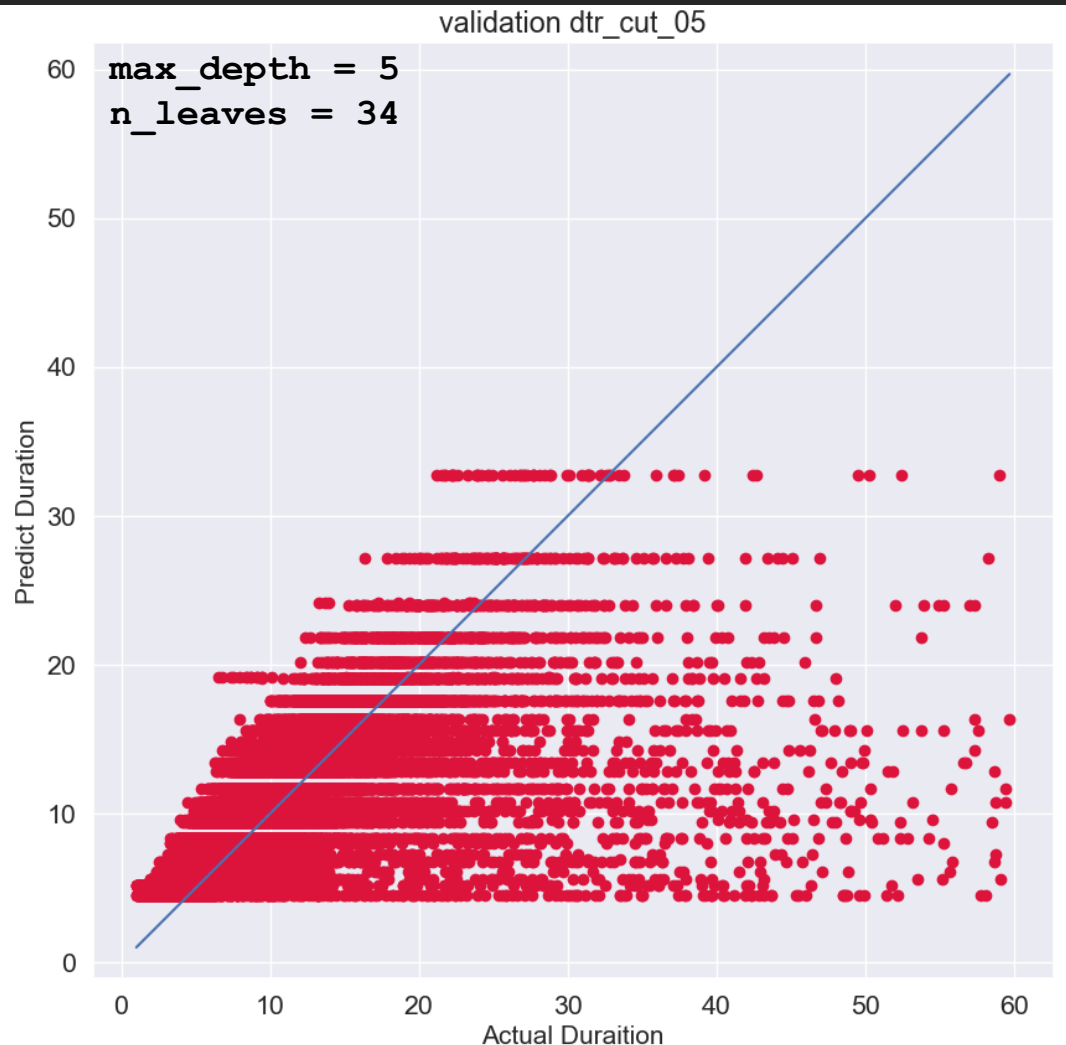
HP:

max_dep = 5
min_samp_split = 5
min_samp_leaf = 5
ccp = 0.0

$Train$: $MAE = 3.322$, $MSE = 34.671$, $RMSE = 5.888$, $R^2 = 0.387$
$Validation$: $MAE = 3.306$, $MSE = 34.666$, $RMSE = 5.887$, $R^2 = 0.388$

$Train$: $MAE = 2.978$, $MSE = 28.166$, $RMSE = 5.307$, $R^2 = 0.427$
$Validation$: $MAE = 3.007$, $MSE = 28.867$, $RMSE = 5.372$, $R^2 = 0.431$





dtr_06

FEATURES:

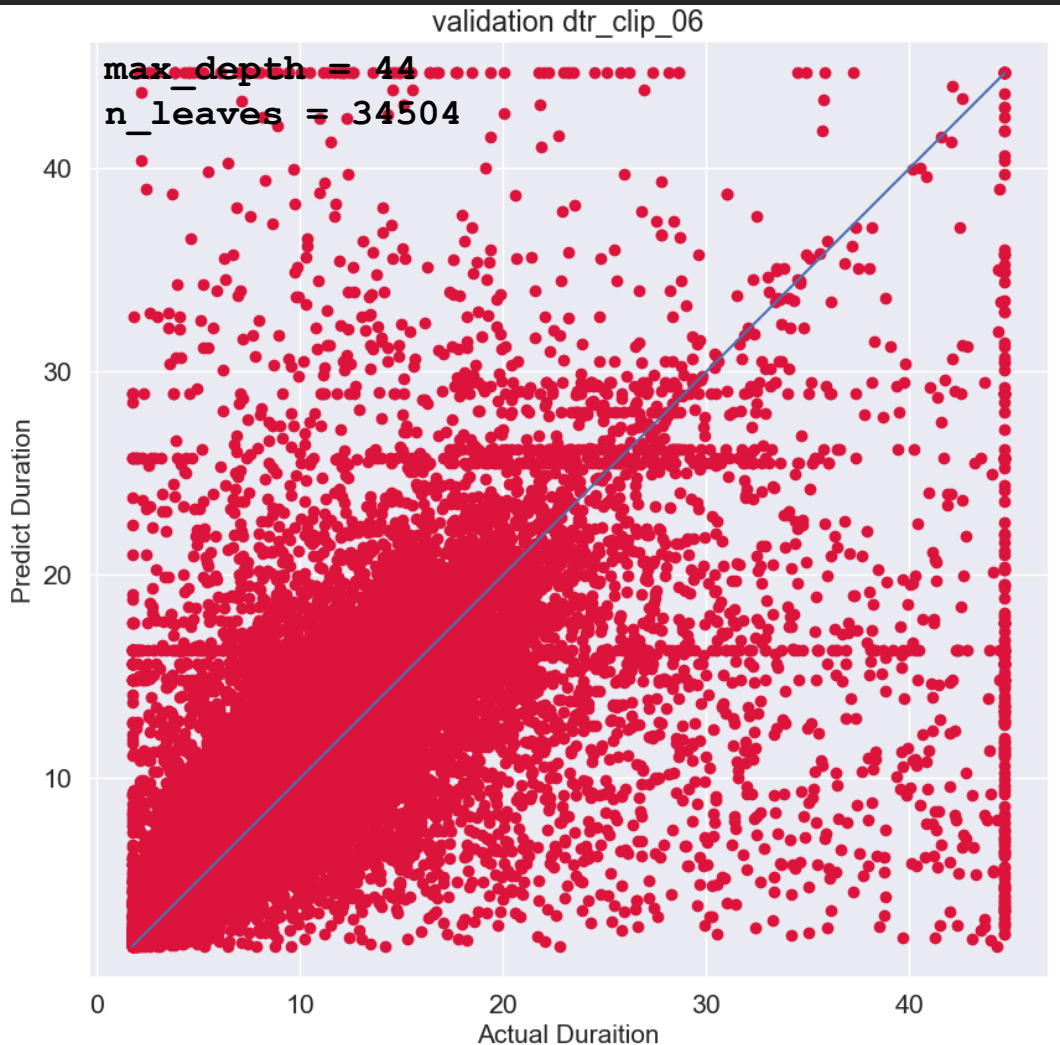
['trip_distance', 'diff_alt_slope_-1', 'diff_alt_slope_0', 'diff_alt_slope_1', 'user_type_Customer', 'user_type_Subscriber', 'member_gender_Female', 'member_gender_Male', 'bike_share_for_all_trip_No', 'bike_share_for_all_trip_Yes']

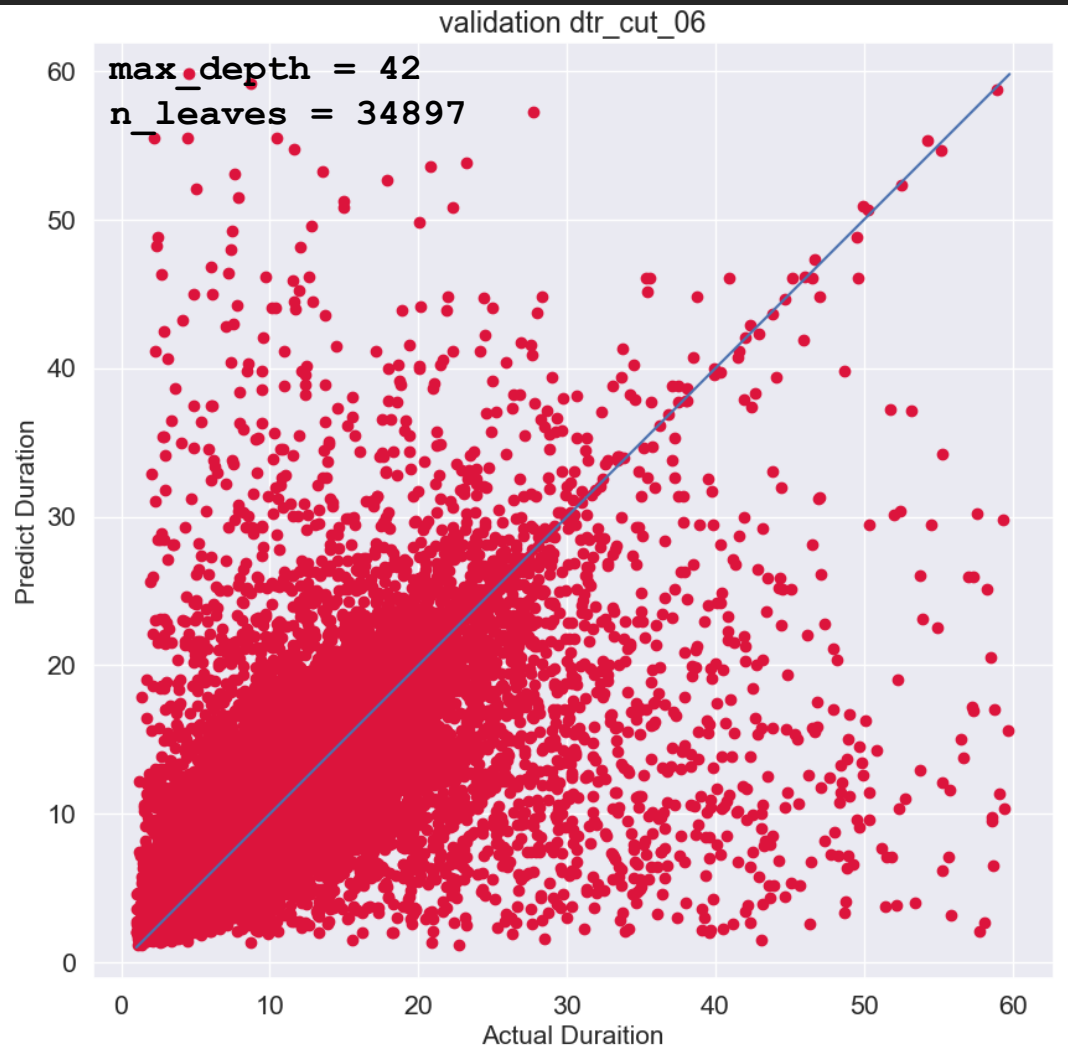
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 1.873$, $MSE = 15.763$, $RMSE = 3.971$, $R^2 = 0.721$
$Validation$: $MAE = 3.168$, $MSE = 37.726$, $RMSE = 6.142$, $R^2 = 0.334$

$Train$: $MAE = 1.584$, $MSE = 10.907$, $RMSE = 3.302$, $R^2 = 0.778$
$Validation$: $MAE = 2.902$, $MSE = 32.634$, $RMSE = 5.712$, $R^2 = 0.356$





dtr_07

FEATURES:

