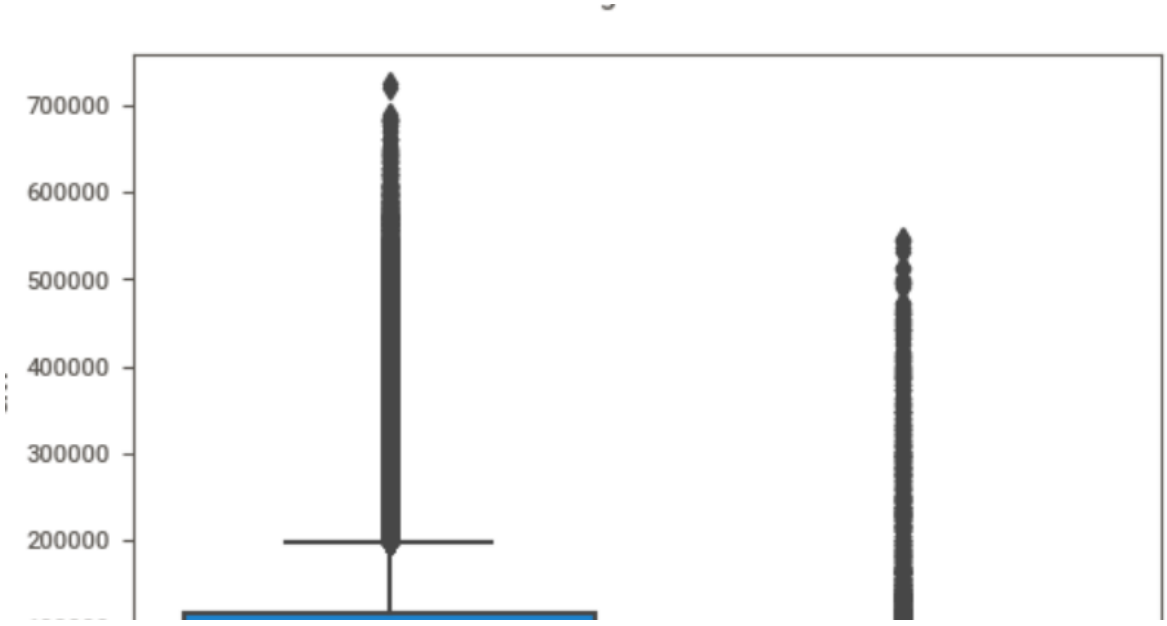
A brief on the approach used to solve the problem.

As we need to predict the continuous value, I used the lightgbm model to do the analyse as it very deep decision trees, which allows it to capture non-linear interactions in the data

Which Data-preprocessing / Feature Engineering ideas really worked? How did you discover them?

Data-preprocessing Steps:

1. As I selected the linear regression model , so need to convert categorical variables into numerical by one hot-encoding
2. Scaled all numerical column to min-max scaling as there are lots of outlier are present in the model



What does your final model look like? How did you reach it?

- I select the feature of the model using RFE, which give me top 10 best feature to work on it.
- Then I used VIF to detect multicollinearity in the dataset.

From RFE and VIF, I selected the feature which need not to train in pycaret. But, that doesn't perform well so we take our all feature. From I select 4 models based on R2_score, MAE score .

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
gbr	Gradient Boosting Regressor	50683.1482	6.949467e+09	83348.3432	0.1586	0.5672	0.5493	5.075
lightgbm	Light Gradient Boosting Machine	50735.4112	6.984592e+09	83558.8401	0.1543	0.5684	0.5495	0.464
llar	Lasso Least Angle Regression	51708.3661	7.014188e+09	83735.0376	0.1508	0.5849	0.5698	0.073
lr	Linear Regression	51795.0314	7.014484e+09	83736.8542	0.1507	0.5869	0.5713	0.455
lasso	Lasso Regression	51792.0144	7.014427e+09	83736.5075	0.1507	0.5869	0.5713	0.641
ridge	Ridge Regression	51792.6270	7.014439e+09	83736.5822	0.1507	0.5869	0.5713	0.060
lar	Least Angle Regression	51792.9841	7.014440e+09	83736.5874	0.1507	0.5869	0.5713	0.089

Then on based on their test data preformation, I decided selected best model.