Advanced Regression Assignment – Part 2

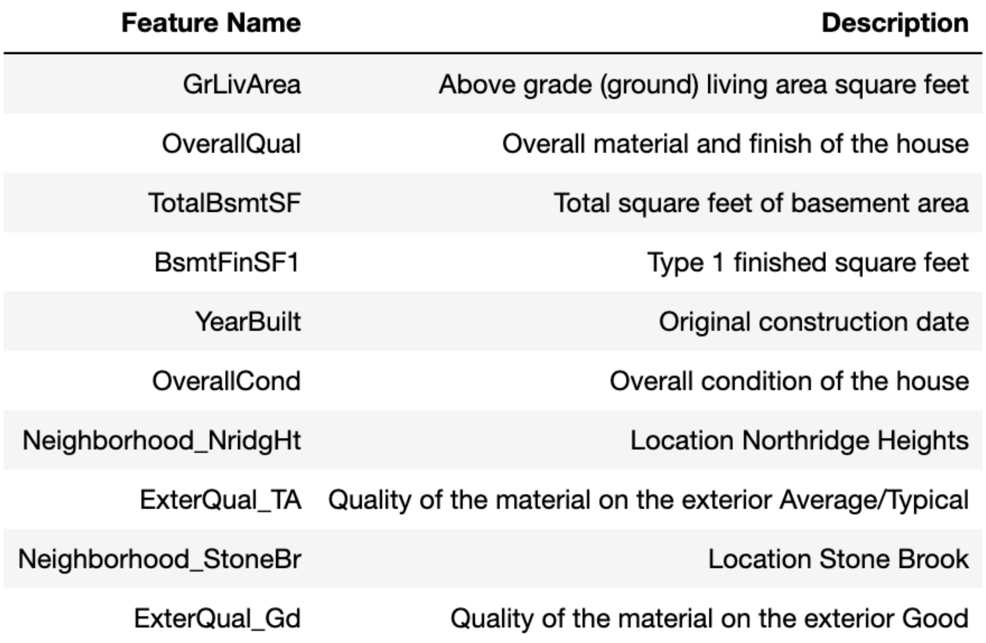
Question 1

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 Values of alpha for Lasso Regression: 100.0

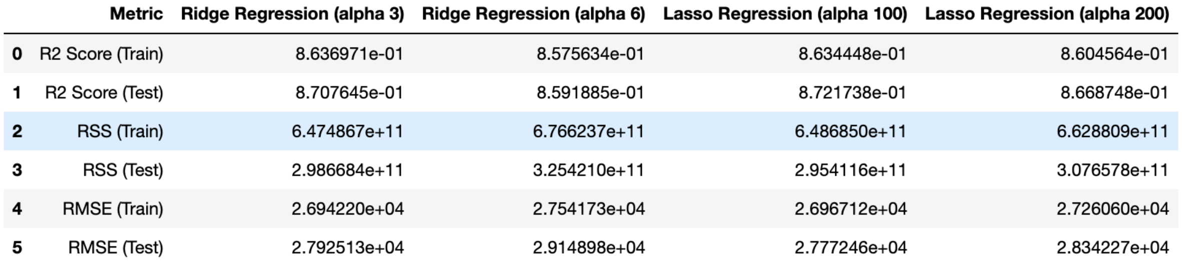
Optimal Values f of alpha or Ridge Regression: 3.0

Significant Features



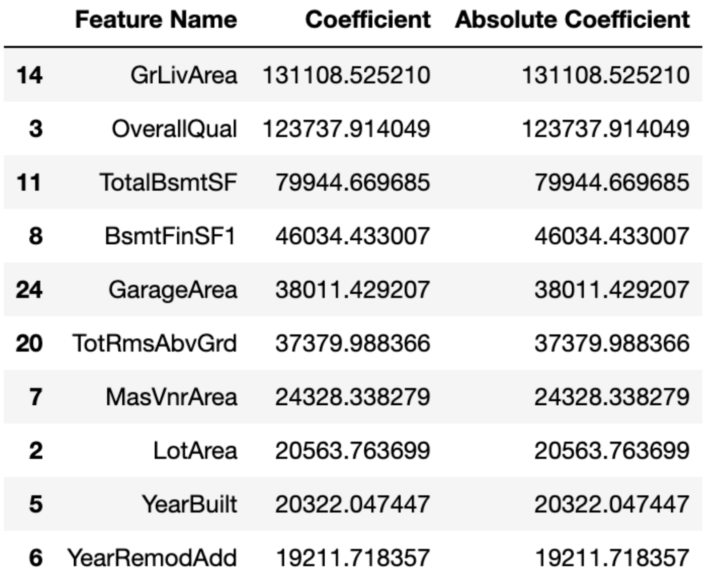
Feature Name	Description
GrLivArea	Above grade (ground) living area square feet
OverallQual	Overall material and finish of the house
TotalBsmtSF	Total square feet of basement area
BsmtFinSF1	Type 1 finished square feet
YearBuilt	Original construction date
OverallCond	Overall condition of the house
Neighborhood_NridgHt	Location Northridge Heights
ExterQual_TA	Quality of the material on the exterior Average/Typical
Neighborhood_StoneBr	Location Stone Brook
ExterQual_Gd	Quality of the material on the exterior Good