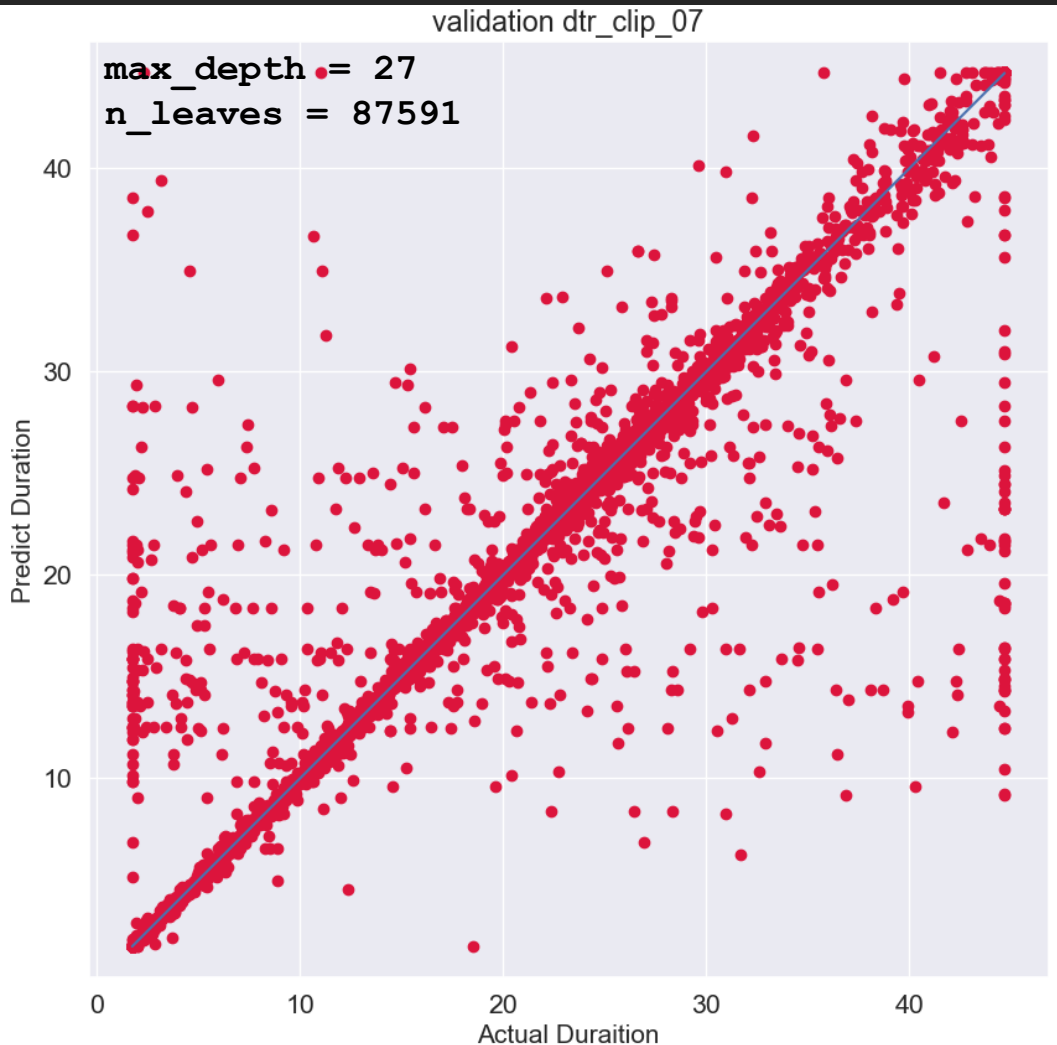
['trip_distance', 'diff_alt', 'avg_trip_speed', 'age', 'diff_alt_slope_-1', 'diff_alt_slope_0', 'diff_alt_slope_1', 'user_type_Customer', 'user_type_Subscriber', 'member_gender_Female', 'member_gender_Male', 'bike_share_for_all_trip_No', 'bike_share_for_all_trip_Yes']

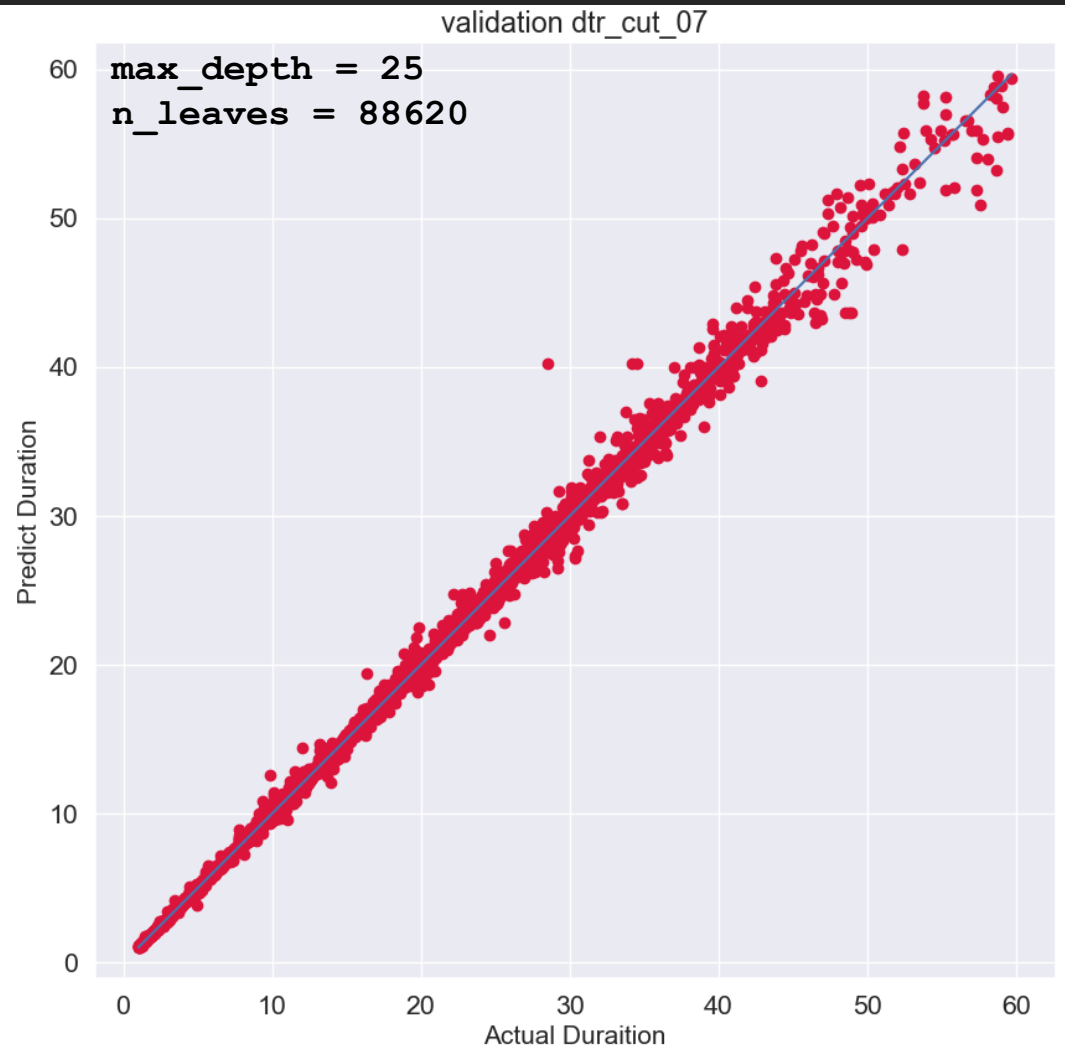
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

$Train$: $MAE = 0.211$, $MSE = 3.385$, $RMSE = 1.839$, $R^2 = 0.941$
$Validation$: $MAE = 0.318$, $MSE = 4.253$, $RMSE = 2.062$, $R^2 = 0.924$

$Train$: $MAE = 1.546$, $MSE = 1.771$, $RMSE = 1.331$, $R^2 = 1$
$Validation$: $MAE = 0.078$, $MSE = 0.0711$, $RMSE = 0.266$, $R^2 = 0.998$





Models Comparison

Clipping_df

lr/dtr	#	tag	part	MAE	MSE	RMSE	R2
dtr	00	clip	train	5.341	56.623	7.524	0
dtr	00	clip	valid	5.358	56.675	7.528	0
dtr	01	clip	train	2.744	26.087	5.107	0.539
dtr	01	clip	valid	3.179	33.634	5.799	0.406
dtr	02	clip	train	2.517	23.461	4.843	0.585
dtr	02	clip	valid	3.205	35.314	5.942	0.376
dtr	03	clip	train	0.257	4.412	2.101	0.922
dtr	03	clip	valid	0.325	4.223	2.055	0.925
dtr	04	clip	train	0.248	4.219	2.054	0.925
dtr	04	clip	valid	0.328	4.251	2.061	0.924
dtr	05	clip	train	3.322	34.671	5.888	0.387
dtr	05	clip	valid	3.306	34.666	5.887	0.388
dtr	06	clip	train	1.873	15.763	3.971	0.721
dtr	06	clip	valid	3.168	37.726	6.142	0.334
dtr	07	clip	train	0.211	3.385	1.839	0.941
dtr	07	clip	valid	0.318	4.253	2.062	0.924

Cutting_df

lr/dtr	#	tag	part	MAE	MSE	RMSE	R2
dtr	00	cut	train	5.026	49.197	7.014	0
dtr	00	cut	valid	5.084	50.731	7.122	0
dtr	01	cut	train	2.405	20.088	4.482	0.597
dtr	01	cut	valid	2.844	27.532	5.247	0.457
dtr	02	cut	train	2.177	17.583	4.193	0.642
dtr	02	cut	valid	2.901	29.413	5.423	0.421
dtr	03	cut	train	1.387	1.495	1.222	1
dtr	03	cut	valid	0.071	0.061	0.248	0.998
dtr	04	cut	train	1.517	1.691	1.301	1
dtr	04	cut	valid	0.075	0.065	0.255	0.998
dtr	05	cut	train	2.978	28.166	5.307	0.427
dtr	05	cut	valid	3.007	28.867	5.372	0.431
dtr	06	cut	train	1.584	10.907	3.302	0.778
dtr	06	cut	valid	2.902	32.634	5.712	0.356
dtr	07	cut	train	1.546	1.771	1.331	1
dtr	07	cut	valid	0.078	0.071	0.266	0.998

dtr_test

FEATURES:

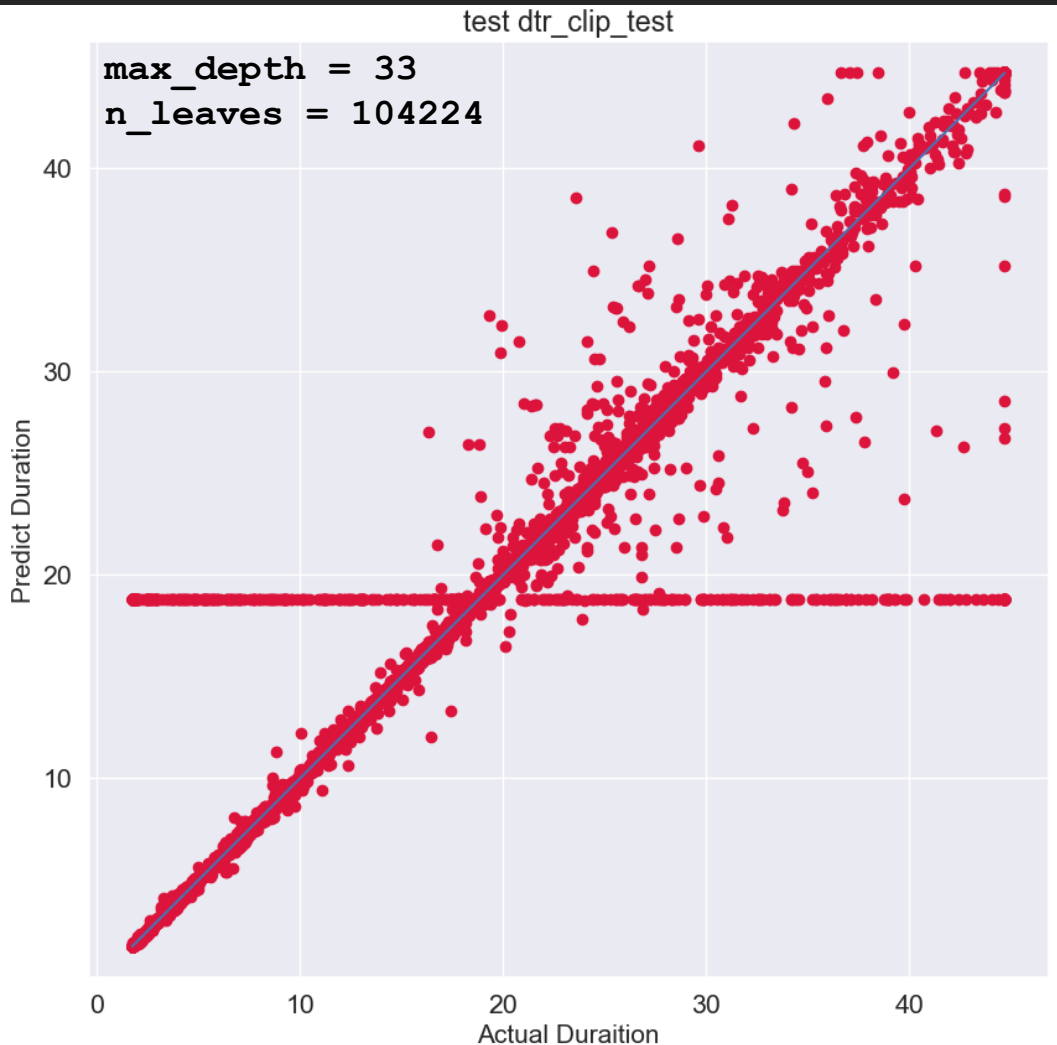
['trip_distance', 'diff_alt', 'avg_trip_speed']

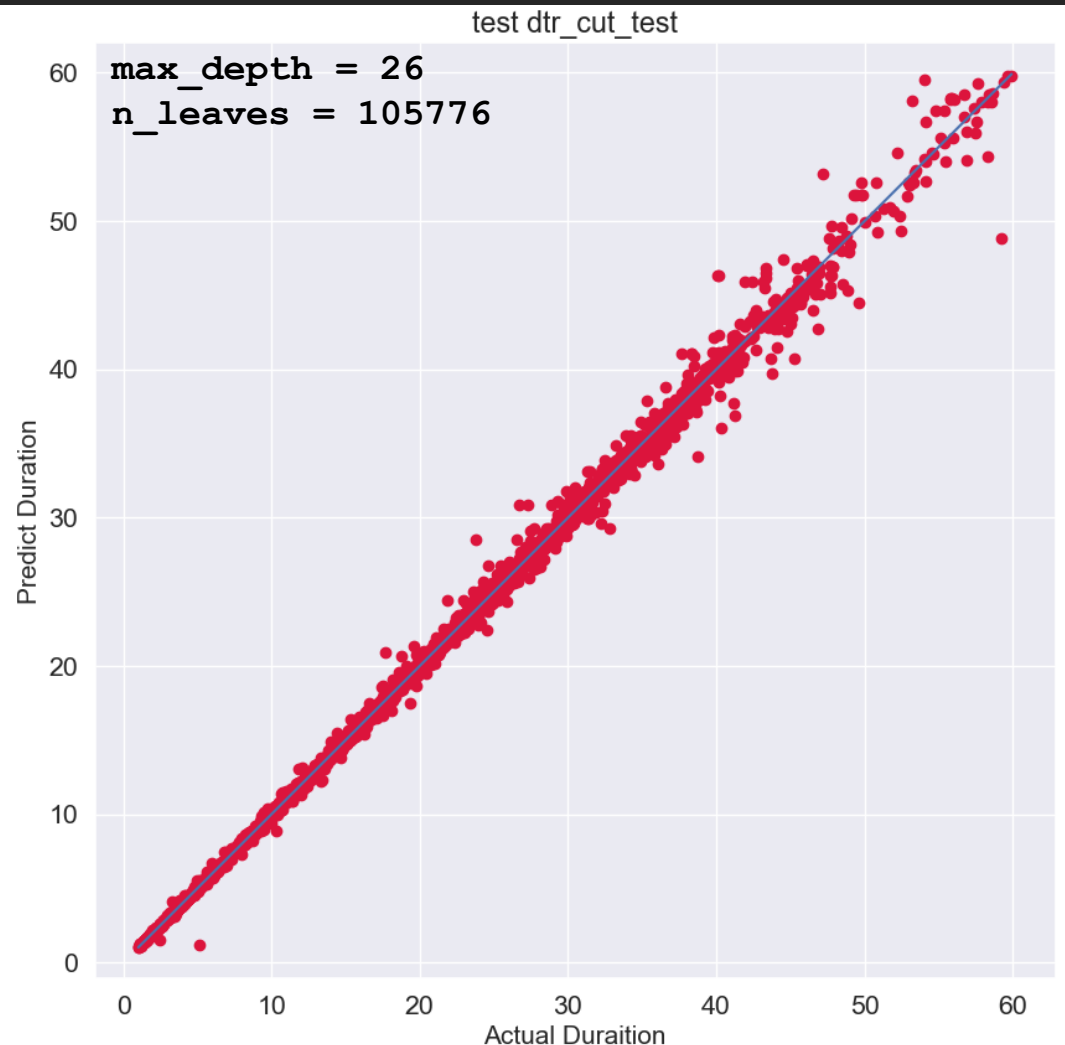
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

