



Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID44773
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	Regression Model: LGBM Regressor MAE: 1327.55 MSE: 9492244.28 RMSE: 3080.95 RMSLE: 8.03 R2 Score: 0.8668 Adjusted R2 Score: 0.8668	 <pre> model = LGBMRegressor(boosting_type='gbdt', learning_rate=0.01, num_estimators=300, objective='l2', n_jobs=-1, random_state=42, reg_alpha=0.001, reg_lambda=0.001) model.fit(X_train, Y_train) # Predict on test set Y_pred = model.predict(X_test) # Calculate metrics MAE = mean_absolute_error(Y_test, Y_pred) MSE = mean_squared_error(Y_test, Y_pred) RMSE = sqrt(MSE) RMSLE = sqrt(mean_squared_log_error(Y_test, Y_pred)) R2 = r2_score(Y_test, Y_pred) AdjR2 = adjusted_r2_score(Y_test, Y_pred) print(f'MAE: {MAE}, MSE: {MSE}, RMSE: {RMSE}, RMSLE: {RMSLE}, R2 Score: {R2}, Adjusted R2 Score: {AdjR2}') </pre>
2.	Tune the Model	Hyperparameter Tuning 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] Validation Method: Grid Search Cross Validation Best Parameters: 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 change 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]
        }
    }
}
```

3) Wandb sweep:

