ML-Report Assignment 1

Aidar Gumerov

September 23, 2021

1 Data Preprocessing and Visualization

The initial dataset has 2 categorical features (airport names) and 2 features associated with arrival and departure times, given in string format. Target - delay. Thus on a task based on data from 2015-2017 to predict the delay in 2018. So this problem can be viewed as a regression problem. To solve this problem, I used the studied models: linear regression (with and without regularization), polynomial regularization.

First, I checked the dataset for empty cells. Making sure that everything is ok, went to the encoding airoport names with LabelEncoder and convert a timeline into several new features: year, month, day of the month, week, day, hour, minutes. I think it is useful, there is a dependence on the time at which the departure takes place - because the load of airports is different at different times of the year and days of the month. And added a new feature FlyDuration (arrival - departure).

The next step I used PCA to check that the main informative vector is FlyDuration. And you can really be convinced of this by getting the graph: On another graph, you can see that there is no

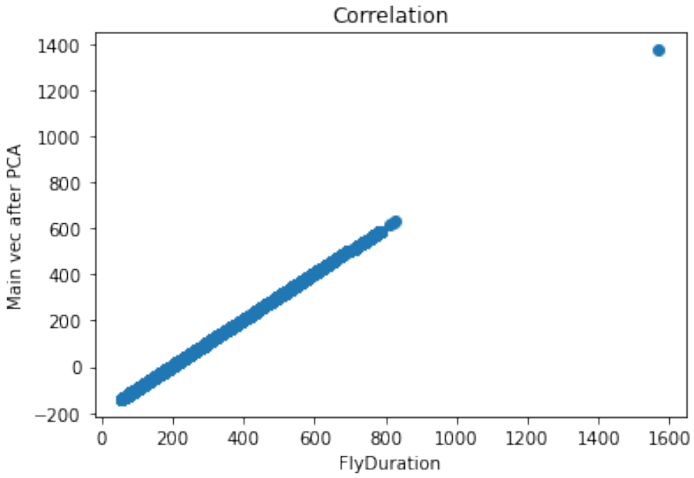

Figure 1: strong linear dependence with an offset of 200

specific distribution of data and it is difficult to understand the dependence between the data. As we can see when first run the model, the rest of the features can be ignored because the results using FLy Duration are better.

MSE with all data: 1644.5433782686932

MAE with all data: 16.16071286562617

MSE using just FlyDuration: 1619.2068848222061

MAE using just FlyDuration: 14.401651841266302

And finally I divide my data into train and test and use a mask in pandas and looked at the delay histogram

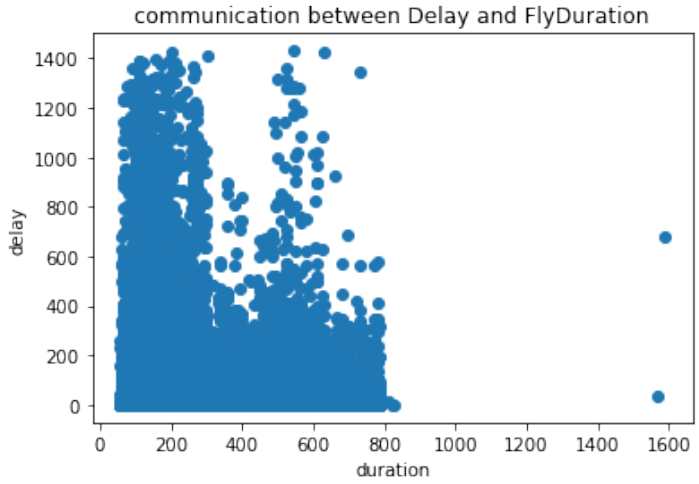

Figure 2: it can be seen that some points are very long delay

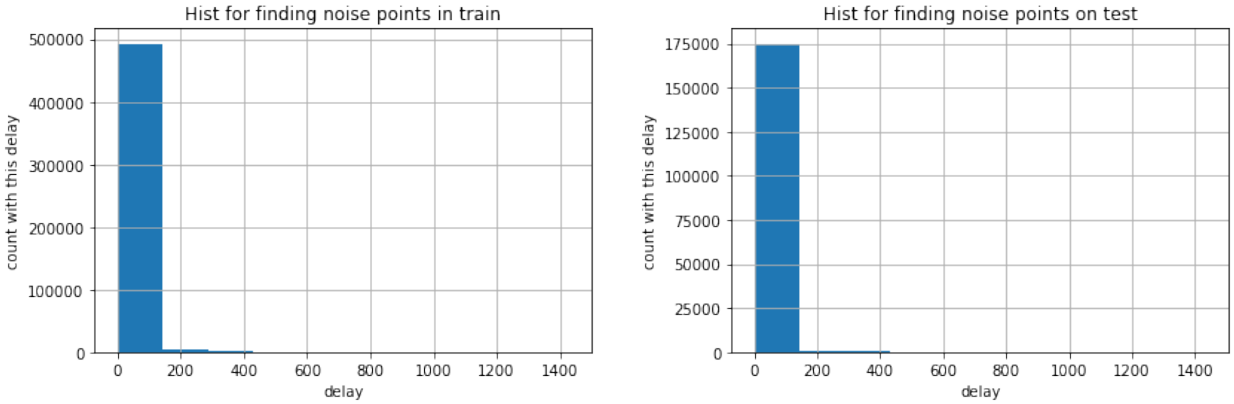

Figure 3: Hists

2 Outlier Detection and Removal

The histogram shows that basically all points are less than 200, which means that the rest can be considered as an anomaly. such points can give a big error when training the model, which means you need to get rid of them. For this I wrote a small function detect-noise () that takes a dataframe and a possible delay interval and gives an list of indices in the dataframe that are greater than this interval + average delay. Further, I consider this interval as a hyperameter - by changing it I will monitor the errors and choose the best one. After that, I can drop by the index of the row from the dataframe and the next step will be training and testing the model.

3 Models and Performance Measurement

Tables 1-4 shows the results with training data from interval 50. To measure the quality of the model, used the R2, MAE, MSE metrics.

Linear Regression model

Degrees for polynominal - (1, 2, 5 , 8 ,10)

Lasso and Ridge regularization.

As we can see our models generally predict a very limited range of delay(Figure 4). Thus, all noisy points that lie well above the predicted curve can significantly degrade the metric. Therefore, in the next step, I selected the optimal data interval for which I will filter out noisy points (using created function detect noise) on the training data and then train the model.(Figure 5)

Polynomial degrees	MAE for Lasso	MAE for Ridge	MAE for Linear Regression	MSE for Lasso
1	6.460904	6.460456	6.460456	98.84333
2	6.460804	6.460156	6.460156	98.84245
5	6.457516	6.457415	6.457416	98.79641
8	6.457779	6.456212	6.4573	98.78987
10	6.457645	6.456129	6.467572	98.78682

Table 1: Errors in different models on the Train

Polynomial degrees	MSE for Ridge	MSE for Linear Regression	R^2 for Lasso	R^2 for Ridge	R^2 for Linear Regression
1	98.84326	98.84326	0.019194	0.019195	0.019195
2	98.84154	98.84154	0.019203	0.019212	0.019212
5	98.7889	98.7889	0.01966	0.019734	0.019734
8	98.76719	98.77526	0.019725	0.01995	0.01987
10	98.75805	98.89055	0.019755	0.020041	0.018726

Table 2: Errors in different models on the Train

Polynomial degrees	MAE for Lasso	MAE for Ridge	MAE for Linear Regrtession	MSE for Lasso
1	9.77716	9.776607	9.776607	1600.453
2	9.777286	9.777012	9.777012	1600.465
5	9.774178	9.775553	9.775554	1600.187
8	9.774514	9.776162	9.77551	1600.102
10	9.774697	9.776332	9.781888	1600.107

Table 3: Errors in different models on the Test

olynomial degrees	MSE for Ridge	MSE for Linear Regression	R^2 for Lasso	R^2 for Ridge	R^2 for Linear Regression
1	1600.439	1600.439	0.001447	0.001455	0.001455
2	1600.483	1600.483	0.001439	0.001428	0.001428
5	1600.154	1600.154	0.001612	0.001633	0.001633
8	1600.242	1600.217	0.001665	0.001578	0.001594
10	1600.204	1599.979	0.001662	0.001602	0.001742

