

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?

Ridge Regression –

Objective = $RSS + \alpha * (\text{sum of square of coefficients})$

Here, α (alpha) is the parameter which balances the amount of emphasis given to minimizing RSS vs minimizing sum of square of coefficients.

The optimal value of alpha for the Ridge Regression is = 5.0

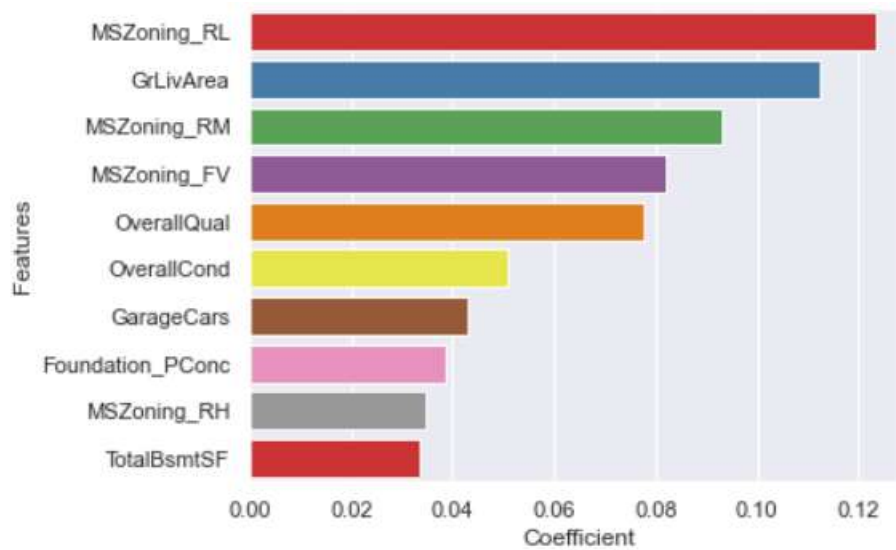
The R2 score for Training data = 91.33%

The R2 score for Testing data = 90.61%

The Mean squared error is = 0.0137

The top 10 predictor features for the model –

	Features	rfe_support	rfe_ranking	Coefficient
10	MSZoning_RL	True	1	0.1233
4	GrLivArea	True	1	0.1125
11	MSZoning_RM	True	1	0.0929
8	MSZoning_FV	True	1	0.0819
0	OverallQual	True	1	0.0776
1	OverallCond	True	1	0.0508
6	GarageCars	True	1	0.0429
14	Foundation_PConc	True	1	0.0386
9	MSZoning_RH	True	1	0.0345
3	TotalBsmtSF	True	1	0.0335



Lasso Regression

Objective = $RSS + \alpha * (\text{sum of absolute value of coefficients})$

Here, α (alpha) works similar to that of ridge and provides a trade-off between balancing RSS and magnitude of coefficients.

The optimal value of alpha for the Ridge Regression is = 0.001

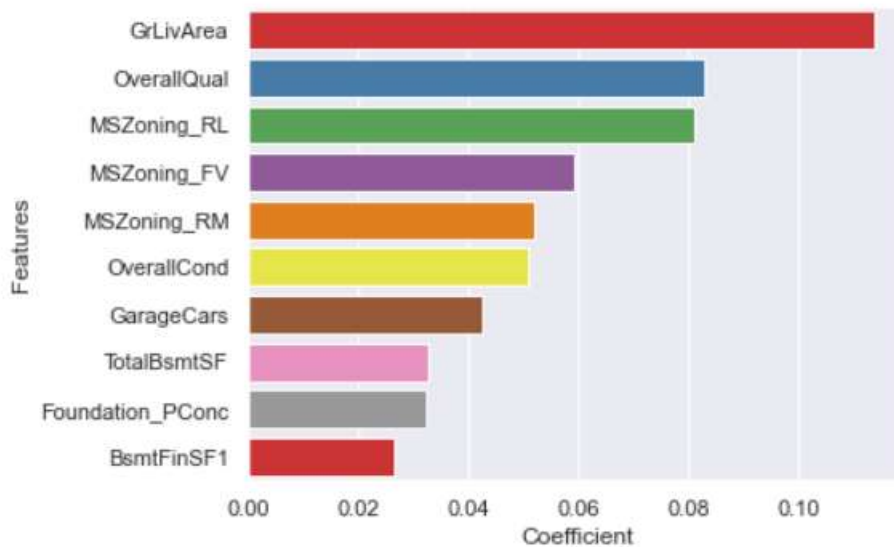
The R2 score for Training data = 91.05%

The R2 score for Testing data = 90.83%

The Mean squared error is = 0.0134

The top 10 predictor features for the model –

	Features	rfe_support	rfe_ranking	Coefficient
5	GrLivArea	True	1	0.113762
1	OverallQual	True	1	0.082824
12	MSZoning_RL	True	1	0.081251
10	MSZoning_FV	True	1	0.059342
13	MSZoning_RM	True	1	0.051924
2	OverallCond	True	1	0.050977
8	GarageCars	True	1	0.042684
4	TotalBsmtSF	True	1	0.032876
14	Foundation_PConc	True	1	0.032325
3	BsmtFinSF1	True	1	0.026386



