Assignment Part – II

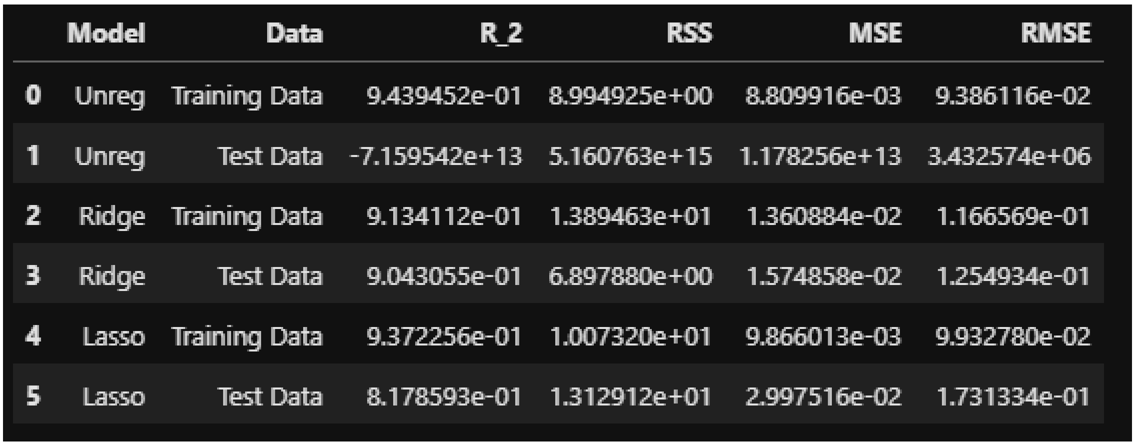
Question 1:

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:

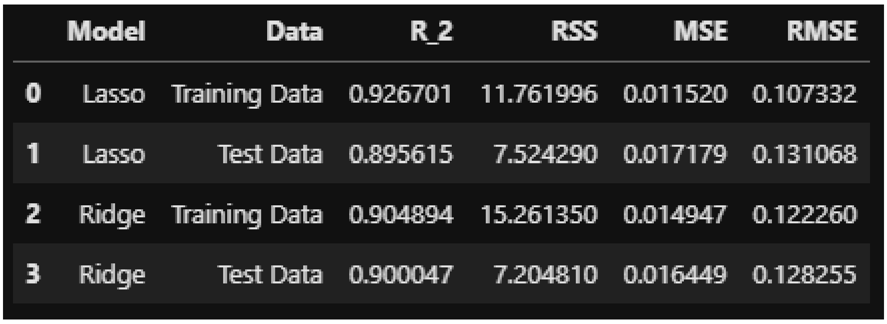
The optimal value of alpha for ridge & lasso regression is ‘5’ & ‘0.0001’. Now, If we double the optimal alpha, then, model‘ll lead towards under fitting which means higher MSE value.

This is the model performance at their best alpha.



	Model	Data	R_2	RSS	MSE	RMSE
0	Unreg	Training Data	9.439452e-01	8.994925e+00	8.809916e-03	9.386116e-02
1	Unreg	Test Data	-7.159542e+13	5.160763e+15	1.178256e+13	3.432574e+06
2	Ridge	Training Data	9.134112e-01	1.389463e+01	1.360884e-02	1.166569e-01
3	Ridge	Test Data	9.043055e-01	6.897880e+00	1.574858e-02	1.254934e-01
4	Lasso	Training Data	9.372256e-01	1.007320e+01	9.866013e-03	9.932780e-02
5	Lasso	Test Data	8.178593e-01	1.312912e+01	2.997516e-02	1.731334e-01

This is the model performance when we double the alpha.

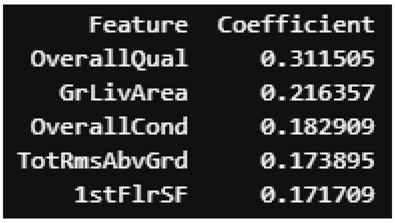


	Model	Data	R_2	RSS	MSE	RMSE
0	Lasso	Training Data	0.926701	11.761996	0.011520	0.107332
1	Lasso	Test Data	0.895615	7.524290	0.017179	0.131068
2	Ridge	Training Data	0.904894	15.261350	0.014947	0.122260
3	Ridge	Test Data	0.900047	7.204810	0.016449	0.128255

We can clearly notice the increase in MSE value.

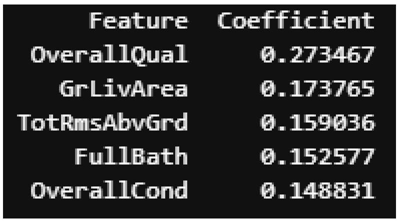
And, the changes in predictor variables are as following:

Before



Feature	Coefficient
OverallQual	0.311505
GrLivArea	0.216357
OverallCond	0.182909
TotRmsAbvGrd	0.173895
1stFlrSF	0.171709

After



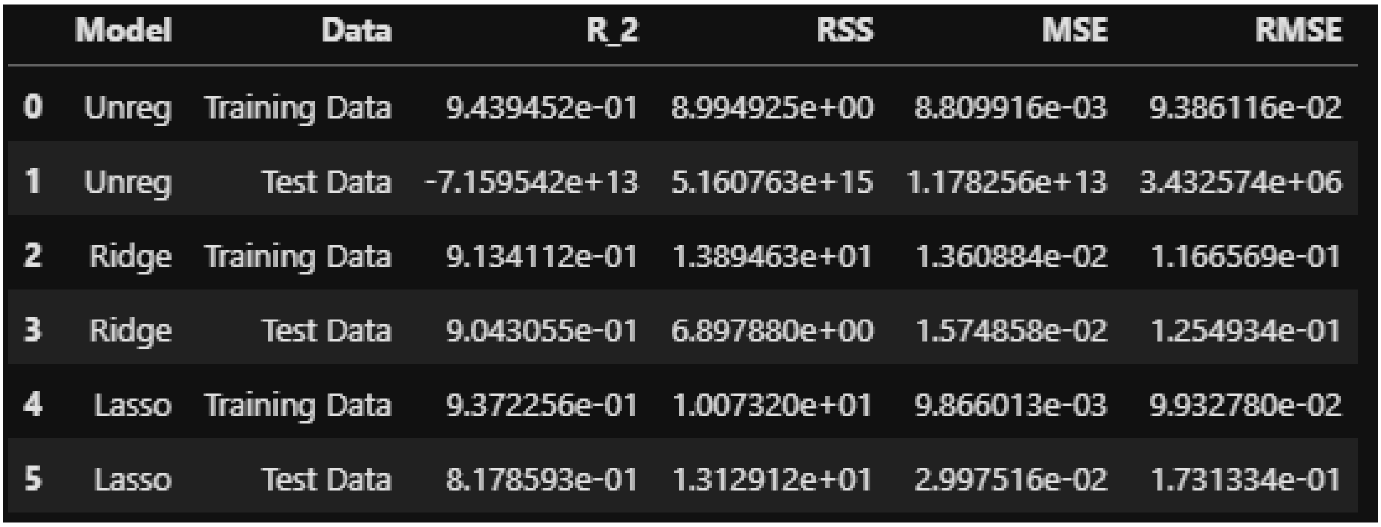
Feature	Coefficient
OverallQual	0.273467
GrLivArea	0.173765
TotRmsAbvGrd	0.159036
FullBath	0.152577
OverallCond	0.148831

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:

This is the final evaluation metrics of different models



	Model	Data	R_2	RSS	MSE	RMSE
0	Unreg	Training Data	9.439452e-01	8.994925e+00	8.809916e-03	9.386116e-02
1	Unreg	Test Data	-7.159542e+13	5.160763e+15	1.178256e+13	3.432574e+06
2	Ridge	Training Data	9.134112e-01	1.389463e+01	1.360884e-02	1.166569e-01
3	Ridge	Test Data	9.043055e-01	6.897880e+00	1.574858e-02	1.254934e-01
4	Lasso	Training Data	9.372256e-01	1.007320e+01	9.866013e-03	9.932780e-02
5	Lasso	Test Data	8.178593e-01	1.312912e+01	2.997516e-02	1.731334e-01

Since, the R square value of test data in lasso model is quite less as compare to the test data on ridge model which indicates that 'Lasso' must have dropped some important variables that must have good correlation with the target variable. This is also quite evident from the MSE values as it increases for test data on Lasso Model.

Also, the difference b/w MSE of training & test data in Ridge regression is way better than that in Lasso regression.

Therefore, Ridge Regression should be preferred for this problem.

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?

Solution:

After removing the earlier top 5 predictors, these are the new top 5 predictors

1. 1stFlrSF
2. OverallQual
3. 2ndFlrSF
4. GarageCars
5. TotRmsAbvGrd

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:

To ensure that a model is robust and generalizable, we can follow these practices:

1. **Use Sufficient and Representative Data:** Make sure your training data is sufficient in quantity and representative of the real-world scenarios you want your model to perform well on. Collecting diverse and comprehensive data can help the model capture the underlying patterns and variations present in the problem domain.

2. **Split Data into Train and Test Sets:** The training set is used to train the model, while the test set is used to evaluate its performance on unseen data. This allows you to assess how well the model generalizes to new data.

3. **Perform Cross-Validation:** Employ cross-validation techniques, such as k-fold cross-validation, to assess the model's performance more reliably. Cross-validation helps evaluate the model's average performance across multiple data partitions and reduces the impact of variations in a single train-test split.

4. **Avoid Overfitting:** Overfitting occurs when a model performs extremely well on the training data but fails to generalize to new data. To mitigate overfitting, use techniques such as regularization (e.g., Ridge or Lasso regression), feature selection, or early stopping during training.

5. **Hyperparameter Tuning**: Fine-tune the model's hyperparameters using techniques like grid search or random search. Hyperparameter tuning helps identify the optimal configuration that maximizes the model's performance on unseen data.

6. **Evaluate on Multiple Metrics:** Besides accuracy, assess the model's performance using other relevant evaluation metrics, such as MSE, RMSE, precision, recall, F1 score, or area under the ROC curve. Different metrics capture different aspects of model performance and can provide a more comprehensive understanding of its generalization ability.

7. **Validate on Unseen Data:** Once you have selected a model and finalized its hyperparameters, validate it on truly unseen data, preferably from a different source or time period. This helps confirm the model's ability to generalize beyond the specific training and test sets used during development.

The implications of ensuring model robustness and generalizability for the accuracy of the model are as follows:

- Model accuracy might decrease slightly. By focusing on robustness and generalizability, you prioritize the model's ability to perform well on unseen data over maximizing accuracy on the training set. This trade-off aims to prevent overfitting and create a more reliable and trustworthy model that can handle new inputs effectively.

- A robust and generalizable model tends to have more consistent and reliable performance across different datasets.

Overall, focusing on model robustness and generalizability helps create a more reliable and trustworthy model that can effectively handle new data and perform well in real-world scenarios.