Train:	$MAE = 0.253,$	$MSE = 4.334,$	$RMSE = 2.081,$	$R^2 = 0.923$
Test:	$MAE = 0.336,$	$MSE = 4.471,$	$RMSE = 2.114,$	$R^2 = 0.921$

Train:	$MAE = 1.7e^{-17},$	$MSE = 1.9e^{-32},$	$RMSE = 1.3e^{-16},$	$R^2 = 1$
Test:	$MAE = 0.064,$	$MSE = 0.056,$	$RMSE = 0.237,$	$R^2 = 0.998$





Decision-Tree Conclusions

The best model we created trough the Decision-Tree method was **lr_03** which based on the following features:
['trip_distance', 'diff_alt`, `avg_trip_speed`]

lr/dtr	#	tag	part	MAE	MSE	RMSE	R2
dtr	test	clip	train	0.253	4.334	**2.081**	**0.923**
dtr	test	clip	test	0.336	4.471	**2.114**	**0.921**
dtr	test	cut	train	1.75E-17	1.94E-32	**1.39E-16**	**1**
dtr	test	cut	test	0.064	0.056	**0.237**	**0.998**

*The RMSE is minutes.

Make sense ??? Business insight ???

We didn't find any business insight when we create the model based on the whole data of the user when it finish to use.
In addition, we based the model on features when those calculated by the target itself.

Genius but instructive!

So, we made another session to think how we can use the current code to get some insights.

Practically, the features we can use as an estimators for a model which can contribute an insight are the features were we know about the user at the beginning of using:

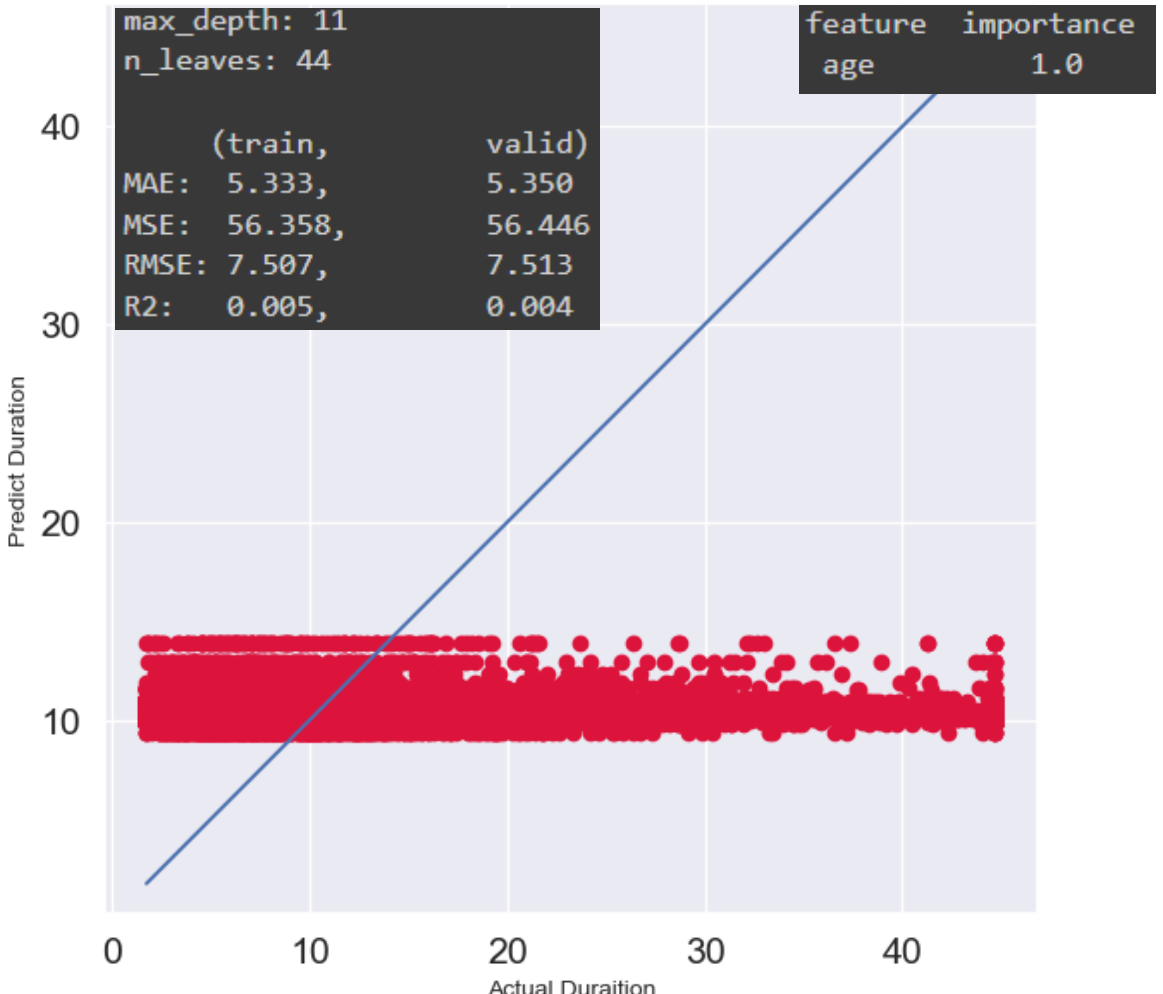
- Age
- User type (Customer/ Subscriber)
- Gender (Male/ Female)
- Payment method (Direct/Indirect)

Ndtr_01

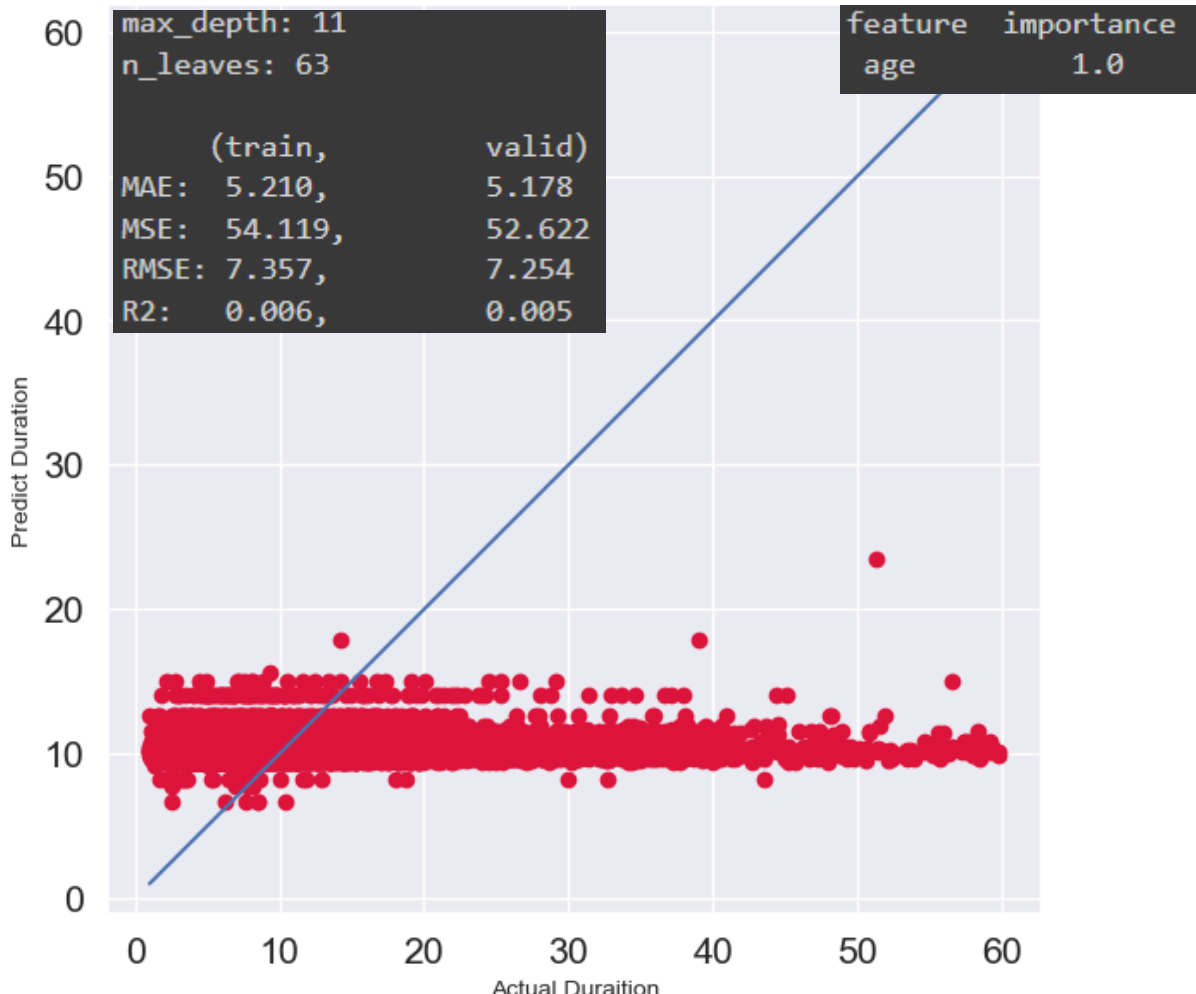
FEATURES:
['age']

HP:
max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

clip_valid_01



cut_valid_01



Ndtr_02

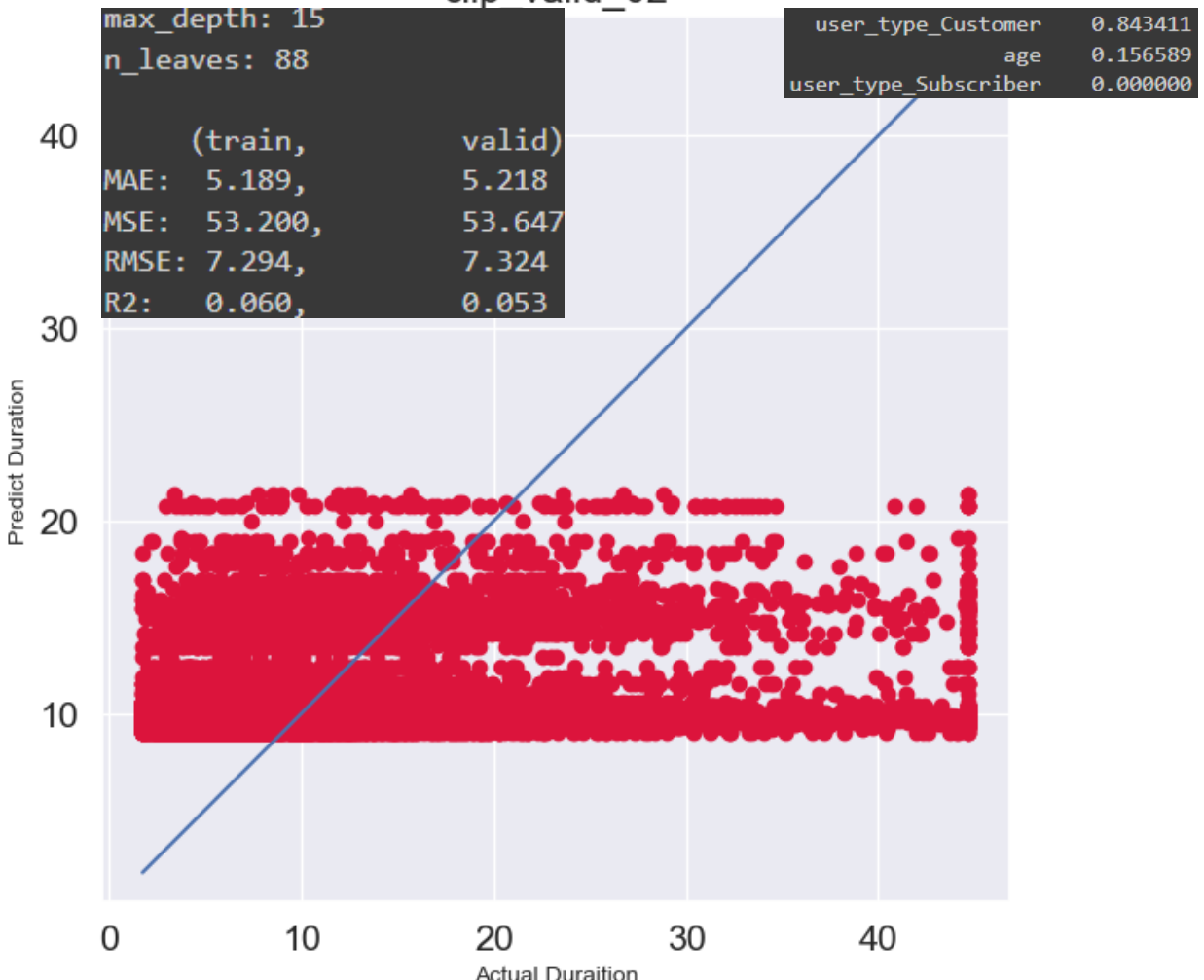
FEATURES:

['age', 'user_type_Customer', 'user_type_Subscriber']

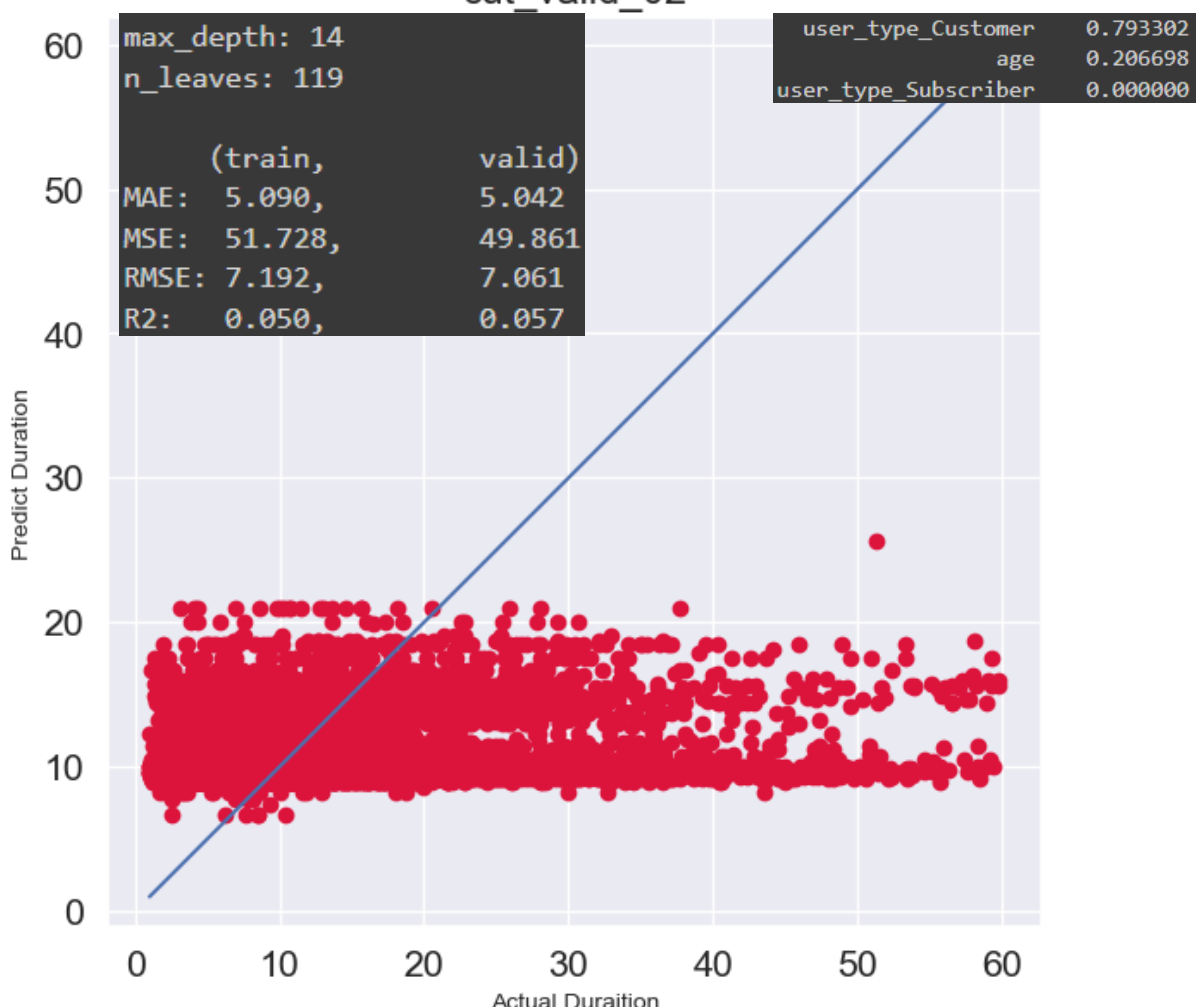
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

clip_valid_02



cut_valid_02



Ndtr_03

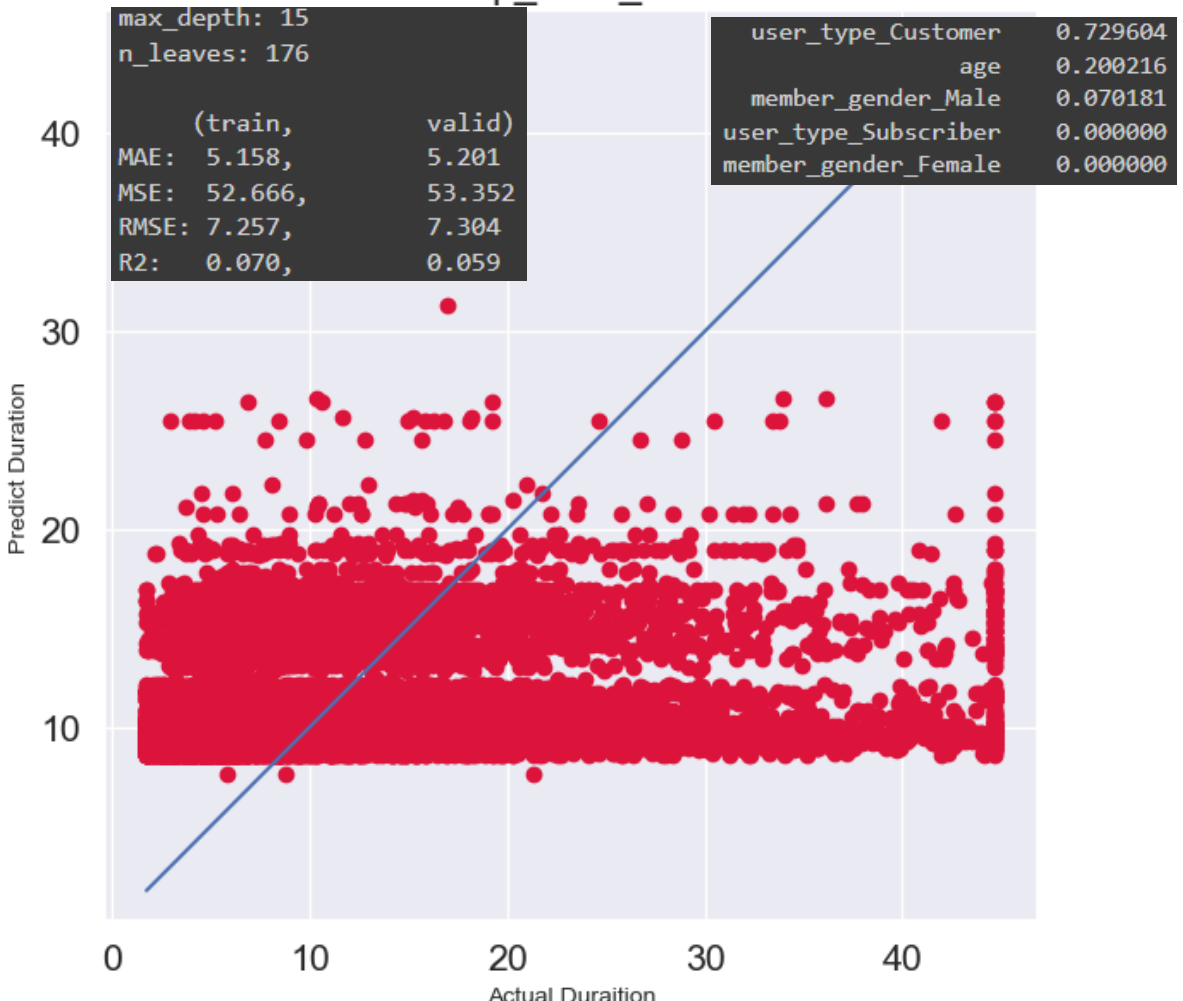
FEATURES:

['age', 'user_type_Customer', 'user_type_Subscriber', 'member_gender_Female', 'member_gender_Male']

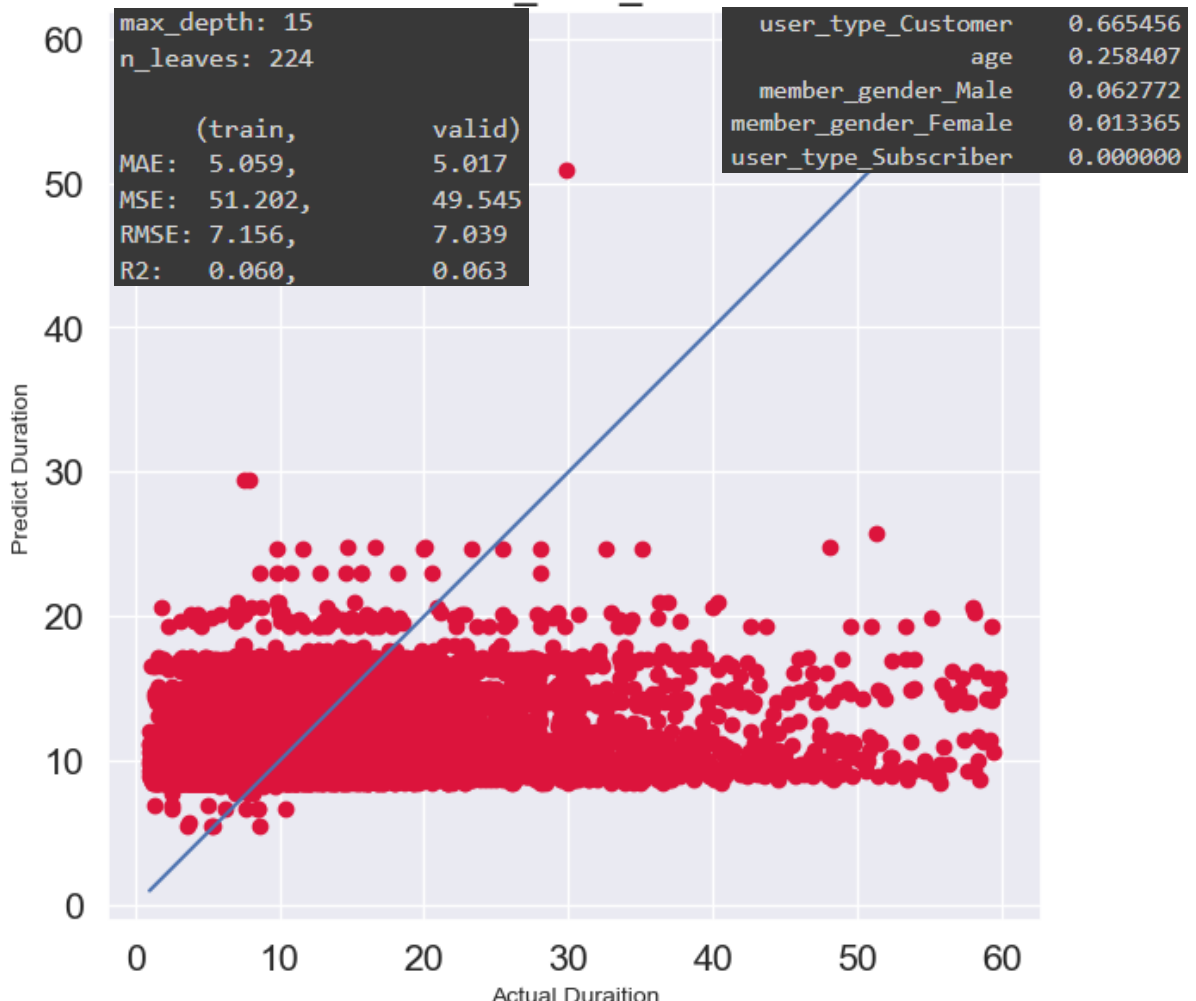
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

clip_valid_03



cut_valid_03



Ndtr_04

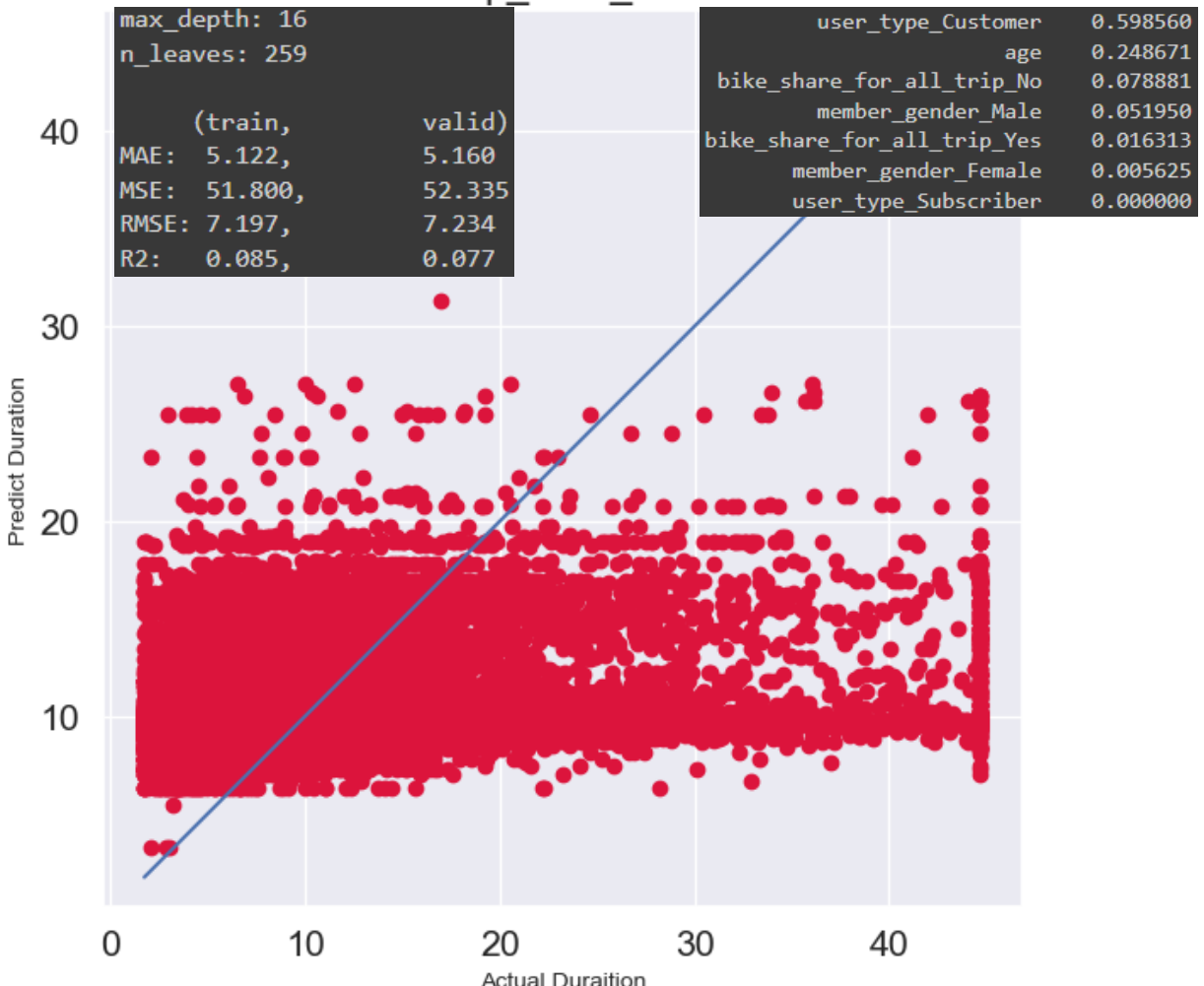
FEATURES:

['age', 'user_type_Customer', 'user_type_Subscriber', 'member_gender_Female', 'member_gender_Male']

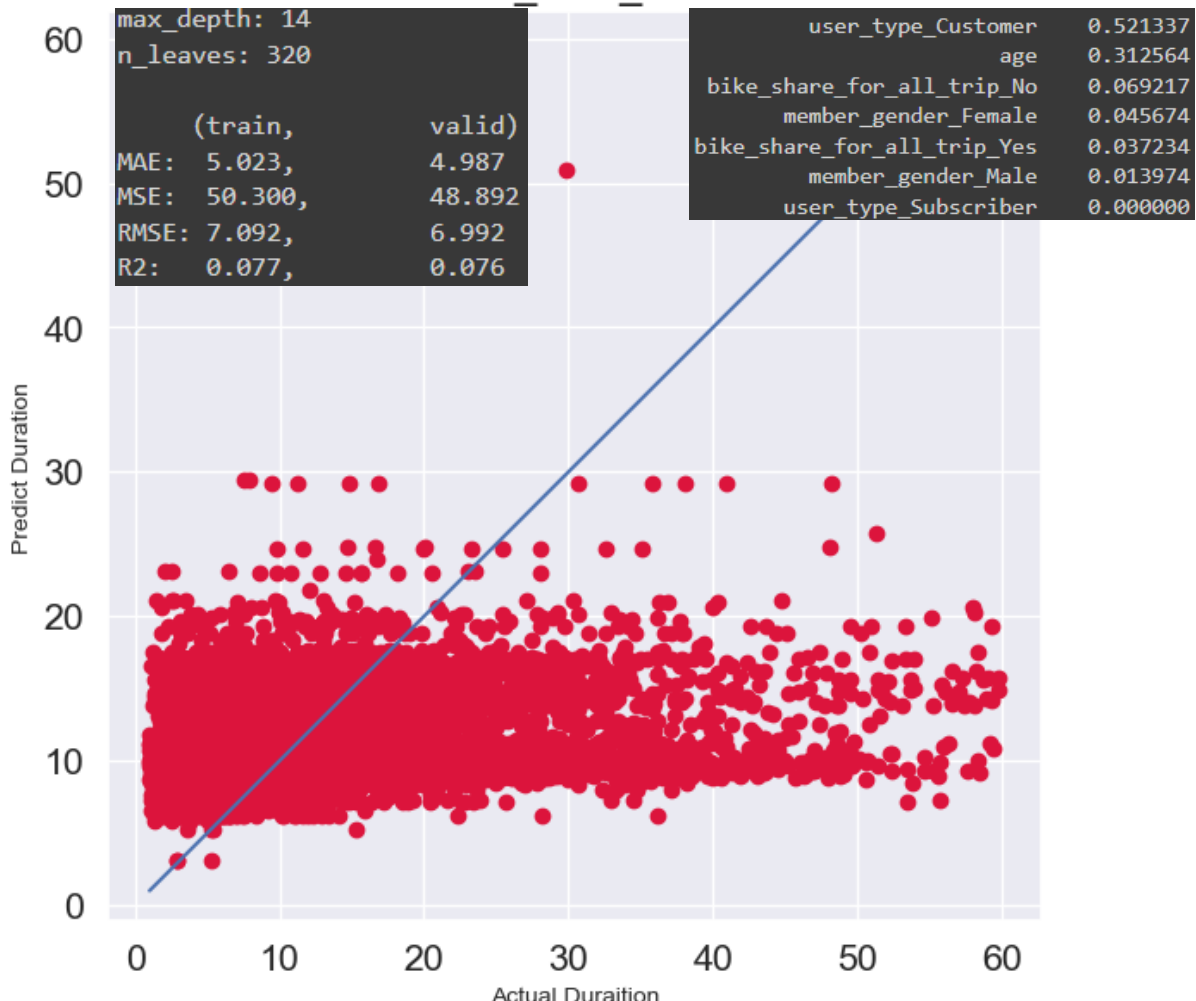
HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0

clip_valid_04



cut_valid_04



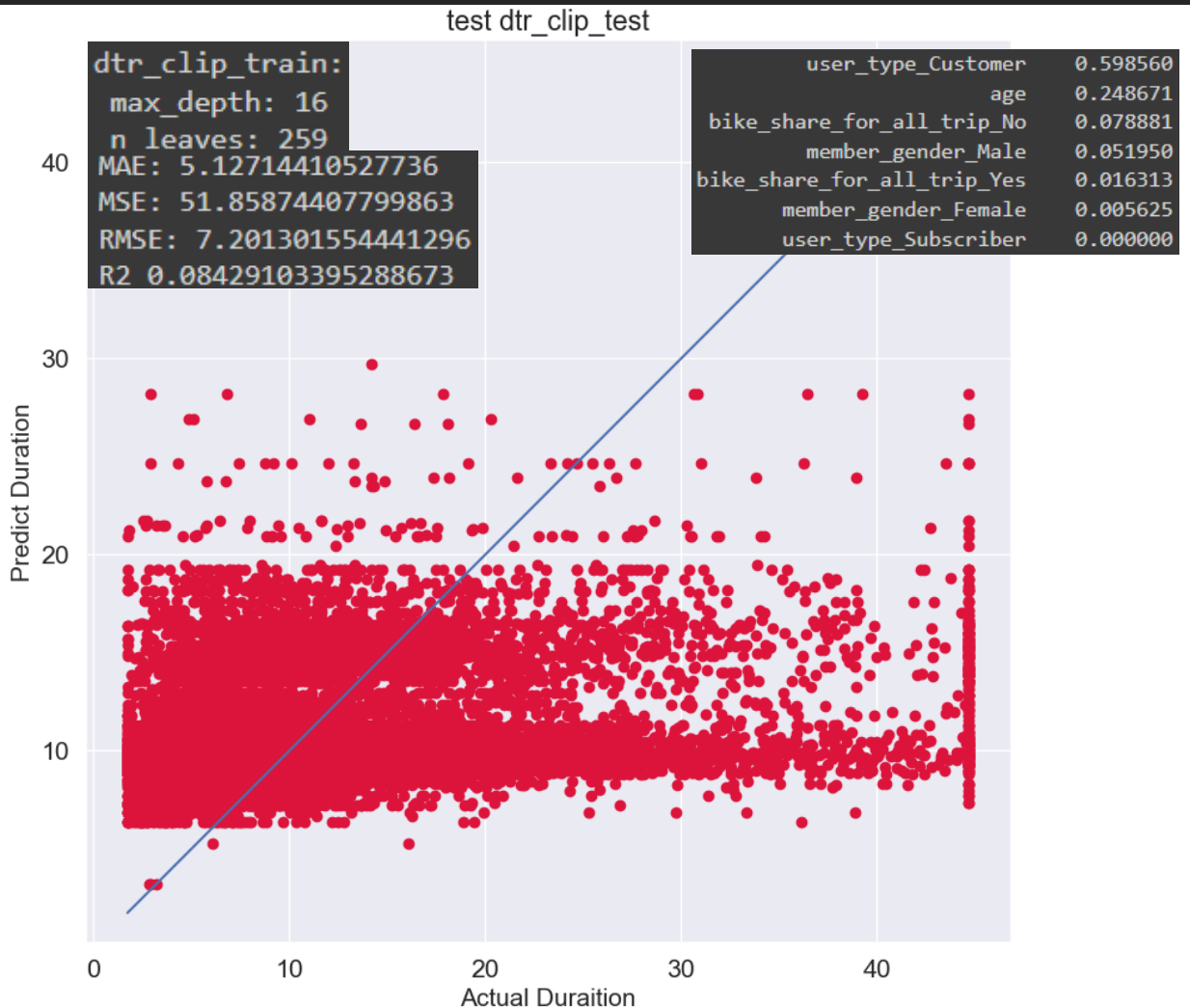
Ndtr_test

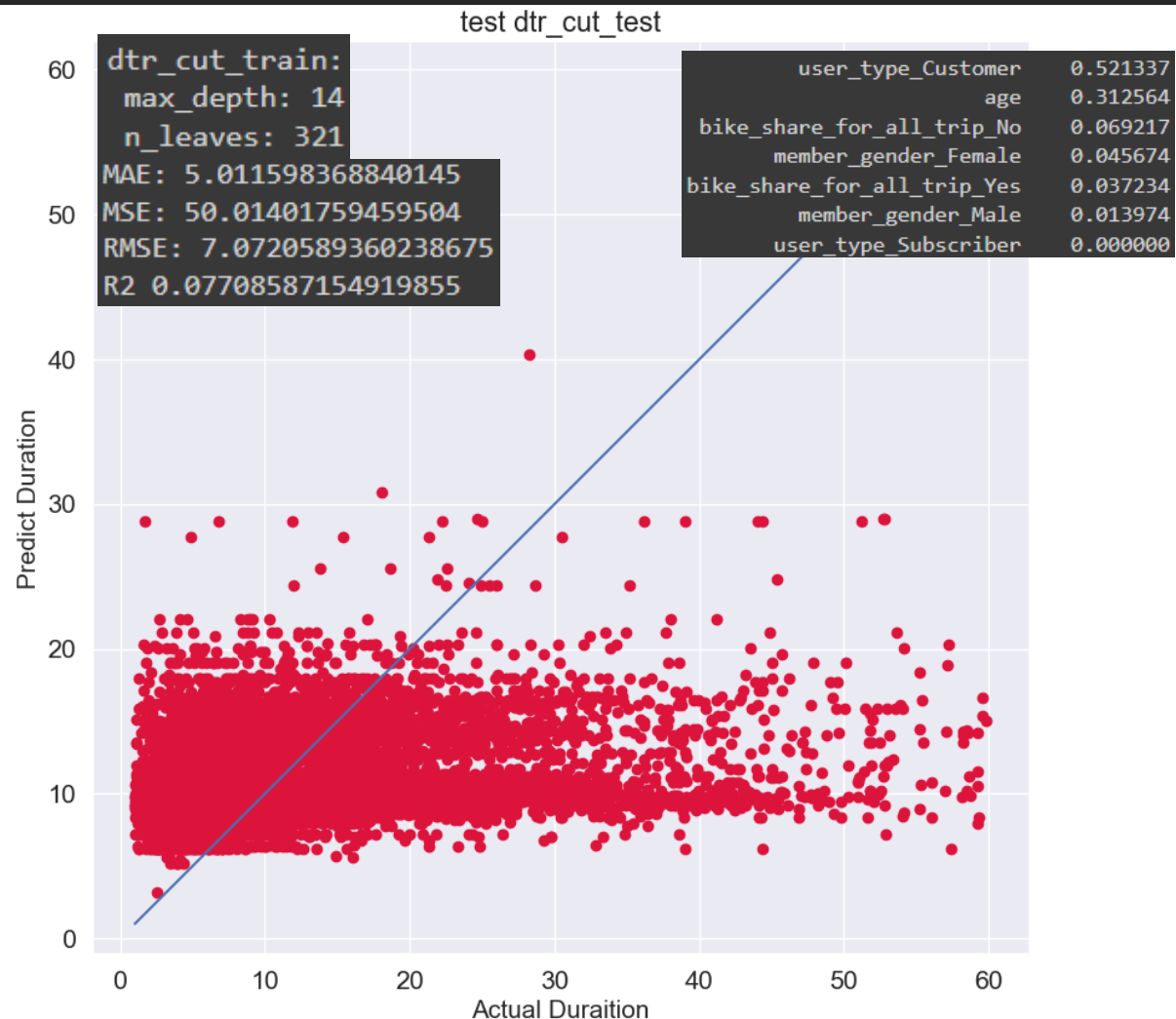
FEATURES:

['age', 'user_type_Customer', 'user_type_Subscriber', 'member_gender_Female', 'member_gender_Male']

HP:

max_dep = None
min_samp_split = 2
min_samp_leaf = 1
ccp = 0.0



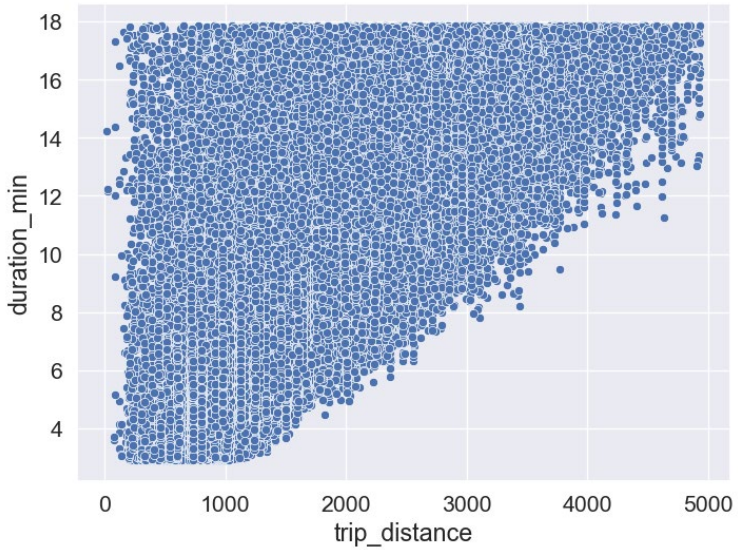


Thanks!

Statistical tuning data

```
def func_statistics_cleaning_numerical_values(
                                        data, x_col, y_col,
                                        x_k=3, max_x_bound=1, min_x_bound=1,
                                        y_k=3, max_y_bound=1, min_y_bound=1
                                        ):
```

```
Raw data:                    (181434, 14)
Splitting data:              (148053, 14)
Train data shape is :        (103637, 14)
Validation data shape is:    (22208, 14)
Test data shape is :         (22208, 14)
Devaluation of 18.39%
```



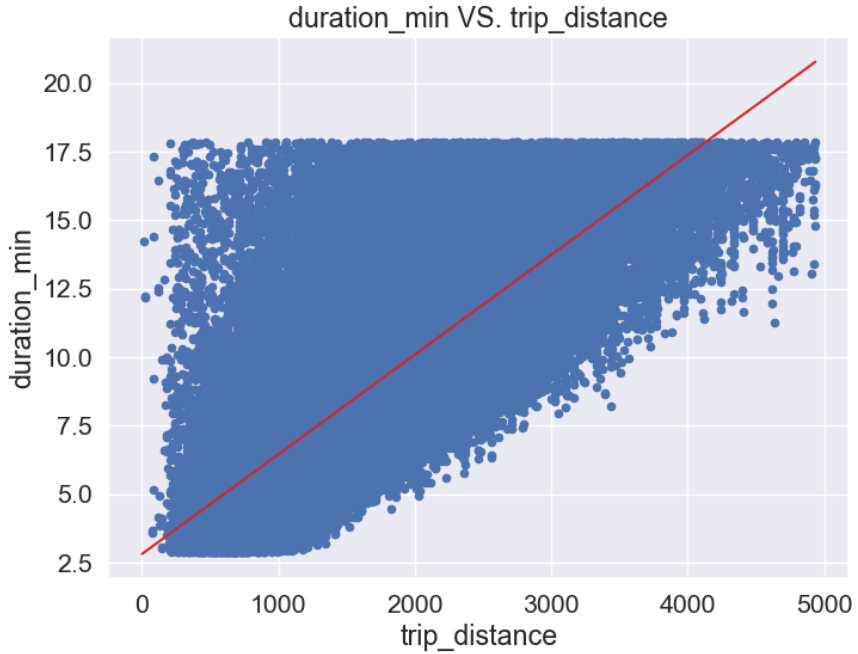
lr_01

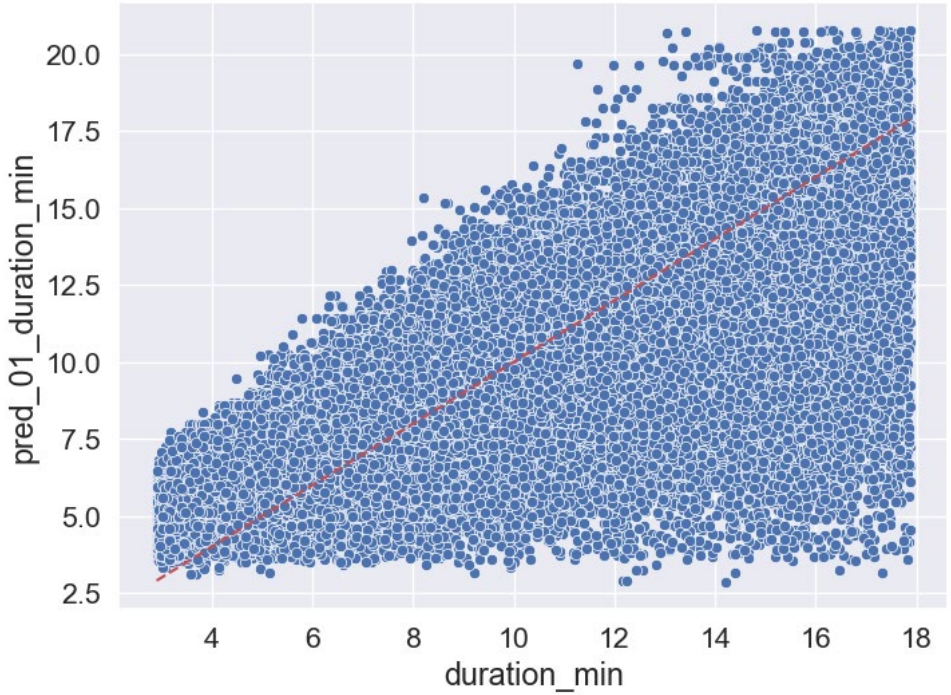
duration_min vs. trip_distance

TARGET = 'duration_min'
FEATURES = 'trip_distance'

The empirical equation is

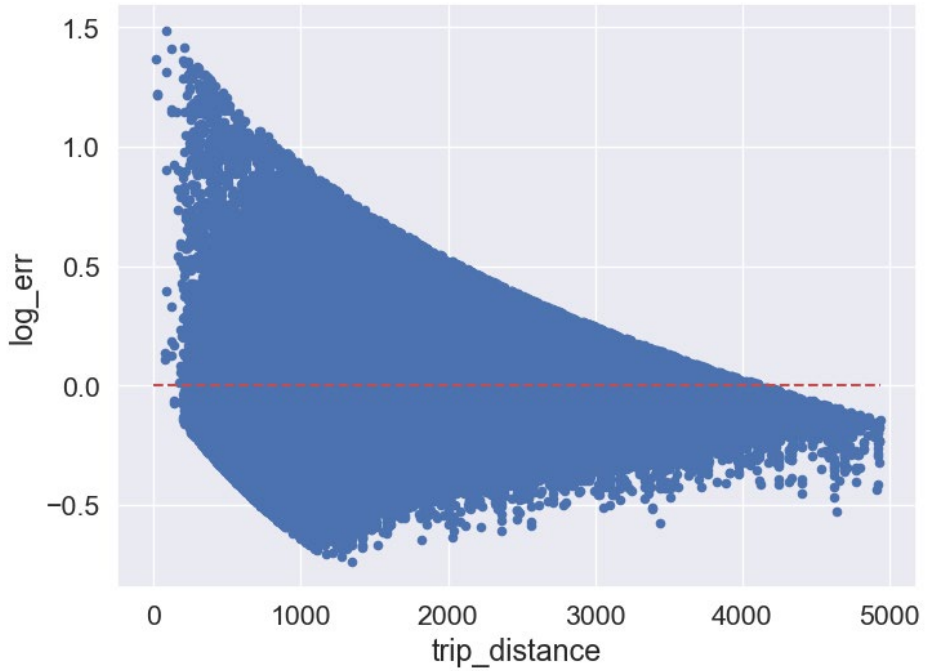
$y = 0.0036x + 2.8227$



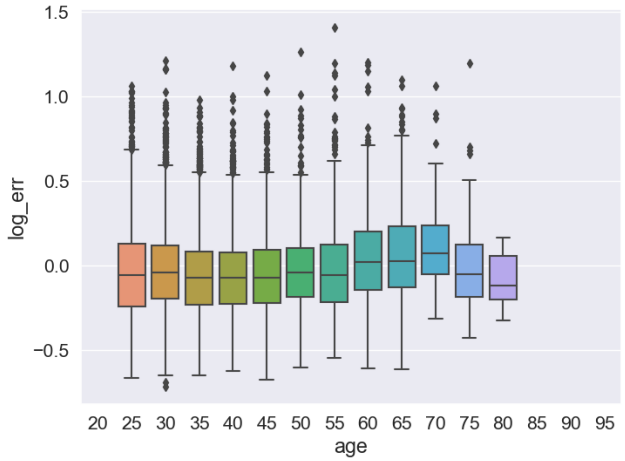


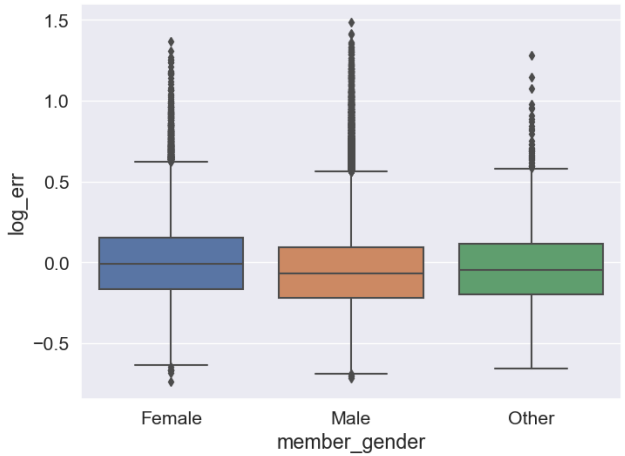
Residual

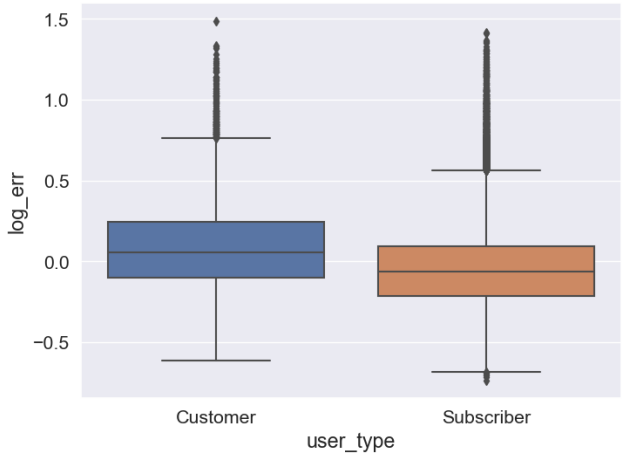
Exploration `log errors` vs. `target_value`