Table 4: Errors in different models on the Test

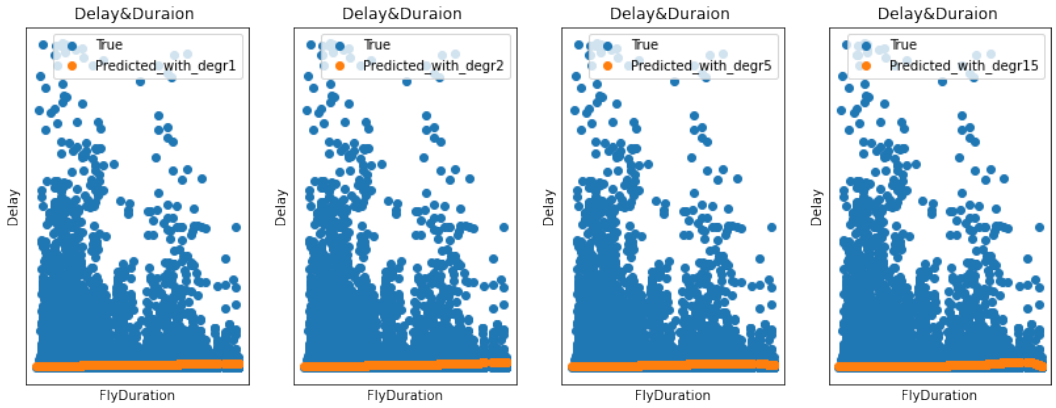

Figure 4: test vs predicted results on test with different poly. degr.

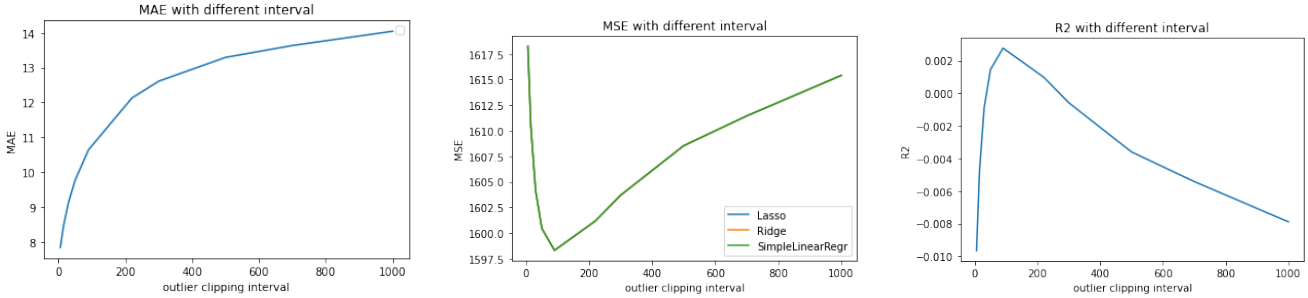

Figure 5: Dependence of errors on the test on the interval of dropped points on the train

From the graph built for each metric, it follows that the optimal interval is 90, that is, points from 0 to (90 + the average delay) for training are used.

Since the test data also contains anamous values, I will take into account, for example, only those that are less than 100 minutes (this is 99% of the test data) delays (we do not consider the rest of the metrics). And finally, it remains to evaluate the model on this data

3.1 Summary.

Following the received metrics and graphs the linear model is too simple for a more accurate solution of this problem. Also adding polymorphism doesn't save the situation. Perhaps this is due to the fact that much more than 90% of the planes arrive on time and on more belated flights the model gives a big error. as a solution I cleared the training data with delay more than 90 (best interval). Also, when calculating metrics, I did not take into account noisy points in the test data (1% of all test data was not considered in the final case (Table 5-6)) As a result, the model showed the best quality, but still with a rather large error - perhaps there are no important features (for example, weather conditions) that contribute but are not contained in this dataset.

Polynomial degrees	MAE for Lasso	MAE for Ridge	MAE for Linear Regression	MSE for Lasso
1	6.856896	6.856414	6.856414	77.62618
2	6.856808	6.856593	6.856593	77.6227
5	6.85375	6.85626	6.856255	77.61767
8	6.854337	6.858309	6.855542	77.6117
10	6.854594	6.858936	6.86228	77.61513

Table 5: Errors in different models on the Test after clear data

Polynomial degrees	MSE for Ridge	MSE for Linear Regression	R^2 for Lasso	R^2 for Ridge	R^2 for Linear Regression
1	77.63465	77.63465	-0.22772	-0.22786	-0.22786
2	77.64307	77.64307	-0.22767	-0.22799	-0.22799
5	77.64497	77.64503	-0.22759	-0.22802	-0.22802
8	77.69507	77.6539	-0.22749	-0.22881	-0.22816
10	77.69701	77.63053	-0.22755	-0.22884	-0.22779

Table 6: Errors in different models on the Test after clear data