Subjective Questions

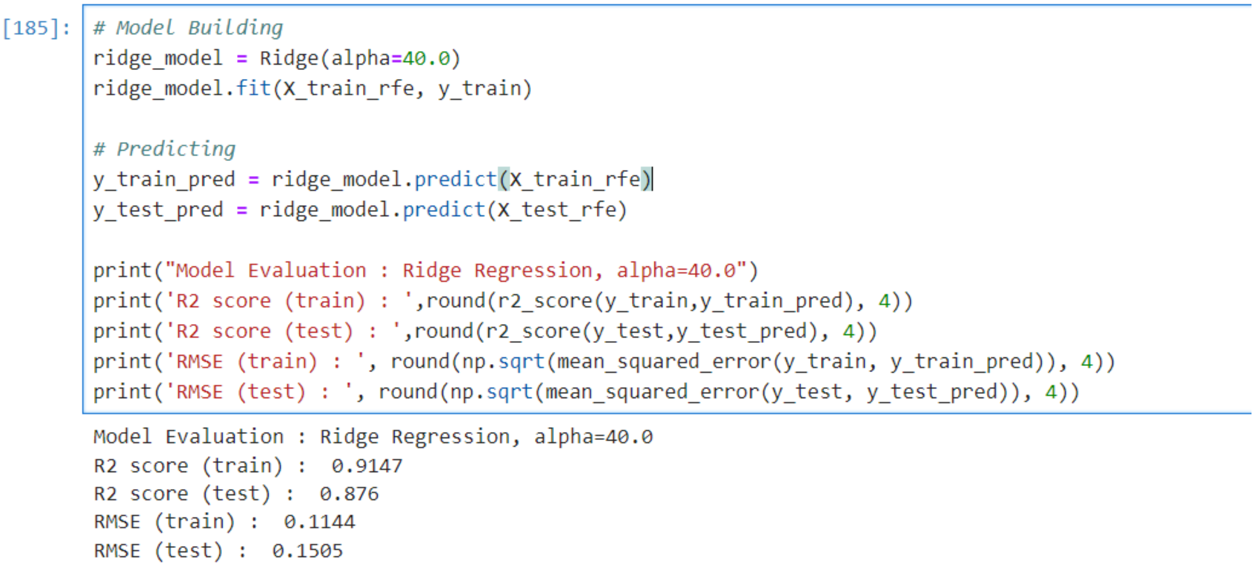
Question 1

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?

Optimal Value of alpha for ridge and lasso regression:

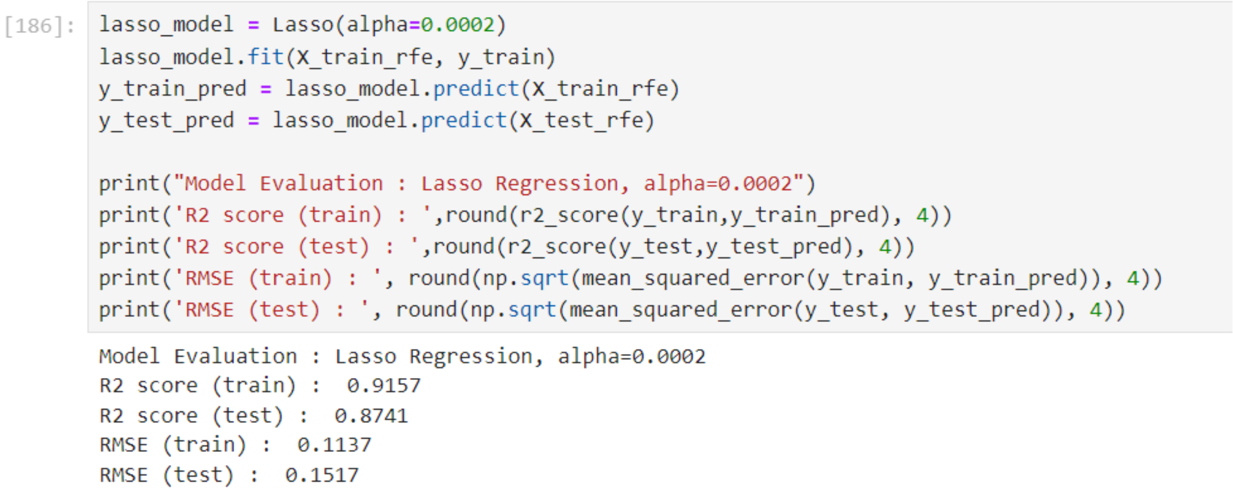
Alpha_ridge = 20.0; Alpha_lasso = 0.0001

