Advance Regression Assignment
Submitted By: Neeraj Kumar (Batch No 39)

<u>Question 1:</u>

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?

<u>Answer 1:</u>

Optimal value of alpha for Ridge regression: 3
Optimal value of alpha for Lasso regression: 0.1

<u>Changes in the model by doubling the alpha value</u>:
1.1 Impact on Matrix: From the matrices analysis, it becomes evident that the impact of doubling the value of alpha has been very minor on the matrices for Ridge as well as Lasso regression. Approximately 1% R2 has gone down and approximate 1% RMSE has gone up for both type of regressions. Please see the snapshot below-

	Matric	Linear Regression	Ridge Regression	Ridge 2X_Best Lambda	Lasso Regression	Lasso 2X Best Lambda
0	R2 - Train	0.897331	0.889344	0.881684	0.885438	0.873265
1	R2 - Test	0.857235	0.866201	0.861982	0.871267	0.861517
2	RSS - Train	735133.030416	792322.318116	847168.469496	820285.853975	907447.080367
3	RSS - Test	255886.950033	239817.005984	247378.750369	230737.241666	248212.067351
4	MSE - Train	639.246113	688.975929	736.668234	713.292047	789.084418
5	MSE - Test	888.496354	832.697937	858.953994	801.170978	861.847456
6	RMSE - Train	25.283317	26.248351	27.141633	26.707528	28.090646
7	RMSE - Test	29.807656	28.856506	29.307917	28.304964	29.357239

1.2 Impact on Co-efficients for Ridge Regression:

```
model_coef1.sort_values(by=['best_lambda'], ascending = False).head(10)
```

	features	best_lambda	2X_best_lambda	lasso_best_lambda	2X_lasso_best_lambda	LReg
5	TotalBsmtSF	102.776396	83.925775	140.651068	135.379798	154.352195
2	OverallQual	71.626655	67.301459	85.834807	100.442730	67.081852
6	2ndFlrSF	71.394817	53.683919	89.189222	68.375143	126.820156
13	TotRmsAbvGrd	54.847709	50.846529	50.708281	41.522277	60.877966
91	RoofMatl_WdShngl	53.177887	41.734703	57.825502	39.716538	72.942712
68	Neighborhood_NoRidge	44.663785	42.441120	48.436837	45.037787	46.114251
75	Neighborhood_StoneBr	44.275746	38.923554	45.950143	38.104676	53.445941
1	LotArea	38.202424	32.037919	39.919710	36.744768	46.000542
14	GarageCars	35.271218	37.371459	34.441291	38.566306	31.949690
3	OverallCond	33.826413	28.357008	38.385573	33.549460	43.661546

```
model_coef1.sort_values(by=['2X_best_lambda'], ascending = False).head(10)
```

	features	best_lambda	2X_best_lambda	lasso_best_lambda	2X_lasso_best_lambda	LReg
5	TotalBsmtSF	102.776396	83.925775	140.651068	135.379798	154.352195
2	OverallQual	71.626655	67.301459	85.834807	100.442730	67.081852
6	2ndFlrSF	71.394817	53.683919	89.189222	68.375143	126.820156
13	TotRmsAbvGrd	54.847709	50.846529	50.708281	41.522277	60.877966
68	Neighborhood_NoRidge	44.663785	42.441120	48.436837	45.037787	46.114251
91	RoofMatl_WdShngl	53.177887	41.734703	57.825502	39.716538	72.942712
75	Neighborhood_StoneBr	44.275746	38.923554	45.950143	38.104676	53.445941
14	GarageCars	35.271218	37.371459	34.441291	38.566306	31.949690
9	FullBath	33.285570	33.495591	20.100975	14.060592	28.034232
1	LotArea	38.202424	32.037919	39.919710	36.744768	46.000542

It is clear from the above snapshots that value of coefficients has gone down by 10 to 20% by doubling of alpha from 3 to 6. Also, it did not change the top 7 features or their sequence. There is minor shuffling of features from 8 to 10 position but I believe it is because their co-efficient values are very close to each other around 35.

1.3 Impact on Co-efficients for Lasso Regression:

```
model_coef1.sort_values(by=['lasso_best_lambda'], ascending = False).head(10)
```

	features	best_lambda	2X_best_lambda	lasso_best_lambda	2X_lasso_best_lambda	LReg
5	TotalBsmtSF	102.776396	83.925775	140.651068	135.379798	154.352195
6	2ndFlrSF	71.394817	53.683919	89.189222	68.375143	126.820156
2	OverallQual	71.626655	67.301459	85.834807	100.442730	67.081852
91	RoofMatl_WdShngl	53.177887	41.734703	57.825502	39.716538	72.942712
13	TotRmsAbvGrd	54.847709	50.846529	50.708281	41.522277	60.877966
68	Neighborhood_NoRidge	44.663785	42.441120	48.436837	45.037787	46.114251
75	Neighborhood_StoneBr	44.275746	38.923554	45.950143	38.104676	53.445941
1	LotArea	38.202424	32.037919	39.919710	36.744768	46.000542
3	OverallCond	33.826413	28.357008	38.385573	33.549460	43.661546
14	GarageCars	35.271218	37.371459	34.441291	38.566306	31.949690

```
model_coef1.sort_values(by=['2X_lasso_best_lambda'], ascending = False).head(10)
```

