ADVANCED REGRESSION – HOUSE PRICE PREDICTION

Sabyasachi Parida

Question 1. What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

Answer.

The following are the optimal alpha value for ridge and Lasso regression:

Ridge: {'alpha': 6.0}, **Lasso**: {'alpha': 0.001}

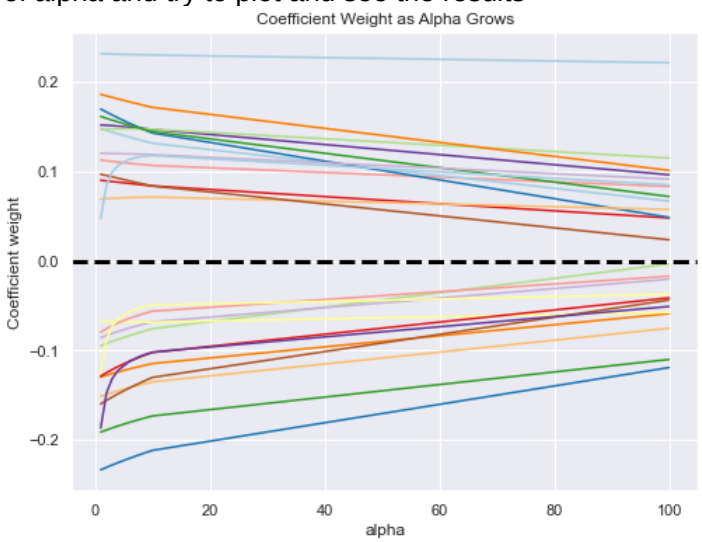
When we double the alpha value

Ridge(alpha=12.0)
Coefficients: array ([0.22984049, -0.20748762, -0.07214405, -0.1698515 , 0.10571695,
0.0825254 , -0.13212512, -0.11178751, -0.06508796, 0.14566952,
-0.06842696, -0.12482807, 0.12854195, 0.13806802, 0.14690312,
0.14122878, -0.05289908, -0.09803391, 0.07170445, 0.16873014,
0.11837107, -0.09840508, -0.04744768, 0.0811038 , 0.11825339])

Results with Optimum alpha

Ridge(alpha=6.0)
Coefficients: array ([0.23070132, -0.22046242, -0.08337824, -0.18031611, 0.10916297,
0.08665453, -0.14149284, -0.12063854, -0.07484314, 0.14955021,
-0.06797865, -0.14182138, 0.13840047, 0.15391857, 0.14767195,
0.15162393, -0.06405571, -0.11202671, 0.0708196 , 0.17782161,
0.11962403, -0.11213092, -0.05623806, 0.08950852, 0.11370932])

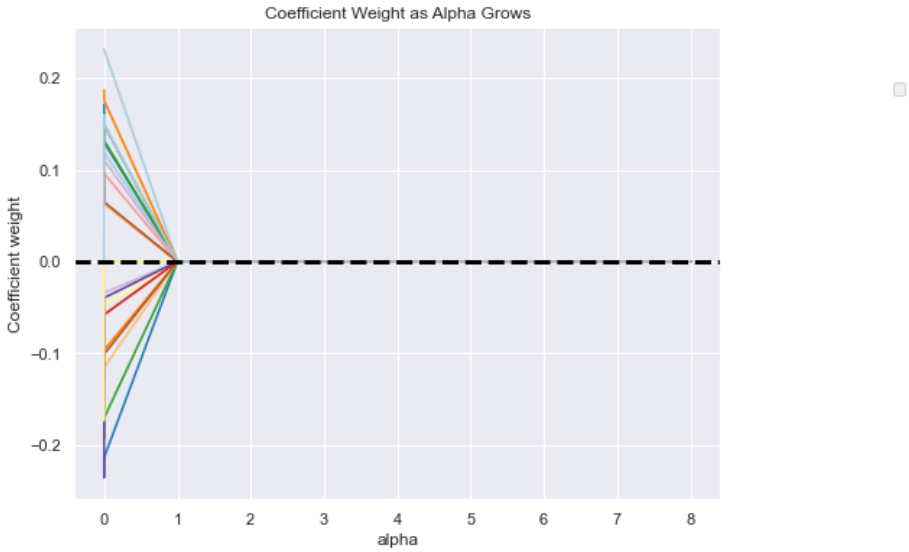
When we use few values of alpha and try to plot and see the results



With increase in alpha value the coefficients have reduced. (i.e they are shrinking towards 0 but doesn't become 0).