⇨ Doubling the Alpha values:



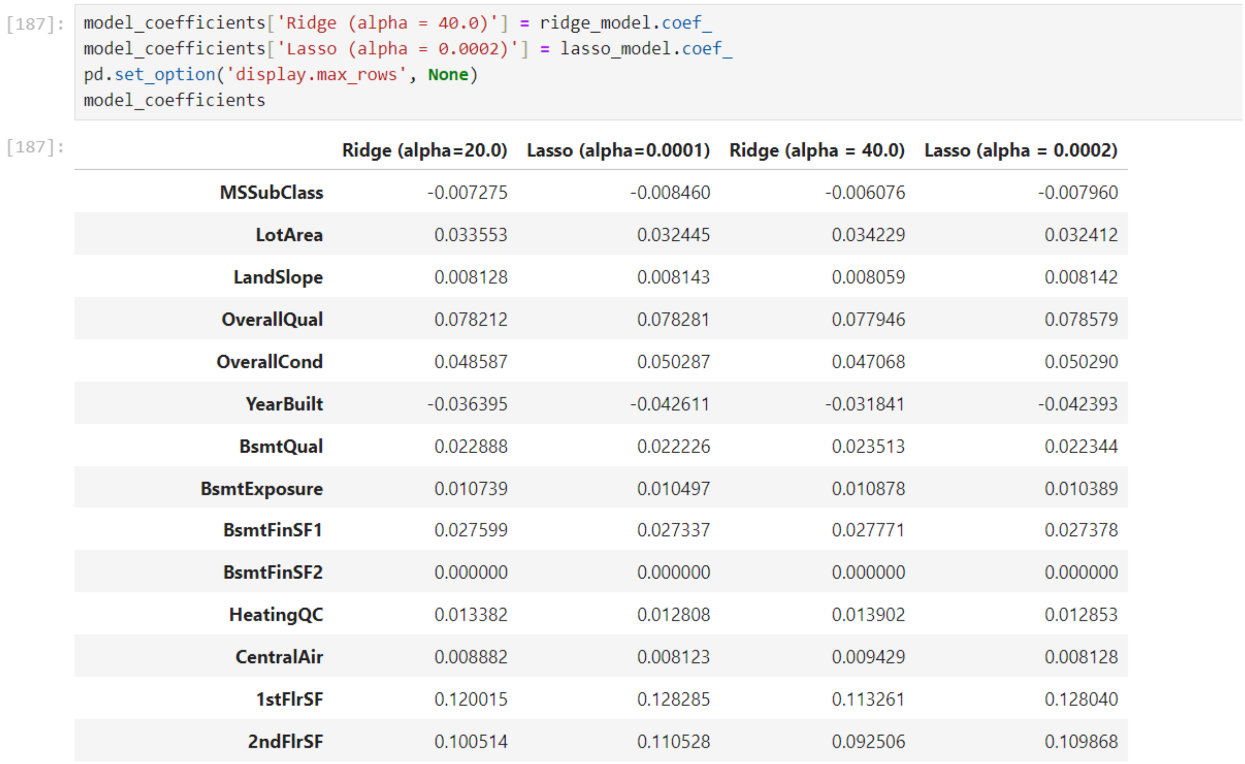
```
[185]: # Model Building
       ridge_model = Ridge(alpha=40.0)
       ridge_model.fit(X_train_rfe, y_train)

       # Predicting
       y_train_pred = ridge_model.predict(X_train_rfe)
       y_test_pred = ridge_model.predict(X_test_rfe)

       print("Model Evaluation : Ridge Regression, alpha=40.0")
       print('R2 score (train) : ',round(r2_score(y_train,y_train_pred), 4))
       print('R2 score (test) : ',round(r2_score(y_test,y_test_pred), 4))
       print('RMSE (train) : ', round(np.sqrt(mean_squared_error(y_train, y_train_pred)), 4))
       print('RMSE (test) : ', round(np.sqrt(mean_squared_error(y_test, y_test_pred)), 4))
```

```
Model Evaluation : Ridge Regression, alpha=40.0
R2 score (train) :  0.9147
R2 score (test) :  0.876
RMSE (train) :  0.1144
RMSE (test) :  0.1505
```



```
[186]: lasso_model = Lasso(alpha=0.0002)
       lasso_model.fit(X_train_rfe, y_train)
       y_train_pred = lasso_model.predict(X_train_rfe)
       y_test_pred = lasso_model.predict(X_test_rfe)

       print("Model Evaluation : Lasso Regression, alpha=0.0002")
       print('R2 score (train) : ',round(r2_score(y_train,y_train_pred), 4))
       print('R2 score (test) : ',round(r2_score(y_test,y_test_pred), 4))
       print('RMSE (train) : ', round(np.sqrt(mean_squared_error(y_train, y_train_pred)), 4))
       print('RMSE (test) : ', round(np.sqrt(mean_squared_error(y_test, y_test_pred)), 4))
```

```
Model Evaluation : Lasso Regression, alpha=0.0002
R2 score (train) :  0.9157
R2 score (test) :  0.8741
RMSE (train) :  0.1137
RMSE (test) :  0.1517
```

