CS6923, Machine Learning
Assignment #5

Submitted By :
Rakshit Sareen (rs5606)

Problem Definition

The project is about predicting the delay of a flight. This is a regression problem and we can use many regression algorithms to predict the delay in the flight.
There are many choices that can be used for this purpose, like K-NN, Decision Trees, Random Forest, Gradient Boosted Trees (Ensemble Methods), Linear Regression, Neural Networks etc.
I decided to work with four algorithms: Linear Regression, Decision Trees, Random Forest Regressor and Gradient Boosting Regressor.

Data Exploration

This section presents some of the many data explorations performed in the assignment, as can be seen in the jupyter notebook.

ACTUAL_ELAPSED_TIME
Numeric

Distinct count	377
Unique (%)	0.0%
Missing (%)	0.0%
Missing (n)	0
Infinite (%)	0.0%
Infinite (n)	0

Mean	140.87
Minimum	31
Maximum	646
Zeros (%)	0.0%

Toggle details

Statistics

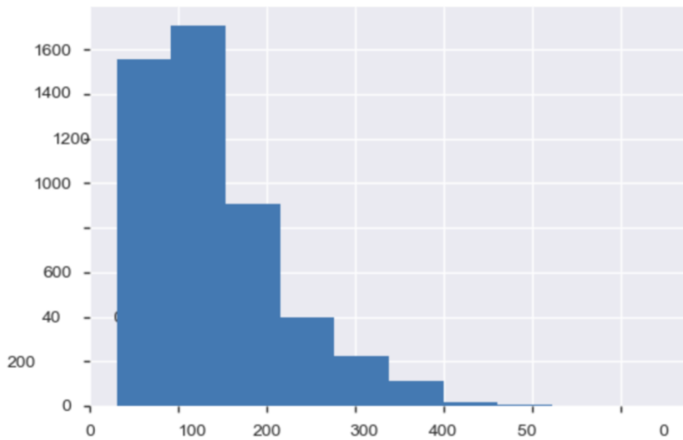
Quantile statistics

Minimum	31
5-th percentile	55
Q1	83
Median	123
Q3	175
95-th percentile	303
Maximum	646
Range	615
Interquartile range	92

Descriptive statistics

Standard deviation	76.23
Coef of variation	0.54115
Kurtosis	1.7911
Mean	140.87
MAD	58.745
Skewness	1.2969
Sum	691798
Variance	5811.1
Memory size	38.4 KiB

Histogram



Common Values

Value	Count	Frequency (%)
77	52	0.0%
74	50	0.0%
80	47	0.0%
75	47	0.0%
79	45	0.0%
68	44	0.0%
93	44	0.0%
78	43	0.0%
101	42	0.0%
89	41	0.0%
Other values (367)	4456	0.0%