Model performance after doubling the value of optimal alpha

Ridge Regression –

Optimal alpha = 5.0

Current alpha = 10.0

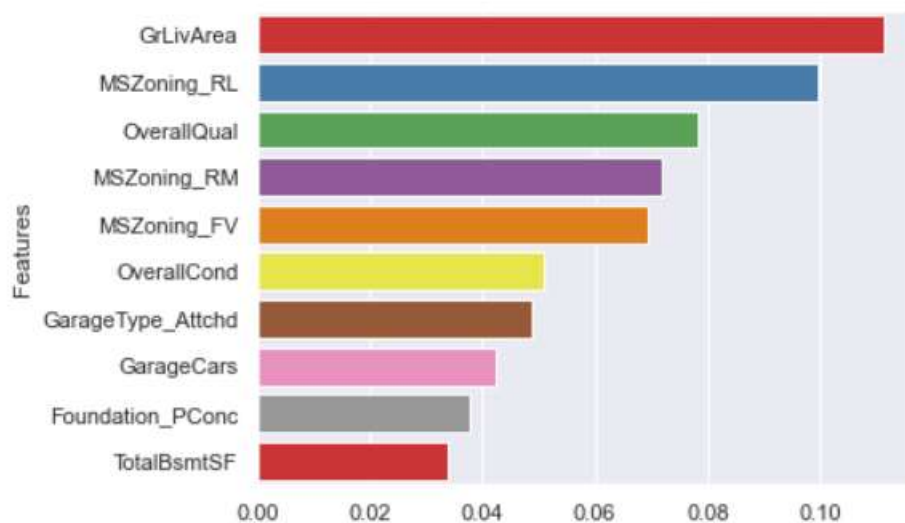
Training R2 Score = 91.26%

Testing R2 Score = 90.63%

The mean squared error = 0.0137

The top 10 predictor features for the model –

	Features	rfe_support	rfe_ranking	Coefficient
9	GrLivArea	True	1	0.1112
22	MSZoning_RL	True	1	0.0995
2	OverallQual	True	1	0.0784
23	MSZoning_RM	True	1	0.0718
20	MSZoning_FV	True	1	0.0695
3	OverallCond	True	1	0.0508
45	GarageType_Attchd	True	1	0.0488
15	GarageCars	True	1	0.0422
43	Foundation_PConc	True	1	0.0376
7	TotalBsmtSF	True	1	0.0337



Lasso Regression:

Optimal value of alpha = 0.001

Current value of alpha = 0.002

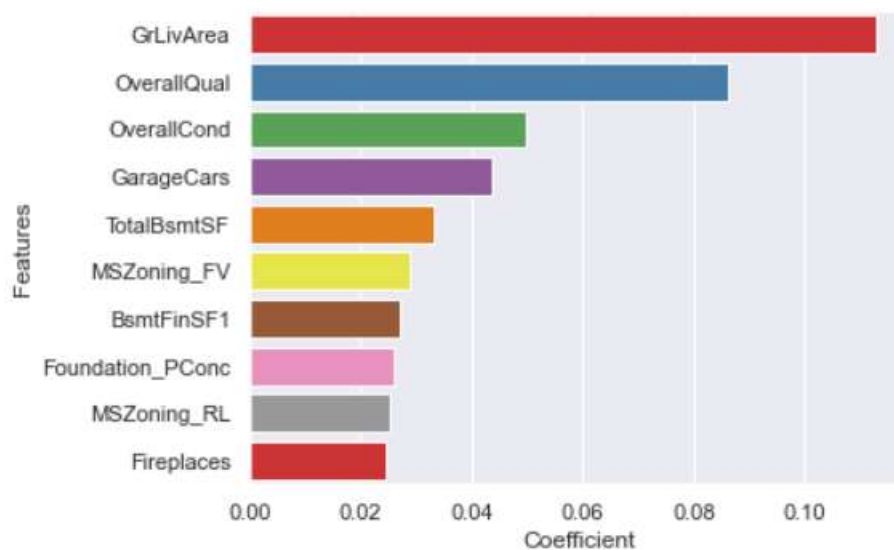
The R2 score for Training data = 90.71%

The R2 score for Testing data = 90.47%

The Mean squared error is = 0.014

The top 10 predictor features for the model –

	Features	rfe_support	rfe_ranking	Coefficient
9	GrLivArea	True	1	0.112809
2	OverallQual	True	1	0.086235
3	OverallCond	True	1	0.049894
15	GarageCars	True	1	0.043713
7	TotalBsmtSF	True	1	0.033237
20	MSZoning_FV	True	1	0.028857
6	BsmtFinSF1	True	1	0.027097
35	Foundation_PConc	True	1	0.025977
22	MSZoning_RL	True	1	0.025182
13	Fireplaces	True	1	0.024413



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?