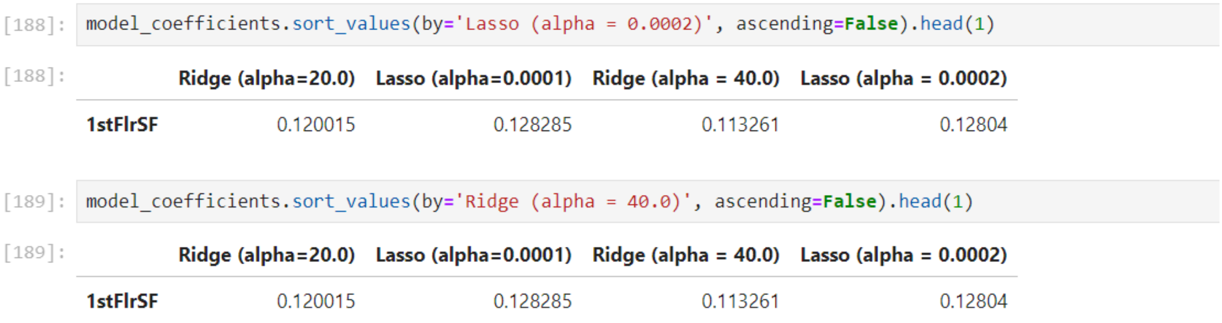


```
[187]: model_coefficients['Ridge (alpha = 40.0)'] = ridge_model.coef_
       model_coefficients['Lasso (alpha = 0.0002)'] = lasso_model.coef_
       pd.set_option('display.max_rows', None)
       model_coefficients
```

[187]:

	Ridge (alpha=20.0)	Lasso (alpha=0.0001)	Ridge (alpha = 40.0)	Lasso (alpha = 0.0002)
MSSubClass	-0.007275	-0.008460	-0.006076	-0.007960
LotArea	0.033553	0.032445	0.034229	0.032412
LandSlope	0.008128	0.008143	0.008059	0.008142
OverallQual	0.078212	0.078281	0.077946	0.078579
OverallCond	0.048587	0.050287	0.047068	0.050290
YearBuilt	-0.036395	-0.042611	-0.031841	-0.042393
BsmtQual	0.022888	0.022226	0.023513	0.022344
BsmtExposure	0.010739	0.010497	0.010878	0.010389
BsmtFinSF1	0.027599	0.027337	0.027771	0.027378
BsmtFinSF2	0.000000	0.000000	0.000000	0.000000
HeatingQC	0.013382	0.012808	0.013902	0.012853
CentralAir	0.008882	0.008123	0.009429	0.008128
1stFlrSF	0.120015	0.128285	0.113261	0.128040
2ndFlrSF	0.100514	0.110528	0.092506	0.109868

⇨ Most important predictor variables after the change is implemented:



```
[188]: model_coefficients.sort_values(by='Lasso (alpha = 0.0002)', ascending=False).head(1)
```

[188]:

	Ridge (alpha=20.0)	Lasso (alpha=0.0001)	Ridge (alpha = 40.0)	Lasso (alpha = 0.0002)
1stFlrSF	0.120015	0.128285	0.113261	0.12804

```
[189]: model_coefficients.sort_values(by='Ridge (alpha = 40.0)', ascending=False).head(1)
```

[189]:

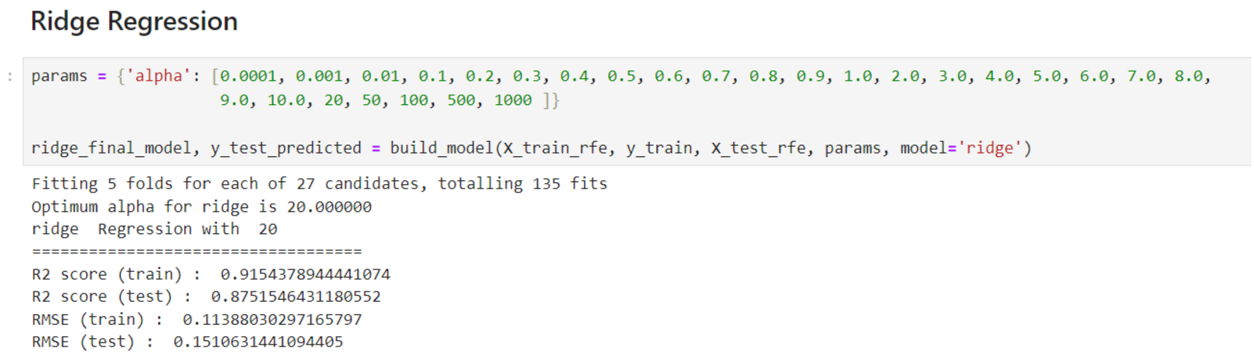
	Ridge (alpha=20.0)	Lasso (alpha=0.0001)	Ridge (alpha = 40.0)	Lasso (alpha = 0.0002)
1stFlrSF	0.120015	0.128285	0.113261	0.12804

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?

Alpha value for Ridge= 20.0, because Ridge Regression model was able to achieve R2 score of 0.87 on test data i.e**. 87%** of the variance in test data can be explained by the model.

Root Mean Square Error = 0.1510 on test data, that means the prediction made by the model can off by 0.1510 unit.