Extreme Values

Minimum 5 values

Value	Count	Frequency (%)
31	1	0.0%
32	1	0.0%
33	1	0.0%
35	7	0.0%
36	4	0.0%

Maximum 5 values

Value	Count	Frequency (%)
495	1	0.0%
504	1	0.0%
516	1	0.0%
560	1	0.0%
646	1	0.0%

Feature Engineering

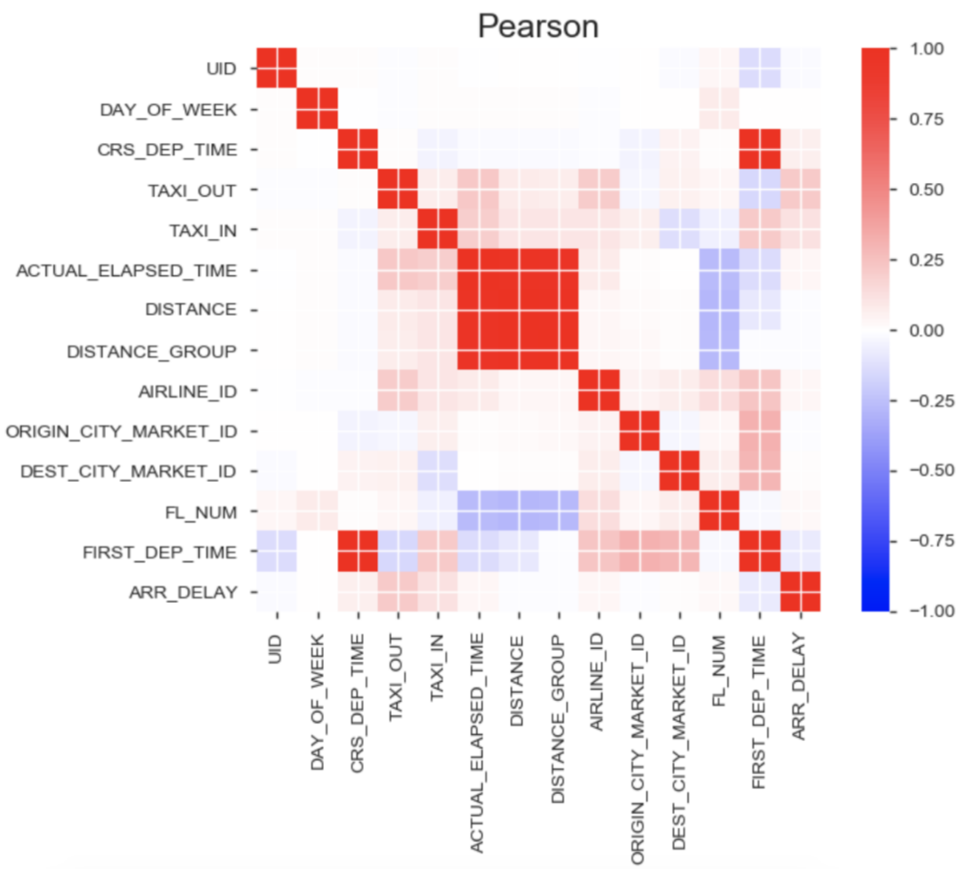
Feature Dropping

I decided to explore the features and get to know more about the data. I found out various features that are not relevant to the task at hand. Some of the features were highly correlated and some had too much cardinality. I decided to drop those.
Here is a table of the features I dropped and the reason for it.

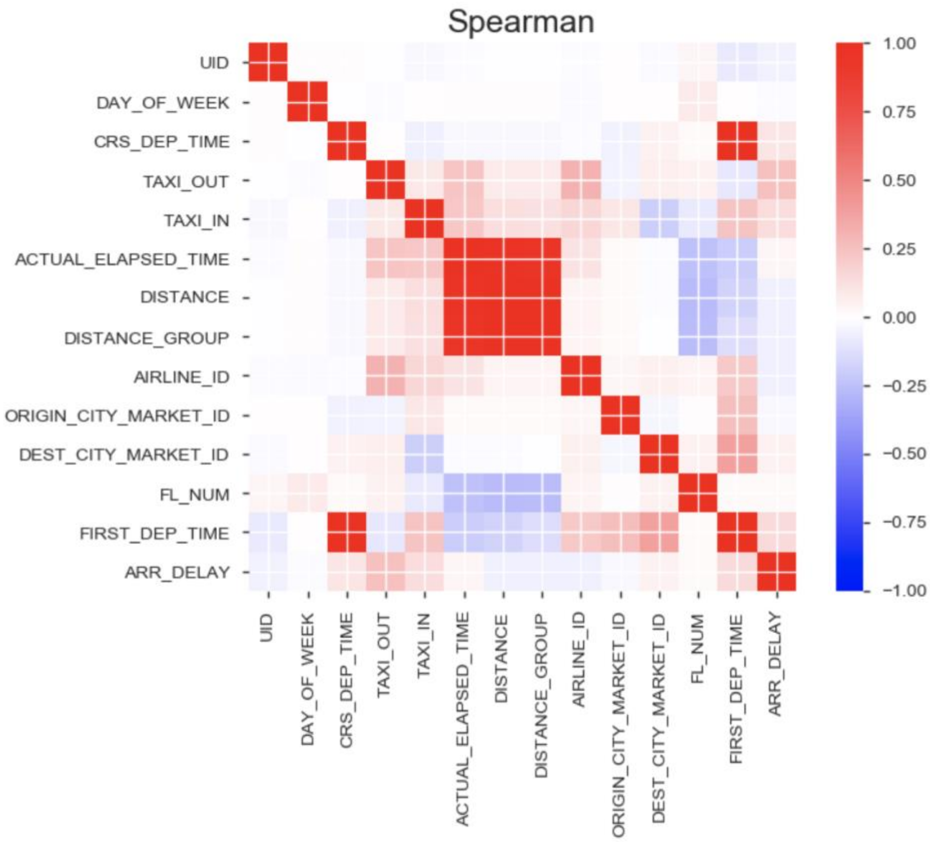
Feature	Reason of Dropping
'ORIGIN_STATE_ABR'	High Cardinality
'UNIQUE_CARRIER'	Same as AIRLINE_ID - High Correlation
'DEST_STATE_ABR'	High Cardinality
'UID'	Just a unique value, Not important to analysis
'DEST'	High Cardinality
'DEST_CITY_NAME'	High Cardinality
'DISTANCE_GROUP'	Highly correlated with DISTANCE
'FIRST_DEP_TIME'	Missing Values for almost all of the data
'FL_DATE'	High Cardinality
'ORIGIN'	High Cardinality
'ORIGIN_CITY_NAME'	High Cardinality
'ACTUAL_ELAPSED_TIME'	Highly correlated with DISTANCE
'CRS_DEP_TIME'	Highly correlated with FIRST_DEP_TIME
'FL_NUM'	High Cardinality