Lasso(alpha=0.002)
Coefficients: array([0.23056639, -0.21314761, -0.05729641, -0.17008439, 0.09585484,
0.06350338, -0.11508384, -0.09623763, -0.03400889, 0.14860862,
-0.04704945, -0.10084595, 0.11885007, 0.12945716, 0.1498428 ,
0.13184501, -0.   , -0.05787391, 0.06288153, 0.17521491,
0.11021065, -0.03943285, -0.   , 0.06482096, 0.14935464])

Lasso(alpha=0.001)
Coefficients : array([0.23109476, -0.22466475, -0.07716763, -0.18159198, 0.10469802,
0.07680192, -0.13399213, -0.11478324, -0.0606204 , 0.15109205,
-0.05721031, -0.13436748, 0.13513462, 0.15190742, 0.14895534,
0.14829622, -0.00980404, -0.09514468, 0.06531927, 0.1815542 ,
0.11525807, -0.05220999, -0.   , 0.08176525, 0.16029652])



In case of Lasso when we doubled the alpha value, we can observe that the coefficient values decreased. If we observe the Coefficients when alpha was 0.001 there was one coefficient which was 0. But when we increased the alpha to 0.002, we can see 2 coefficients which have become 0. From which we can conclude with increase in alpha the coefficients shrink towards zero.
(Detailed Code in the python notebook . ipynb file)

Question 2. You have determined the optimal value of lambda for ridge and lasso regression during the assignment. Now, which one will you choose to apply and why?

Answer.

After evaluating the model, we got the following results.

Ridge_R2_Train: 0.8292080946155551
Ridge_R2_Test: 0.8062497691406121

Lasso_R2_Train: 0.8274983862362728
Lasso_R2_Test: 0.8044082237912342

The R2 score of both ridge and Lasso are somewhat similar. But we will chose Lasso as In the lasso, this makes it easier for the coefficients to be zero(Lasso Regression adds “absolute value of magnitude” of

coefficient as penalty term to the loss function) and therefore easier to eliminate some of the input variable as they are not contributing to the output (In other words it helps in feature selection).

Question 3. After building the model, you realized that the five most important predictor variables in the lasso model are not available in the incoming data. You will now have to create another model excluding the five most important predictor variables. Which are the five most important predictor variables now?

Answer.

After excluding the top 5 most important predictor variables when we built the model the following are the results.((Detailed Code in the python notebook . ipynb file)

```
X_train_new = X_train.drop(Top5, axis=1)
X_test_new=X_test.drop(Top5, axis=1)
```

```
lasso_3 = Lasso(alpha=Lasso_alpha['alpha'])
lasso_3.fit(X_train_new, y_train)

# Predict
y_train_pred_3 = lasso_3.predict(X_train_new)
lasso_3_R2_Train = r2_score(y_true=y_train, y_pred=y_train_pred)

y_test_pred_3 = lasso_3.predict(X_test_new)
lasso_3_R2_Test =r2_score(y_true=y_test, y_pred=y_test_pred)

lasso_3_R2_Train
lasso_3_R2_Test
```

```
Lasso(alpha=0.001)

0.821250822556749

0.7997651569956378
```

```
lasso_coefficients_3 = pd.DataFrame({'Feature Importance_3':lasso_3.coef_},
                        index=X_train_new.columns)
lasso_coefficients_3.sort_values('Feature Importance_3', ascending=False)
```

	Feature Importance_3
OverallQual_Others	0.333718
OverallQual_8	0.327383
GarageCond_TA	0.294146
MSZoning_RL	0.250584
Neighborhood_Somerst	0.170898
MSSubClass_60	0.148480
LotConfig_CulDSac	0.104085
BsmtCond_TA	0.087267
SaleCondition_Normal	0.081329
Neighborhood_NWAmes	-0.000000
MasVnrType_Others	-0.000000
Neighborhood_OldTown	-0.020575
MSSubClass_Others	-0.061735
MSSubClass_50	-0.070666
Foundation_Others	-0.117075
Neighborhood_NAmes	-0.173756
Neighborhood_Sawyer	-0.211820
Neighborhood_Edwards	-0.213079
SaleType_WD	-0.254158
SaleType_Others	-0.267855