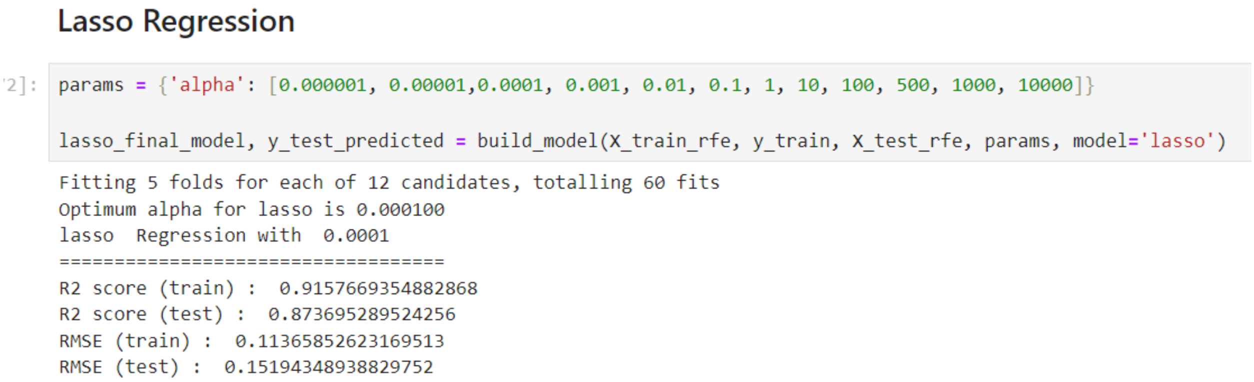
Ridge Regression

```
: params = {'alpha': [0.0001, 0.001, 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0,
                      9.0, 10.0, 20, 50, 100, 500, 1000 ]}

  ridge_final_model, y_test_predicted = build_model(X_train_rfe, y_train, X_test_rfe, params, model='ridge')
```

```
Fitting 5 folds for each of 27 candidates, totalling 135 fits
Optimum alpha for ridge is 20.000000
ridge  Regression with  20
==================================
R2 score (train) :  0.9154378944441074
R2 score (test) :  0.8751546431180552
RMSE (train) :  0.11388030297165797
RMSE (test) :  0.1510631441094405
```

- ⇨ Alpha value for Lasso= 0.0001, because Lasso Regression model was able to achieve R2 score of 0.87 on test data i.e**. 87%** of the variance in test data can be explained by the model.

Root Mean Square Error = 0.1519 on test data, that means the prediction made by the model can off by 0.1519 unit.



Lasso Regression

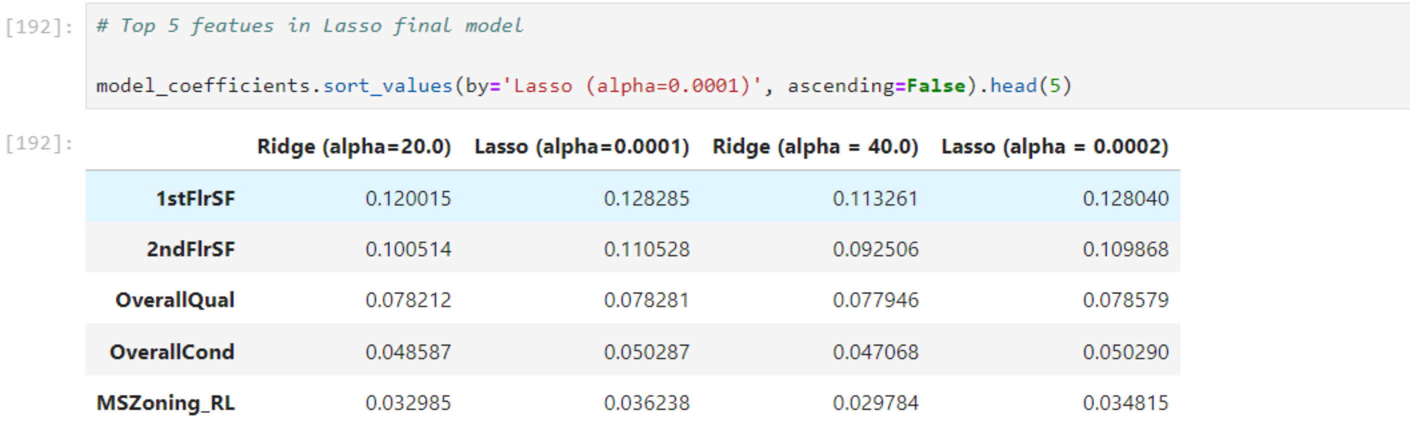
```
2]: params = {'alpha': [0.000001, 0.00001,0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 500, 1000, 10000]}

    lasso_final_model, y_test_predicted = build_model(X_train_rfe, y_train, X_test_rfe, params, model='lasso')
```

```
Fitting 5 folds for each of 12 candidates, totalling 60 fits
Optimum alpha for lasso is 0.000100
lasso  Regression with  0.0001
==================================
R2 score (train) :  0.9157669354882868
R2 score (test) :  0.873695289524256
RMSE (train) :  0.11365852623169513
RMSE (test) :  0.15194348938829752
```