- `DEST` has a high cardinality: 228 distinct values Warning
- `DEST_CITY_NAME` has a high cardinality: 224 distinct values Warning
- `DISTANCE` is highly correlated with `ACTUAL_ELAPSED_TIME` ($\rho = 0.97193$) Rejected
- `DISTANCE_GROUP` is highly correlated with `DISTANCE` ($\rho = 0.98868$) Rejected
- `FIRST_DEP_TIME` is highly correlated with `CRS_DEP_TIME` ($\rho = 0.9976$) Rejected
- `FL_DATE` has a high cardinality: 365 distinct values Warning
- `ORIGIN` has a high cardinality: 239 distinct values Warning
- `ORIGIN_CITY_NAME` has a high cardinality: 235 distinct values Warning

I tried to find out the correlation of features among themselves.
Here I present two correlation matrices, Pearson and Spearman.



Pearson
UID
DAY_OF_WEEK
CRS_DEP_TIME
TAXI_OUT
TAXI_IN
ACTUAL_ELAPSED_TIME
DISTANCE
DISTANCE_GROUP
AIRLINE_ID
ORIGIN_CITY_MARKET_ID
DEST_CITY_MARKET_ID
FL_NUM
FIRST_DEP_TIME
ARR_DELAY
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00



Feature Creation

Creation of new features is an important part of data analysis and exploration. We can add missing data, augment data with new columns , extract meaning from different features and combine them into a new feature.
Below table presents what features have been created and their description.

Feature	Description
Day	The day of the month
Month	The month in the
SPEED	The speed of the airplane in the journey
IS_HOLIDAY	1,0 based on whether FL_DATE was holiday
IS_WEEKEND	1,0 based on whether FL_DATE was a weekend

Feature Encoding

I decided to encode the AIRLINE_ID. I used one-hot encoding provided by the pandas library.
The reason to use one-hot encoding instead of integer encoding is because AIRLINE_ID does not have any ordered relationship with each other.

Also, since AIRLINE_ID did not have much cardinality, I decided to use one-hot encoding to it.

GRID SEARCH CV (CROSS VALIDATION = 10)

I performed grid search over the hyperparameters for each of the chosen models. The parameters chosen were adopted by Cross Validation with cv = 10.
Here are the parameters used for all the algorithms and their values:

Linear Regression

Parameter	Description	Values
fit_intercept	Whether to calculate the intercept for this model	True, False
normalize	X will be normalized before regression	True, False

Grid Search CV provides the best score and the best parameters it found.

Result :

best_score_ : 1940.90185776
best_params_ :{'normalize': False, 'fit_intercept': True}

Decision Trees

Parameter	Description	Values
splitter	Strategy used at each split of the node	best ,random
max_features	The number of features to consider when looking for the best split	auto, sqrt, log2
min_samples_split	The minimum number of samples required to split an internal node	2,4,8
max_depth	The maximum depth of the tree	2,4,6,8,10
presort	Presort the data to speed up the finding of best splits in fitting	True

Result :

best_score_ : 1999.59277975
best_params_ : {'max_features': 'auto', 'min_samples_split': 2, 'presort': True, 'max_depth': 2, 'splitter': 'random'}

Random Forest

Parameter	Description	Values
n_estimators	The number of trees in the forest.	10,20,30,40,50,60
max_features	The number of features to consider when looking for the best split	auto, sqrt, log2
min_samples_split	The minimum number of samples required to split an internal node	2,4,8
bootstrap	Whether bootstrap samples are used when building trees.	True, False
warm_start	When set to True, reuse the solution of the previous call to fit and add more estimators to the ensemble, otherwise, just fit a whole new forest.	True,False

Result :

best_score_ : 1985.79949748
best_params_ : {'max_features': 'log2', 'min_samples_split': 8, 'bootstrap': True, 'n_estimators': 60, 'warm_start': True}

Gradient Boosting

Parameter	Description	Values
n_estimators	The number of trees in the forest.	10,20,30,40,50,60
max_features	The number of features to consider when looking for the best split	auto, sqrt, log2
min_samples_split	The minimum number of samples required to split an internal node	2,4,8
learning_rate	learning rate shrinks the contribution of each tree by learning_rate.	True, False
warm_start	When set to True, reuse the solution of the previous call to fit and add more estimators to the ensemble, otherwise, just fit a whole new forest.	True,False

Result :

best_score_ : 1943.52304248
best_params_ : {'warm_start': False, 'loss': 'ls', 'learning_rate': 0.2, 'n_estimators': 40, 'min_samples_split': 2, 'max_features': 'auto'}