After doubling the values for alpha (ie) 200 for Lasso and 6.0 for Ridge, this is how the comparison looks like,



	Metric	Ridge Regression (alpha 3)	Ridge Regression (alpha 6)	Lasso Regression (alpha 100)	Lasso Regression (alpha 200)
0	R2 Score (Train)	8.636971e-01	8.575634e-01	8.634448e-01	8.604564e-01
1	R2 Score (Test)	8.707645e-01	8.591885e-01	8.721738e-01	8.668748e-01
2	RSS (Train)	6.474867e+11	6.766237e+11	6.486850e+11	6.628809e+11
3	RSS (Test)	2.986684e+11	3.254210e+11	2.954116e+11	3.076578e+11
4	RMSE (Train)	2.694220e+04	2.754173e+04	2.696712e+04	2.726060e+04
5	RMSE (Test)	2.792513e+04	2.914898e+04	2.777246e+04	2.834227e+04

As we can observe, there is a slight reduction in the R2 scores after doubling the alpha values. The feature set remains similar though.

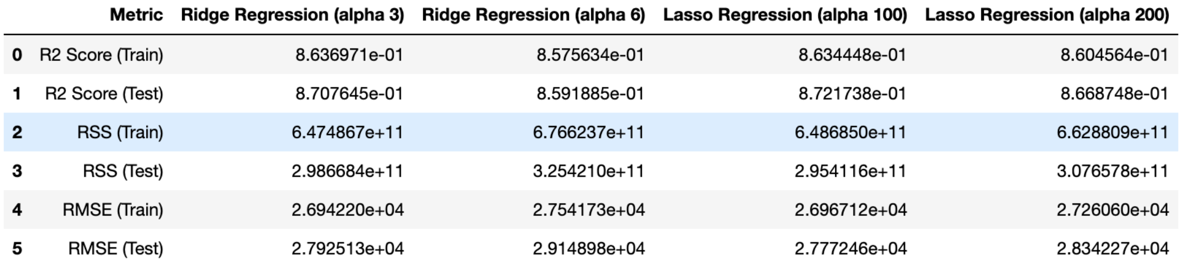


	Feature Name	Coefficient	Absolute Coefficient
14	GrLivArea	131108.525210	131108.525210
3	OverallQual	123737.914049	123737.914049
11	TotalBsmtSF	79944.669685	79944.669685
8	BsmtFinSF1	46034.433007	46034.433007
24	GarageArea	38011.429207	38011.429207
20	TotRmsAbvGrd	37379.988366	37379.988366
7	MasVnrArea	24328.338279	24328.338279
2	LotArea	20563.763699	20563.763699
5	YearBuilt	20322.047447	20322.047447
6	YearRemodAdd	19211.718357	19211.718357

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?

Although both models have similar R2 scores on the test data, I would decide to use the Lasso model because it has slightly better scores as shown in the tables. Furthermore, Lasso is preferred over the Ridge model because it can eliminate coefficients to absolute zero, which Ridge cannot. This means that Lasso will likely have a lower number of predictor variables, making the model less complex and easier to interpret. By using fewer variables, we can still make accurate predictions using the Lasso model.