Question 3

After building the model, you realised 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?

Top 5 features are-> ['1stFlrSF', '2ndFlrSF', 'OverallQual', 'OverallCond', 'MSZoning_RL']

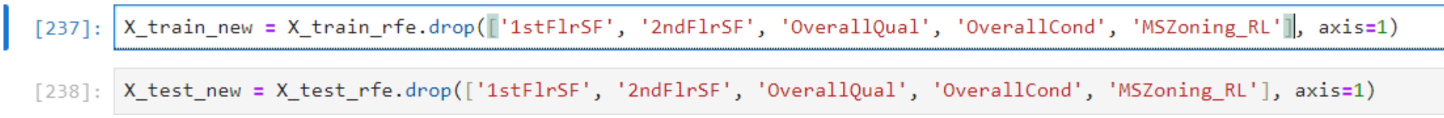


```
[192]: # Top 5 featues in Lasso final model

       model_coefficients.sort_values(by='Lasso (alpha=0.0001)', ascending=False).head(5)
```

[192]:

	Ridge (alpha=20.0)	Lasso (alpha=0.0001)	Ridge (alpha = 40.0)	Lasso (alpha = 0.0002)
1stFlrSF	0.120015	0.128285	0.113261	0.128040
2ndFlrSF	0.100514	0.110528	0.092506	0.109868
OverallQual	0.078212	0.078281	0.077946	0.078579
OverallCond	0.048587	0.050287	0.047068	0.050290
MSZoning_RL	0.032985	0.036238	0.029784	0.034815