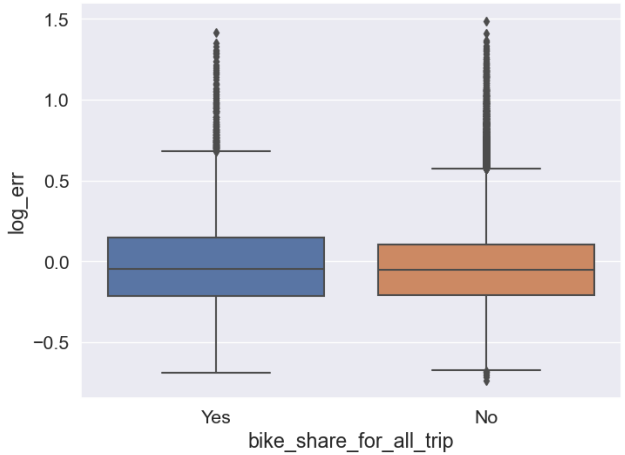


Exploration additional relationships for potential improvement of tuning

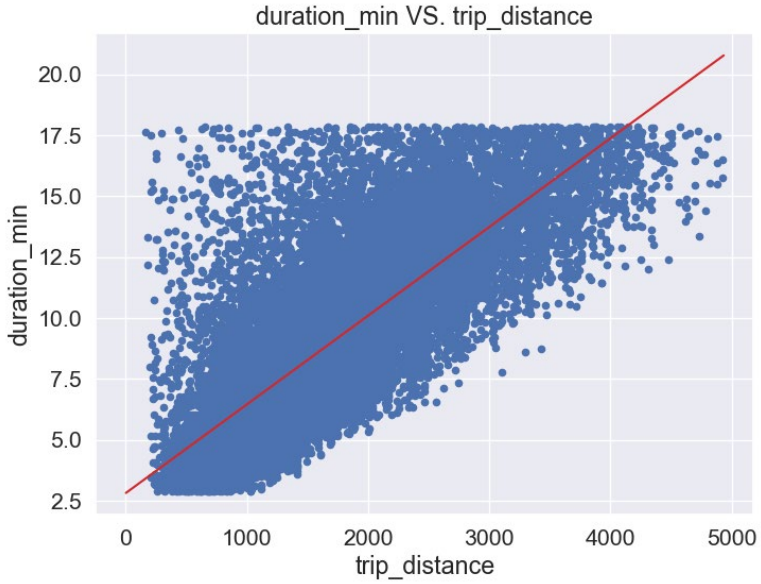


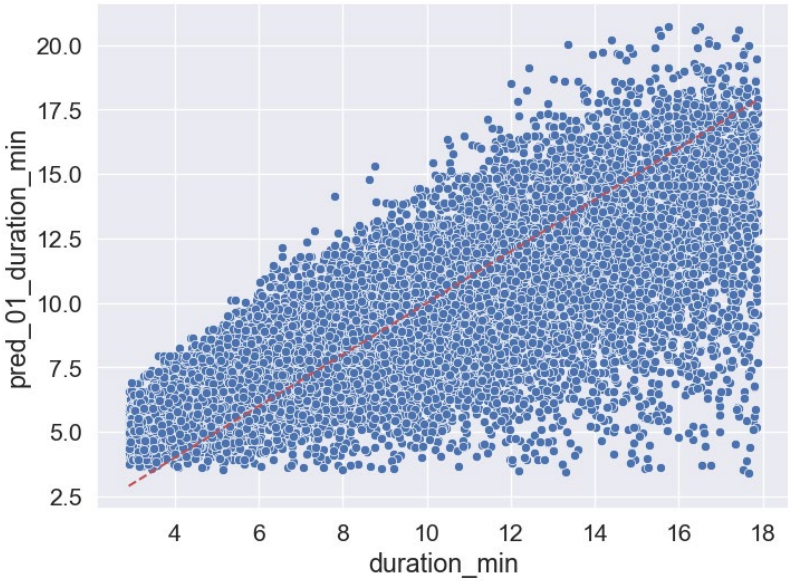


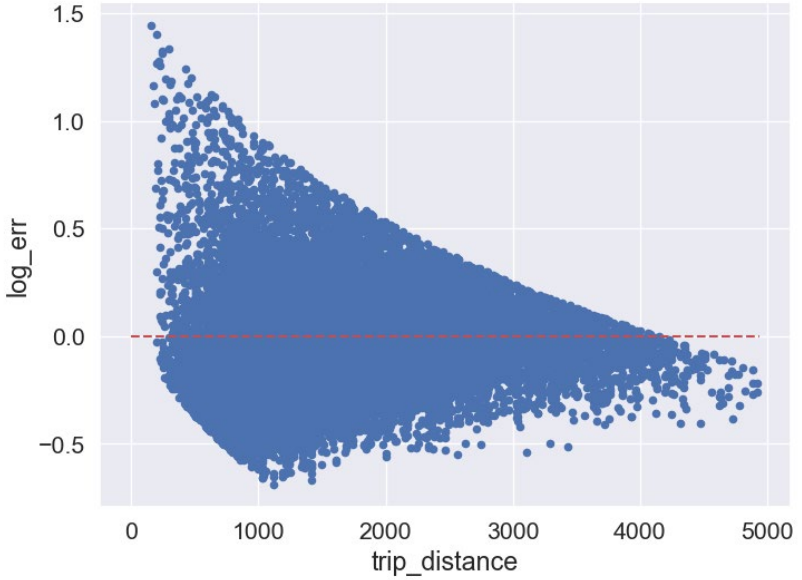




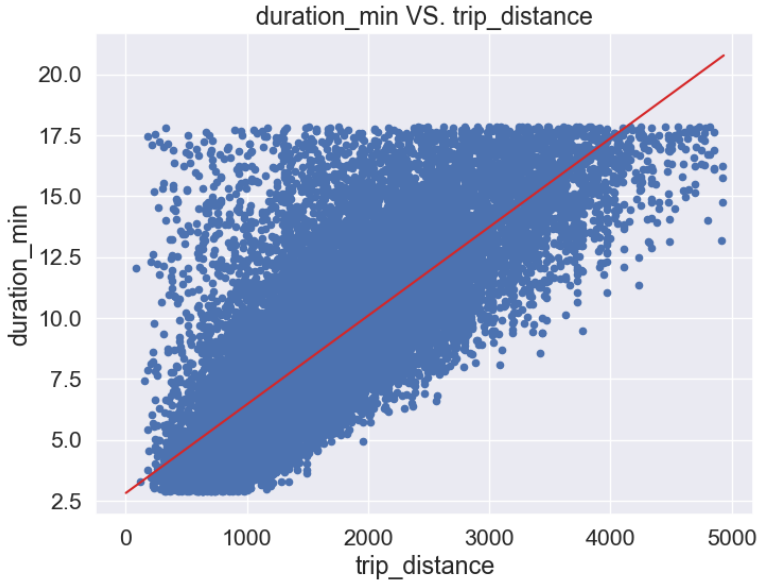
Validate the model

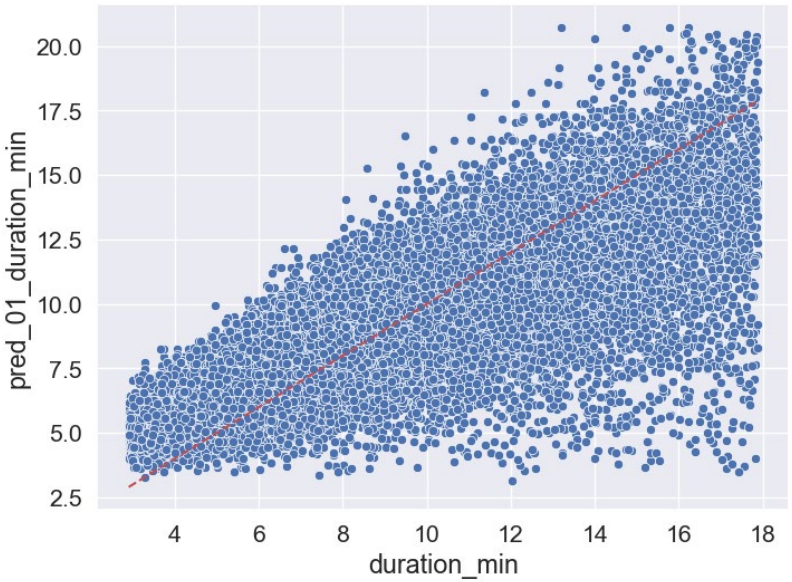


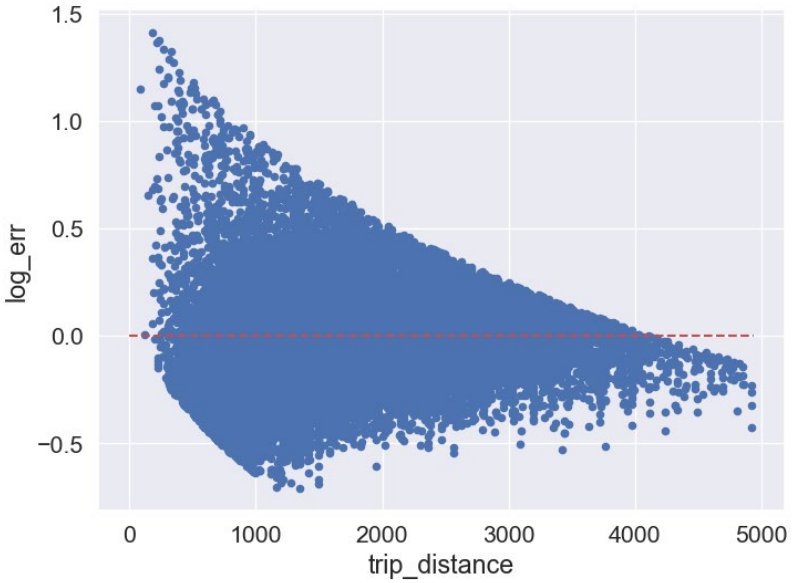




Testing the model







Evaluation

RMSE score
As lower *RMSE score* as better the prediction results;
For this project, the *RMSE score* value represent a minutes.

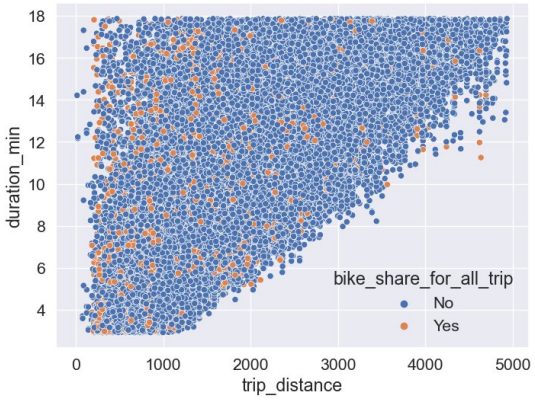
R^2 score
Coefficient of Determination.
The best possible *R^2 score* is **1.0** and it can be negative (because the model can be arbitrarily worse).
Generally, when the true Y is non-constant, a constant model that always predicts the Average of Y disregarding the input features would get a *R^2 score* of 0.

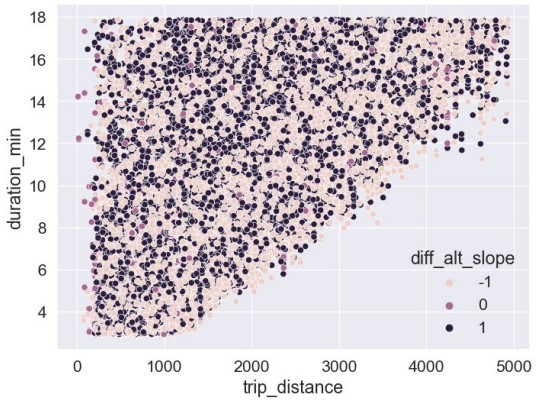
lr_01	fdf_train	fdf_valid	fdf_test
RMSE score	2.3723961060413967	2.365876472027405	2.37110246288299
R^2 score	0.6110326559694819	0.6137572392225006	0.6136474098447284

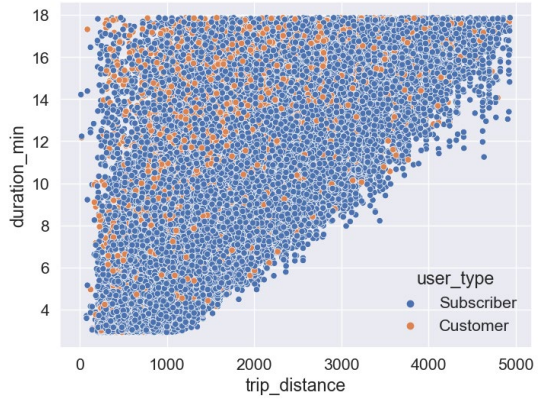
lr_02

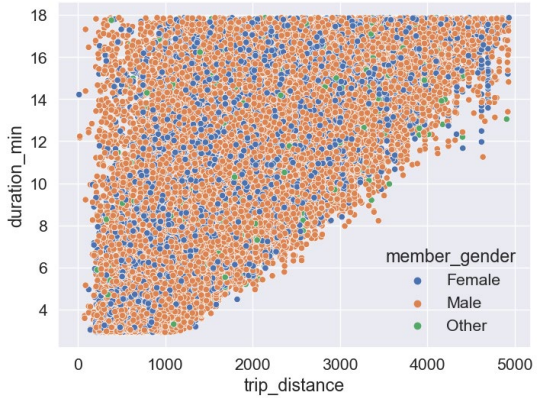
duration_min vs. trip_distance, member_gender, user_type, bike_share_for_all_trip, diff_alt_slope, `avg_trip_speed`

Enumerate categorical columns:['gender_numeric', 'numeric_type', 'numeric_all_trip']
TARGET = 'duration_min'
FEATURES = ['trip_distance', 'gender_numeric', 'numeric_type', 'numeric_all_trip','diff_alt_slope','avg_trip_speed']



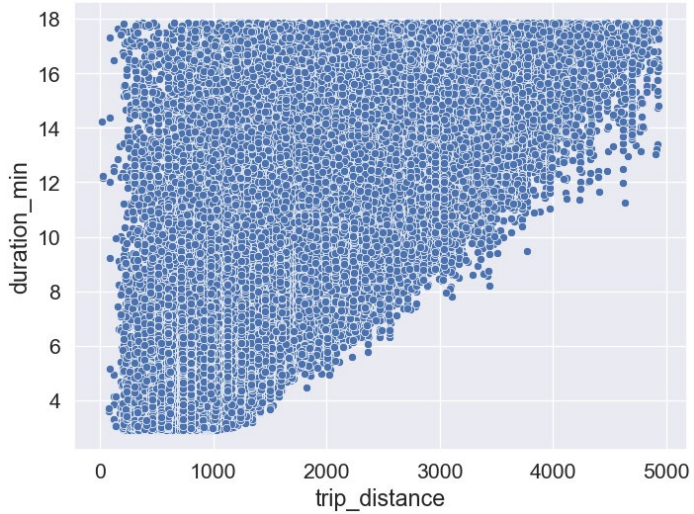


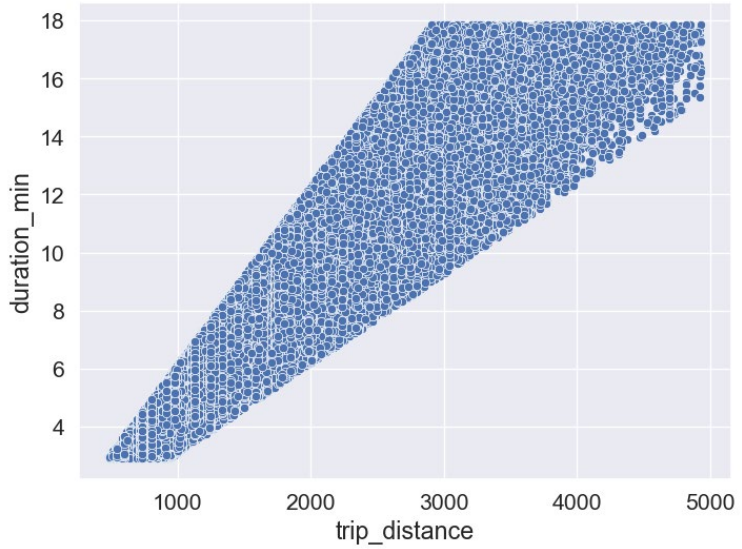




Circumstantial cleaning

By cleaning the observations below the average speed minus 1 standard deviation threshold we clean the users which probably take a tour and using Ford-Go-Bike service as a “tour vehicle” and not as a transportation tool






```
Raw data:                    (148053, 17)
Splitting data:              (101277, 17)
Train data shape is :        (70893, 17)
Validation data shape is:    (15192, 17)
Test data shape is :         (15192, 17)
```