	features	best_lambda	2X_best_lambda	lasso_best_lambda	2X_lasso_best_lambda	LReg
5	TotalBsmtSF	102.776396	83.925775	140.651068	135.379798	154.352195
2	OverallQual	71.626655	67.301459	85.834807	100.442730	67.081852
6	2ndFlrSF	71.394817	53.683919	89.189222	68.375143	126.820156
68	Neighborhood_NoRidge	44.663785	42.441120	48.436837	45.037787	46.114251
13	TotRmsAbvGrd	54.847709	50.846529	50.708281	41.522277	60.877966
91	RoofMatl_WdShngl	53.177887	41.734703	57.825502	39.716538	72.942712
14	GarageCars	35.271218	37.371459	34.441291	38.566306	31.949690
75	Neighborhood_StoneBr	44.275746	38.923554	45.950143	38.104676	53.445941
1	LotArea	38.202424	32.037919	39.919710	36.744768	46.000542
3	OverallCond	33.826413	28.357008	38.385573	33.549460	43.661546

In case of Lasso regression, value of coefficients of most of the variable has gone down but at the same time; value of coef of one of the 3rd feature by significance has gone up I.e. OverallQual. Range of value reduction of the coef is wide I.e. 5% to 20%. Significance order of the features has also changed.

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 2:
I will choose to go for the lasso regression because it has number of features which are predicted to have coeff of zero. This way model will be less complex than the Ridge regression model. Statistically also if you look at the matrices, there is almost no benefit of going for the more complex model.

	Matric	Linear Regression	Ridge Regression	Ridge 2X_Best Lambda	Lasso Regression	Lasso 2X Best Lambda
0	R2 - Train	0.897331	0.889344	0.881684	0.885438	0.873265
1	R2 - Test	0.857235	0.866201	0.861982	0.871267	0.861517
2	RSS - Train	735133.030416	792322.318116	847168.469496	820285.853975	907447.080367
3	RSS - Test	255886.950033	239817.005984	247378.750369	230737.241666	248212.067351
4	MSE - Train	639.246113	688.975929	736.668234	713.292047	789.084418
5	MSE - Test	888.496354	832.697937	858.953994	801.170978	861.847456
6	RMSE - Train	25.283317	26.248351	27.141633	26.707528	28.090646
7	RMSE - Test	29.807656	28.856506	29.307917	28.304964	29.357239

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 3:

```
model_coef1.sort_values(by=['lasso_best_lambda'], ascending = False).head(10)
```

	features	best_lambda	2X_best_lambda	lasso_best_lambda	2X_lasso_best_lambda	LReg
5	TotalBsmtSF	102.776396	83.925775	140.651068	135.379798	154.352195
6	2ndFlrSF	71.394817	53.683919	89.189222	68.375143	126.820156
2	OverallQual	71.626655	67.301459	85.834807	100.442730	67.081852
91	RoofMatl_WdShngl	53.177887	41.734703	57.825502	39.716538	72.942712
13	TotRmsAbvGrd	54.847709	50.846529	50.708281	41.522277	60.877966
68	Neighborhood_NoRidge	44.663785	42.441120	48.436837	45.037787	46.114251
75	Neighborhood_StoneBr	44.275746	38.923554	45.950143	38.104676	53.445941
1	LotArea	38.202424	32.037919	39.919710	36.744768	46.000542
3	OverallCond	33.826413	28.357008	38.385573	33.549460	43.661546
14	GarageCars	35.271218	37.371459	34.441291	38.566306	31.949690

In the snapshot above, features are listed in the priority order in which they impact the sale price of the house as per Lasso model. If top 5 features are not present in the incoming data, then I believe next five as listed below will play the most significant role in the new model and will be top 5 features. Needless to say their coefficient values will change in the new model.

Neighborhood_NoRidge
Neighborhood_StoneBr
LotArea
OverallCond
GarageCars

<u>Question 4:</u>

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?

<u>Answer 4</u>:
Generalization of a simple model is much more than the complex ones. Also, Simpler models are more robust. Robustness of a model is characterized by the fact that they are not as sensitive to the specifics of the training data as the complex models are.

Impact on the accuracy:

Accuracy, Robustness and generalization of a model are the result of Bias-Variance tradeoff. Bias indicates how well model is able to generalize the training data provided. High bias will lead to a very generalized model, as an extreme it might be just returning a constant. Variance indicates that how well model has learned the training data. High Variance leads to very good results on the training data because model has learned the data itself rather than understanding the pattern of the data but errors on the test data. This is called overfitting. One golden rule to identify the overfitting is when model is performing well on the training data but performing poorly on the test data / unseen data. In this case, model should be corrected.