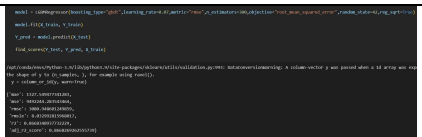



## Project Development Phase Model Performance Test

Date	18 November 2022
Team ID	PNT2022TMID25594
Project Name	Project – Car Resale Value Prediction
Maximum Marks	10 Marks

### Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Metrics	<b>Regression Model: LGBM Regressor</b>  MAE: 1327.56 MSE: 9492244.25 RMSE: 3080.93 RMSLE: 8.05 R2 Score: 0.8664 Adjusted R2 Score: 0.8666	 <pre> model = LGBMRegressor() model.fit(X_train, y_train) y_pred = model.predict(X_test) mse = mean_squared_error(y_test, y_pred) rmse = np.sqrt(mse) mae = mean_absolute_error(y_test, y_pred) r2 = r2_score(y_test, y_pred) adj_r2 = adjusted_r2_score(y_test, y_pred, n_features=X_test.shape[1]) print('MSE: %.2f, RMSE: %.2f, MAE: %.2f, R2: %.4f, Adjusted R2: %.4f' % (mse, rmse, mae, r2, adj_r2)) </pre>
2.	Tune the Model	<b>Hyperparameter Tuning</b> 1) Learning Rate: [0.01, 0.03, 0.05, 0.07] 2) Boosting Type: ['gbdt', 'dart', 'goss', 'rf'] 3) Number of Estimators: [100, 200, 300]  <b>Validation Method:</b> Grid Search Cross Validation  <b>Best Parameters:</b> Learning Rate – 0.07 Boosting Type – 'gbdt' Number of Estimators - 300	 <pre> lgbm_configs = {     "name": "LGBMRegressor",     "method": "grid",     "metric": {         "name": "adj_r2",         "goal": "maximize"     },     "parameters": {         "learning_rate": {             "values": [0.01, 0.03, 0.05, 0.07]         },         "objective": {             "values": ["root_mean_squared_error"]         },         "boosting_type": {             "values": ["gbdt", "dart", "goss", "rf"]         },         "reg_sqrt": {             "values": [True]         },         "metric": {             "values": ["rmse"]         },         "n_estimators": {             "values": [100, 200, 300]         },         "random_state": {             "values": [42]         }     } } </pre>

### Screenshots

## 1) Metrics

```
model = LGBMRegressor(boosting_type="gbdt", learning_rate=0.07, metric="rmse", n_estimators=300, objective="root_mean_squared_error", random_state=42, reg_sqrt=True)

model.fit(X_train, Y_train)

Y_pred = model.predict(X_test)

find_scores(Y_test, Y_pred, X_train)

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please use the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)

{'mae': 1327.549477341283,
 'mse': 9492244.283543464,
 'rmse': 3080.948601249859,
 'rmsele': 8.032992815968017,
 'r2': 0.8668348937732229,
 'adj_r2_score': 0.8668269262555739}
```

## 2) Tune the model

```
lgbm_configs = {
    "name": 'LGBMRegressor',
    "method": "grid",
    "metric": {
        "name": "adj_r2",
        "goal": "maximize"
    },
    "parameters": {
        "learning_rate": {
            "values": [0.01, 0.03, 0.05, 0.07]
        },
        "objective": {
            "values": ['root_mean_squared_error']
        },
        "boosting_type": {
            "values": ['gbdt', 'dart', 'goss', 'rf']
        },
        "reg_sqrt": {
            "values": [True]
        },
        "metric": {
            "values": ['rmse']
        },
        "n_estimators": {
            "values": [100, 200, 300]
        },
        "random_state": {
            "values": [42]
        }
    }
}
```

Wandb Sweep :