Another devaluation of 31.59%

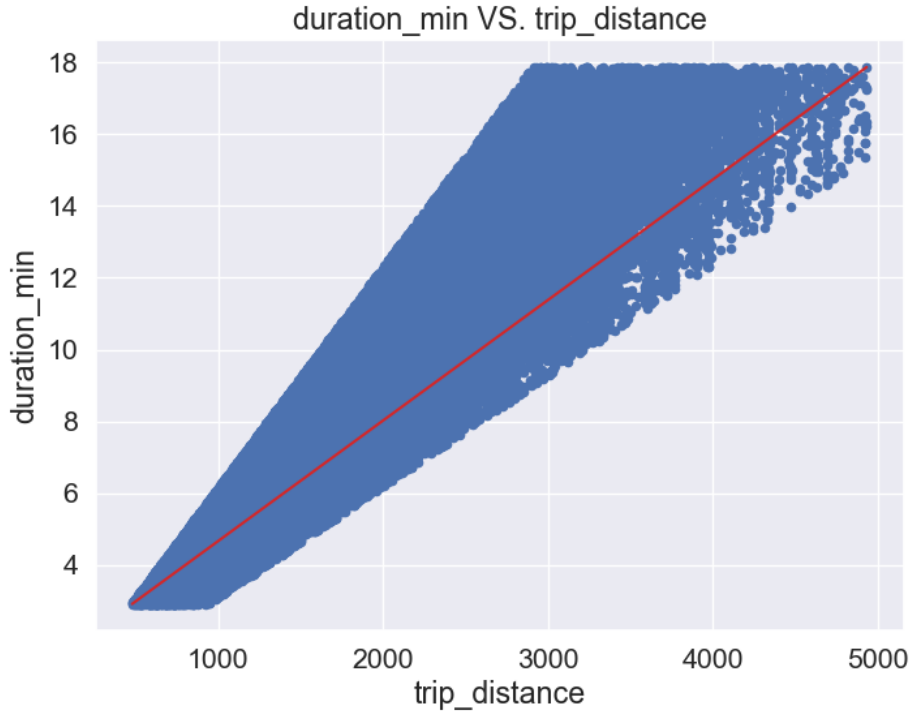
Total devaluation of 44.17%

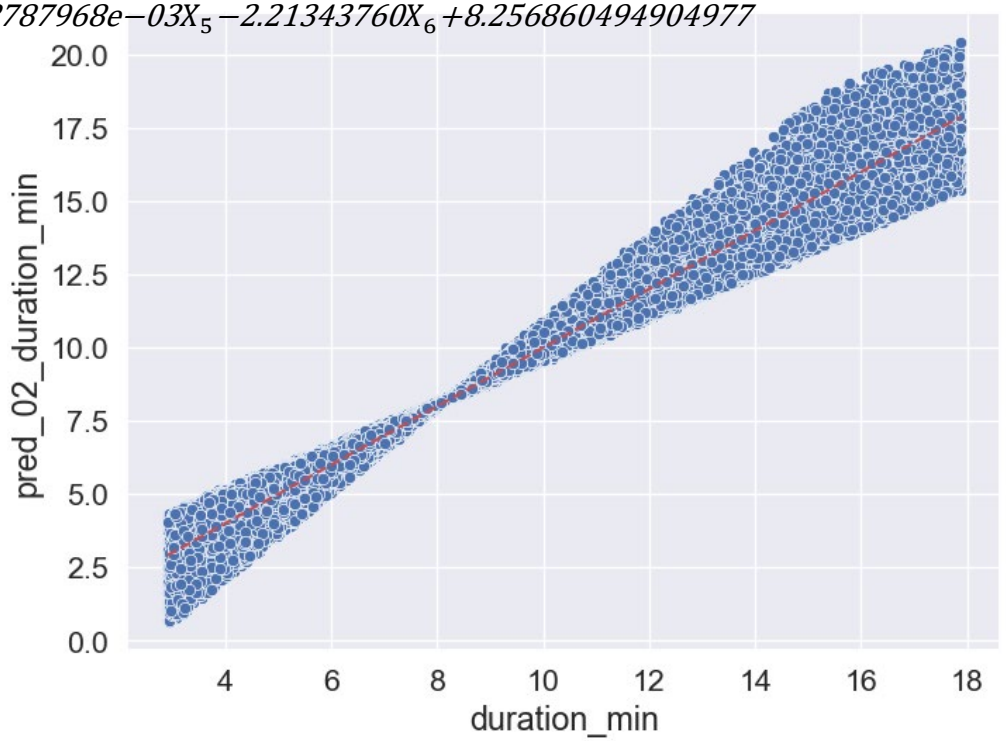
Predict

duration_min vs. trip_distance

The empirical equation is

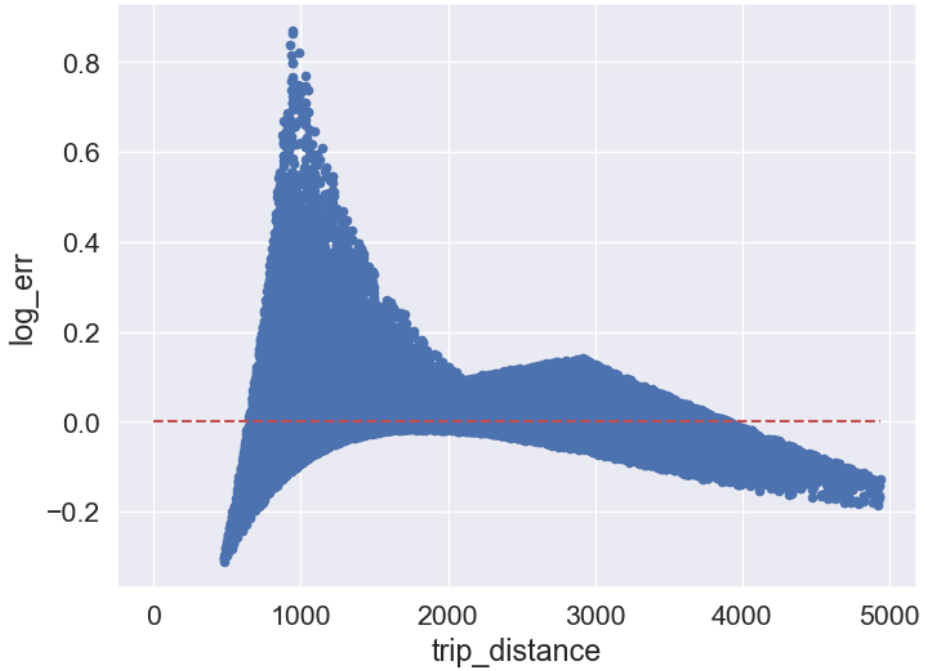
$$y = 4.54056103e{-}03X_1 + -1.38787968e{-}03X_5 - 2.21343760X_6 + 8.256860494904977$$



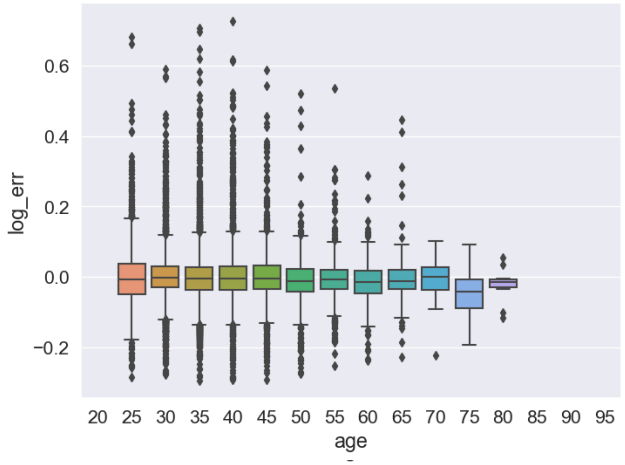


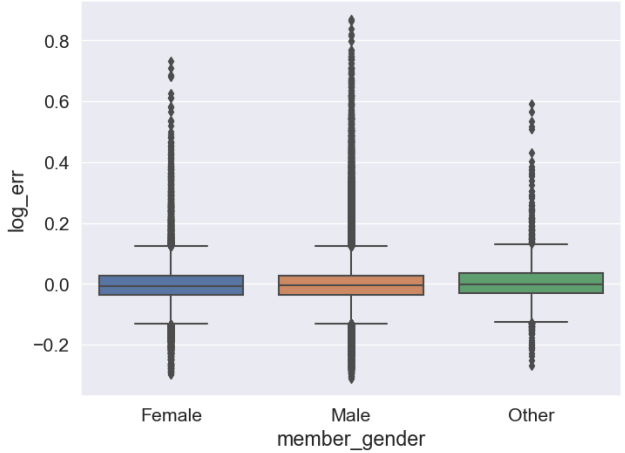
Residual

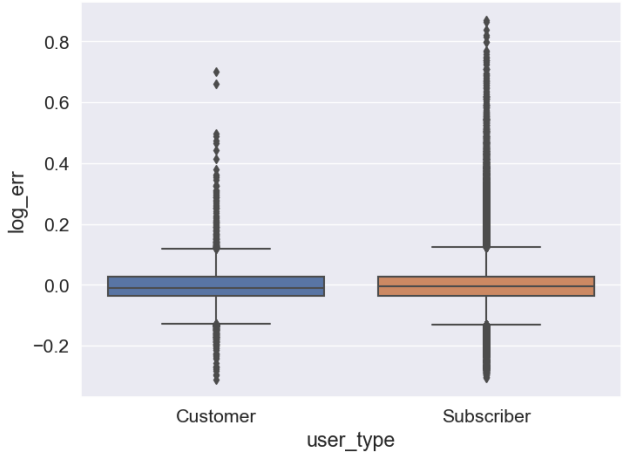
Exploration `log errors` vs. `target_value`

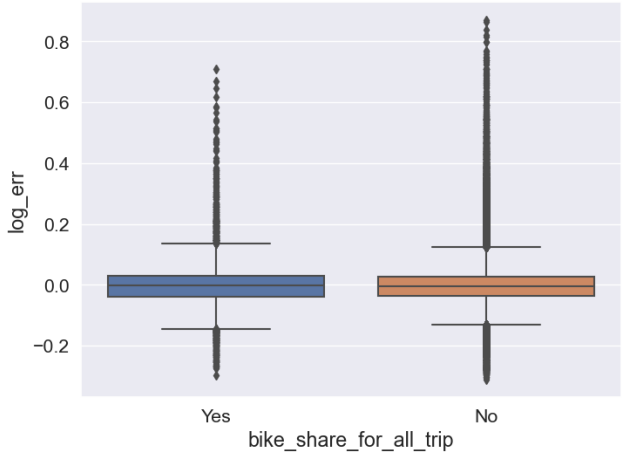


Exploration additional relationships for potential improvement of tuning

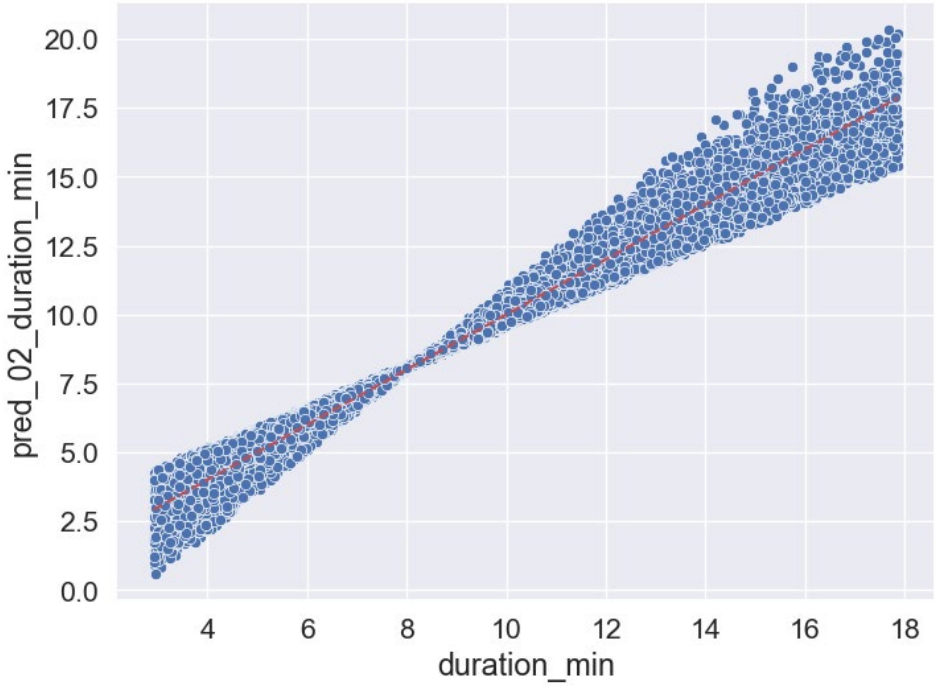


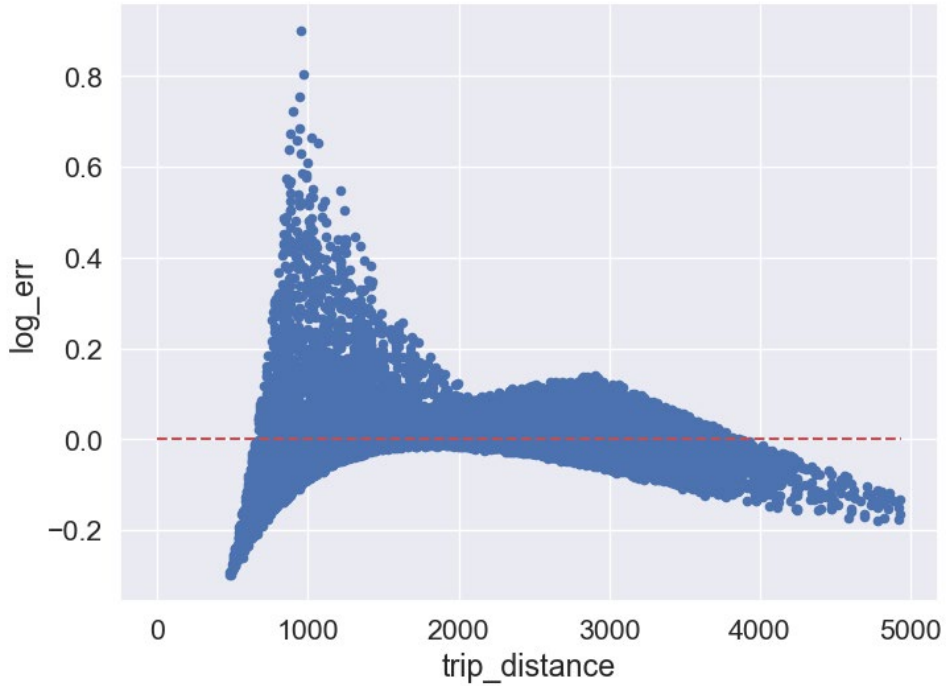






Validate the model





Testing the model