Top 5 predictor variables are (overall absolute value)

- OverallQual_Others
- OverallQual_8
- GarageCond_TA
- SaleType_Others
- SaleType_WD

Top 5 which positively affect are

- OverallQual_Others
- OverallQual_8
- GarageCond_TA
- MSZoning_RL
- Neighborhood_Somerst

Top 5 which negatively affect are

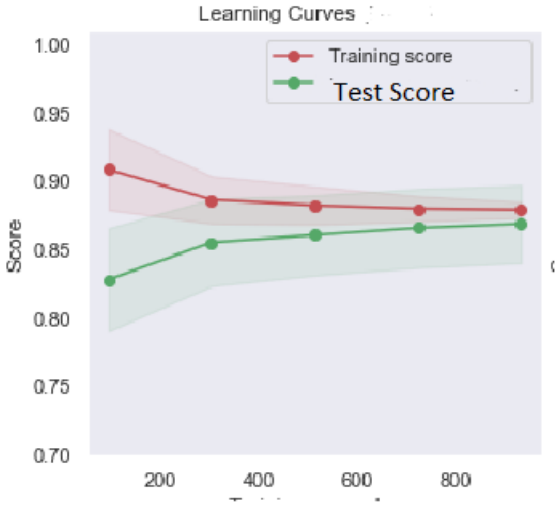
- Neighborhood_NAmes
- Neighborhood_Sawyer
- Neighborhood_Edwards
- SaleType_WD
- SaleType_Others

Question 4. How can you make sure that a model is robust and generalizable? What are the implications of the same for the accuracy of the model and why?

Answer.

A model can be called as robust and generalisable when its train and test accuracy are similar i.e. it doesn't vary much. Its accuracy should not reduce when exposed to unseen dataset.
In order to make sure that the model is robust we need to take care of few things while model building.

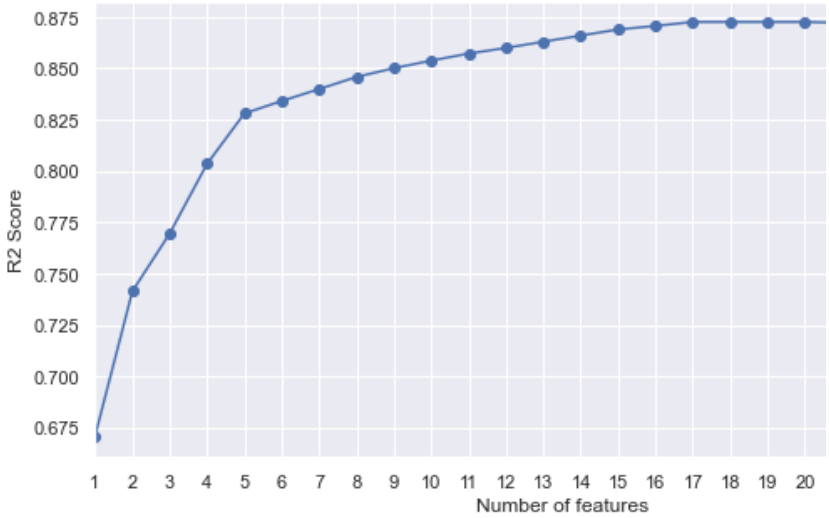
1. If Possible, add more data while training.



It allows the "data to tell for itself," instead of relying on assumptions and weak correlations. Presence of more data results in better and accurate models. If we observe the learning curve with increase in data points the test and train r2 score tend to come closer to each other.

2. **Treat missing Values and Outliers**
   The unwanted outliers and missing values can often reduce the accuracy of the model. This is because model doesn't analyse the behaviours and relationship with other variables correctly. Hence before model building, we need to treat them.

3. **Feature selection.**



   We can observe that with increase in features the r2 score increases. So, we would need to select those features which can explain the maximum variance of a dataset. For e.g. We can use RFE or Lasso for feature selection.

4. While selecting a model we would need to keep Occam's razor in mind as well(A model should be as simple as necessary, but no simpler)

   If a model is not robust it cannot be used for predictive analysis as the results cannot be trusted .