|  |  |
| --- | --- |
| DATE | 16-11-2022 |
| TEAM ID | PNT2022TMID47297 |
| PROJECT | CAR RESALE VALUE PREDICTION |
| ASSIGNED TASKS | BUILDING MODELS |

**CHOOSE THE APPROPRIATE MODEL**

In [103]:

*# Linear Regression*

**from** sklearn.linear\_model **import** LinearRegression

lr **=** LinearRegression()

lr**.**fit(x\_train, y\_train)

Out[103]:

LinearRegression()

In [104]:

*# Lasso Regression*

**from** sklearn.linear\_model **import** Lasso

lasso **=** Lasso(alpha**=**0.01, normalize**=True**)

lasso**.**fit(x\_train, y\_train)

Out[104]:

Lasso(alpha=0.01, normalize=True)

In [105]:

*# Ridge Regression*

**from** sklearn.linear\_model **import** Ridge

ridge **=** Ridge(alpha**=**0.01, normalize**=True**)

ridge**.**fit(x\_train, y\_train)

Out[105]:

Ridge(alpha=0.01, normalize=True)

In [106]:

*# Decision Tree*

**from** sklearn.tree **import** DecisionTreeRegressor

DT **=** DecisionTreeRegressor()

DT**.**fit(x\_train, y\_train)

Out[106]:

DecisionTreeRegressor()

In [107]:

*# KNN*

**from** sklearn.neighbors **import** KNeighborsRegressor

knn **=** KNeighborsRegressor()

knn**.**fit(x\_train, y\_train)

Out[107]:

KNeighborsRegressor()

In [122]:

**from** sklearn.ensemble **import** RandomForestRegressor

RF **=** RandomForestRegressor()

RF**.**fit(x\_train, y\_train)

Out[122]:

RandomForestRegressor()

**Checking the Metrics of the models**

In [110]:

*# Linear Regression*

lr**.**score(x\_test, y\_test)

Out[110]:

0.38523564993870363

In [111]:

**from** sklearn.metrics **import** mean\_squared\_error

np**.**sqrt(mean\_squared\_error(y\_test,lr**.**predict(x\_test)))

Out[111]:

3541.5061860257147

In [112]:

*# Lasso Regression*

lasso**.**score(x\_test, y\_test)

Out[112]:

0.38526963129719227

In [113]:

np**.**sqrt(mean\_squared\_error(y\_test,lasso**.**predict(x\_test)))

Out[113]:

3541.408305549754

In [114]:

*# Ridge Regression*

ridge**.**score(x\_test, y\_test)

Out[114]:

0.3852402445126166

In [115]:

np**.**sqrt(mean\_squared\_error(y\_test,ridge**.**predict(x\_test)))

Out[115]:

3541.492951895974

In [116]:

*# K Nearest Neighbour*

knn**.**score(x\_test, y\_test)

Out[116]:

0.36606715277167357

In [117]:

np**.**sqrt(mean\_squared\_error(y\_test,knn**.**predict(x\_test)))

Out[117]:

3596.294881454182

In [118]:

*# Decision Tree*

DT**.**score(x\_test, y\_test)

Out[118]:

0.7361183941834686

In [119]:

np**.**sqrt(mean\_squared\_error(y\_test,DT**.**predict(x\_test)))

Out[119]:

2320.268405314096

In [123]:

RF**.**score(x\_test, y\_test)

Out[123]:

0.861339598980134

**Saving the Model**

In [121]:

**import** pickle

pickle**.**dump(RF, open('Car Resale Value Prediction.pkl', 'wb'))