The Lasso Regression would be the best option it would help in feature elimination and the model will be robust.

Lasso tends to do well if there are a small number of significant parameters and the others are close to zero (ergo: when only a few predictors actually influence the response). **Ridge** works well if there are many large parameters of about the same value (ergo: when most predictors impact the response).

The **Lasso's** penalty term is based on the sum of absolute coefficients, and the specification of a penalty coefficient is similar to that of **Ridge regression**; however, the **Lasso** is **more computationally intensive**.

Question 3

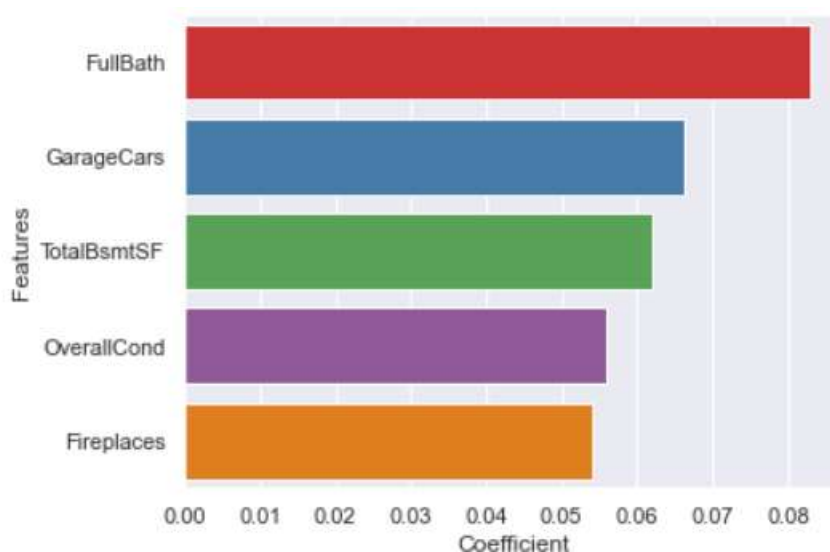
After building the model, you realized that the five most important predictor variables in the lasso model are not available in the incoming data. You will now have to create another model excluding the five most important predictor variables. Which are the five most important predictor variables now?

The top 5 predictor variables in the model with optimal alpha (0.001) were –

- GrLivArea
- OverallQual
- MSZoning_RL
- MSZoning_FV
- MSZoning_RM

If we drop them and build the model with same alpha – The top 5 predictor variables would now be –

	Features	rfe_support	rfe_ranking	Coefficient
9	FullBath	True	1	0.082904
13	GarageCars	True	1	0.066248
6	TotalBsmtSF	True	1	0.062023
2	OverallCond	True	1	0.056003
11	Fireplaces	True	1	0.053944



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?

Per, Occam's Razor—given two models that show similar 'performance' in the finite training or test data, we should pick the one that makes fewer on the test data due to following reasons:-

- Simpler models are usually more 'generic' and are more widely applicable
- Simpler models require fewer training samples for effective training than the more complex ones and hence are easier to train.
- Simpler models are more robust.

- Complex models tend to change wildly with changes in the training data set
- Simple models have low variance, high bias and complex models have low bias, high variance
- Simpler models make more errors in the training set. Complex models lead to over fitting —they work very well for the training samples, fail miserably when applied to other test samples

Therefore to make the model more robust and generalizable, make the model simple but not simpler which will not be of any use. Regularization can be used to make the model simpler. Regularization helps to strike the delicate balance between keeping the model simple and not making it too naive to be of any use. For regression, regularization involves adding a regularization term to the cost that adds up the absolute values or the squares of the parameters of the model.

Also, Making a model simple leads to Bias-Variance Trade-off:

- A complex model will need to change for every little change in the dataset and hence is very unstable and extremely sensitive to any changes in the training data.
- A simpler model that abstracts out some pattern followed by the data points given is unlikely to change wildly even if more points are added or removed

Bias quantifies how accurate is the model likely to be on test data. A complex model can do an accurate job prediction provided there is enough training data. Models that are too naïve, for e.g., one that gives same answer to all test inputs and makes no discrimination whatsoever has a very large bias as its expected error across all test inputs are very high.

Variance refers to the degree of changes in the model itself with respect to changes in the training data.

Thus accuracy of the model can be maintained by keeping the balance between Bias and Variance as it minimizes the total error as shown in the below graph