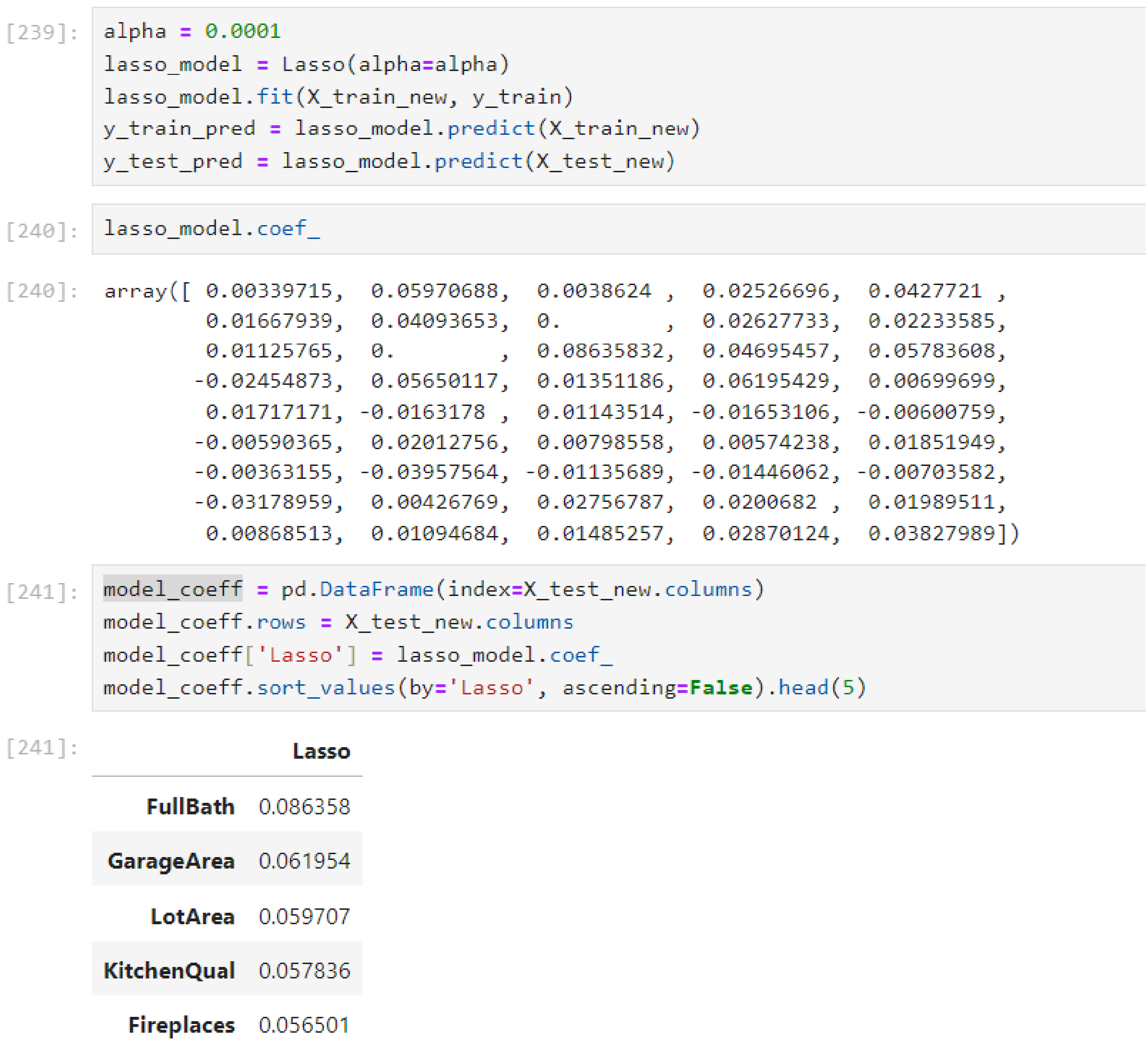
Dropping these 5 features from the data:



```
[237]: X_train_new = X_train_rfe.drop(['1stFlrSF', '2ndFlrSF', 'OverallQual', 'OverallCond', 'MSZoning_RL'], axis=1)

[238]: X_test_new = X_test_rfe.drop(['1stFlrSF', '2ndFlrSF', 'OverallQual', 'OverallCond', 'MSZoning_RL'], axis=1)
```

Now creating another model excluding these top 5 features or predictor variables.

With alpha value= 0.0001 using Lasso Regression, making a model to predict House Prices.



```
[239]: alpha = 0.0001
       lasso_model = Lasso(alpha=alpha)
       lasso_model.fit(X_train_new, y_train)
       y_train_pred = lasso_model.predict(X_train_new)
       y_test_pred = lasso_model.predict(X_test_new)
```

```
[240]: lasso_model.coef_
```

```
[240]: array([ 0.00339715,  0.05970688,  0.0038624 ,  0.02526696,  0.0427721 ,
               0.01667939,  0.04093653,  0.        ,  0.02627733,  0.02233585,
               0.01125765,  0.        ,  0.08635832,  0.04695457,  0.05783608,
              -0.02454873,  0.05650117,  0.01351186,  0.06195429,  0.00699699,
               0.01717171, -0.0163178 ,  0.01143514, -0.01653106, -0.00600759,
              -0.00590365,  0.02012756,  0.00798558,  0.00574238,  0.01851949,
              -0.00363155, -0.03957564, -0.01135689, -0.01446062, -0.00703582,
              -0.03178959,  0.00426769,  0.02756787,  0.0200682 ,  0.01989511,
               0.00868513,  0.01094684,  0.01485257,  0.02870124,  0.03827989])
```

```
[241]: model_coeff = pd.DataFrame(index=X_test_new.columns)
       model_coeff.rows = X_test_new.columns
       model_coeff['Lasso'] = lasso_model.coef_
       model_coeff.sort_values(by='Lasso', ascending=False).head(5)
```

