**Improvement in terms of Feature Engineering**

Chart, histogram

Description automatically generated Chart, histogram

Description automatically generated

If we look at the distribution plot of the numeric data columns (figures above) namely the floor\_area\_sqm and age, they do not share the same scale and without standardization, the variable with the larger range will be more domine during regression. Next, both plots have vastly different ranges of value and hence, normalization is required to minimize the inherent biases toward data with larger values during regression.

Actions taken: fit then transform the numeric datasets with StandardScaler() and Normalizer() class from SKLearn

    (Pipeline([

        ('scaler', StandardScaler()),

        ('normalize', Normalizer()),

    ]),

     make\_column\_selector(dtype\_include=np.number)),

**Improvement in terms of Modeling**

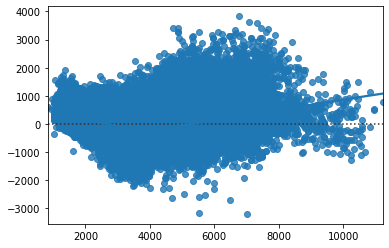
For our improvement, we analyzed the P-value of each feature from our original linear regression result. We noticed there are 5 features with P-value > 0.05. This suggests that these features have little to no influence on the outcome of the prediction. Using filtering and mapping functions from the python library, we removed these features and re-fit the regression model. On top of this, we went one step future to examine the performance of the decision tree and random forest regression model.

We compared the mean squared error and r2\_score between our original linear regression model with the new linear regression model, decision tree regression model, and random forest regression model.

|  |  |  |
| --- | --- | --- |
| Model | Mean square error | R2\_score |
| Original Regression Model | 249307.39 | 0.8578 |
| Improvised Regression Model with feature selection | 264329.86 | 0.8866 |
| Decision Tree Model (max depth=2) | 1456782.7736734557 | 0.16940572619690164 |
| Decision Tree Model (max depth=5) | 1103550.5746500914 | 0.3708033862556195 |
| Decision Tree Model (max depth=8) | 855868.7588298644 | 0.5120208015512975 |
| Random Forest Model (max depth=2) | 1452489.8983405593 | 0.17185333728490415 |
| Random Forest Model (max depth=5) | 1088995.8077608538 | 0.3791018822656015 |
| Random Forest Model (max depth=8) | 830615.3229060769 | 0.5264192140333861 |

From the result, we can see that the mean square error and R2 score get better with a high max depth value used for both the Decision tree and Random Forest. However, high computing power is required to solve the model for a larger max depth value. Therefore, we have decided not to persuade deeper with these 2 models.

Despite our best effort to improve our original model, the new model still falls behind by a small margin. The Kaggle score for the new model is 0.88603 vs the 0.89386 of the original model. Looking at the residual plot of the predicted resale price vs actual resale price found using the original model, there are hints of heteroscedasticity suggested by the non-zero mean line (Figure Below), meaning that the residuals get larger as the prediction moves from small to large (or from large to small).



One potential reason that causes this is missing regression variables which could be caused by the non-linear relationship between variables and the resale price which calls for further investigation.