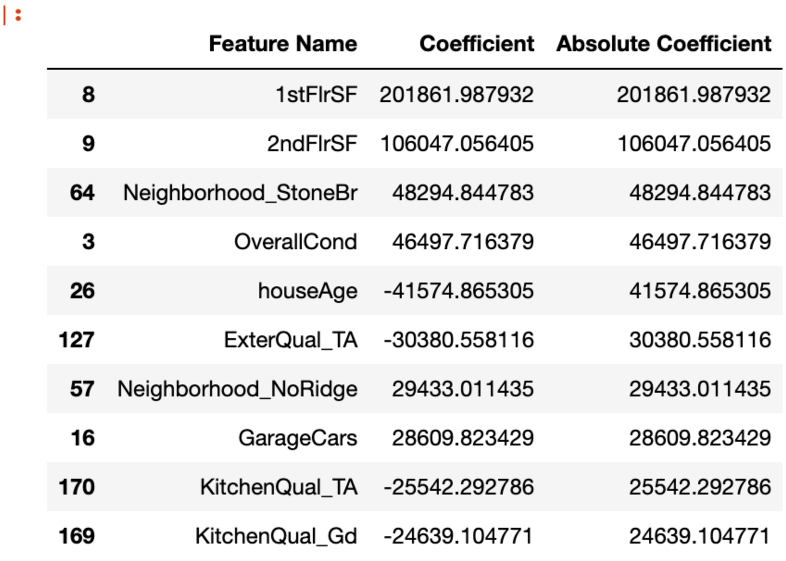


	Metric	Ridge Regression (alpha 3)	Ridge Regression (alpha 6)	Lasso Regression (alpha 100)	Lasso Regression (alpha 200)
0	R2 Score (Train)	8.636971e-01	8.575634e-01	8.634448e-01	8.604564e-01
1	R2 Score (Test)	8.707645e-01	8.591885e-01	8.721738e-01	8.668748e-01
2	RSS (Train)	6.474867e+11	6.766237e+11	6.486850e+11	6.628809e+11
3	RSS (Test)	2.986684e+11	3.254210e+11	2.954116e+11	3.076578e+11
4	RMSE (Train)	2.694220e+04	2.754173e+04	2.696712e+04	2.726060e+04
5	RMSE (Test)	2.792513e+04	2.914898e+04	2.777246e+04	2.834227e+04

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?

After rebuilding the model post the removal of the top 5 variables, the significant predictor variables are as below:



| :

	Feature Name	Coefficient	Absolute Coefficient
8	1stFlrSF	201861.987932	201861.987932
9	2ndFlrSF	106047.056405	106047.056405
64	Neighborhood_StoneBr	48294.844783	48294.844783
3	OverallCond	46497.716379	46497.716379
26	houseAge	-41574.865305	41574.865305
127	ExterQual_TA	-30380.558116	30380.558116
57	Neighborhood_NoRidge	29433.011435	29433.011435
16	GarageCars	28609.823429	28609.823429
170	KitchenQual_TA	-25542.292786	25542.292786
169	KitchenQual_Gd	-24639.104771	24639.104771

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?

To make sure that a model is robust and generalizable, we need to evaluate its performance on data that was not used to train it. The following steps can help ensure that a model is robust and generalizable:

Use a diverse dataset: The model should be trained on a dataset that is diverse, including a range of values for each feature. This will ensure that the model learns the underlying patterns and relationships in the data rather than memorizing the training set.

Split the data into training and testing sets: The data should be split into training and testing sets. The model should be trained on the training set and evaluated on the testing set. This will help determine if the model can generalize to new data.

Use cross-validation: Cross-validation is a technique used to evaluate the robustness of the model. In cross-validation, the data is split into several folds, and the model is trained on one fold and tested on the others. This process is repeated multiple times, and the results are averaged. This ensures that the model is not biased towards any specific part of the data.

Evaluate on unseen data: After the model is trained, it should be evaluated on a completely unseen dataset to ensure that it is generalizable.

The implications of having a robust and generalizable model are significant. A robust model will perform well on a variety of datasets, even those that are not similar to the training dataset. This means that the model will not be biased towards any specific type of data, and its accuracy will not be affected by the noise or randomness in the training data.
Additionally, a generalizable model will be able to make accurate predictions on new, unseen data. This is crucial in real-world scenarios where the model is expected to perform well on data that it has not seen before.

In contrast, if the model is not robust and generalizable, it may perform well on the training data but may fail to make accurate predictions on new data. This is known as overfitting, where the model has memorized the training data and is not able to generalize well. Therefore, it is important to ensure that a model is robust and generalizable to ensure its accuracy and usefulness in practical applications.