[241]:

	Lasso
FullBath	0.086358
GarageArea	0.061954
LotArea	0.059707
KitchenQual	0.057836
Fireplaces	0.056501

Top 5 variables now after excluding earlier top 5 predictor variables are :

1. **FullBath**
2. **GarageArea**
3. **LotArea**
4. **KitchenQual**
5. **Fireplaces**

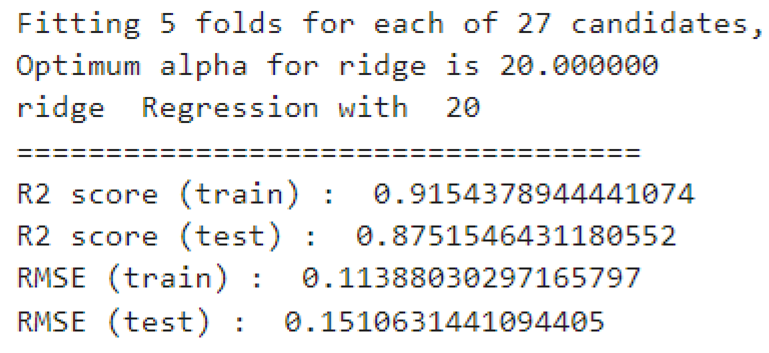
Question 4

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

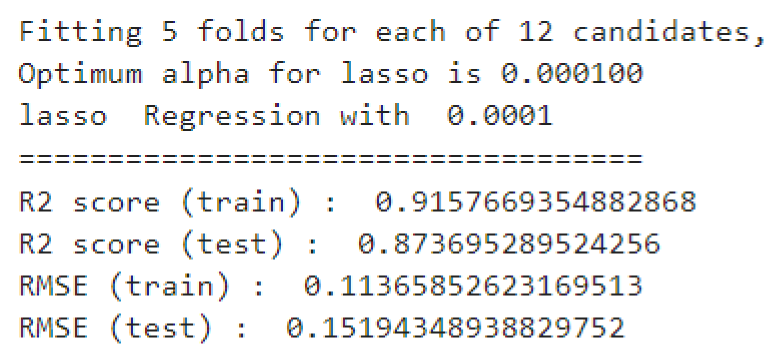
- Model is robust and generalizable when:
    - No outliers
    - No Missing values
    - No Errors
    - Simple(less number of predictors)
    - Moderate Bias and Variance

- Accuracy of Model can be explained by R-squared metrics, when R2 scores are greater than 75%,
- When error terms RSS and RMSE are very low.

With above mentioned conditions, we can say our model is accurate and good.



```
Fitting 5 folds for each of 27 candidates,
Optimum alpha for ridge is 20.000000
ridge  Regression with  20
====================================
R2 score (train) :  0.9154378944441074
R2 score (test) :  0.8751546431180552
RMSE (train) :  0.11388030297165797
RMSE (test) :  0.1510631441094405
```



```
Fitting 5 folds for each of 12 candidates,
Optimum alpha for lasso is 0.000100
lasso  Regression with  0.0001
====================================
R2 score (train) :  0.9157669354882868
R2 score (test) :  0.873695289524256
RMSE (train) :  0.11365852623169513
RMSE (test) :  0.15194348938829752
```