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:

Optimal value for alpha in Ridge Regression is 10

Optimal value for alpha in Lasso Regression is 500

If we double the alpha for ridge :

- Coefficients values are increasing as we doubled alpha.
- r2_score of train data is slightly dropped.

If we double the alpha for lasso :

2_score of train data is dropped to ~ 79.6% on train and test data.

Most important predictor variables after the change is implemented :

- Neighborhood_NoRidge , Neighborhood_NridgHt , Neighborhood_Veenker , 2ndFlrSF , Neighborhood_Somerst

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:

We can go ahead with lasso regression as it has further removed 8 variables from total features given by

RFE to make it further less complex and generalized. Also the model accuracy is not affected much. It means model will be more simpler and generalized.

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?

Answer:

After excluding the top 5 features, model r2 score has dropped to 75.4% on train set and r2 score has

dropped to 76.9% on test set.

5 Top variables now are:

OverallQual , HouseStyle_2Story ,1stFlrSF , KitchenQual , MSZoning_RL

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?

Answer:

There are several steps that can help ensure a model is robust and generalizable:

1. **Use a diverse and representative dataset**: The dataset used to train the model should be diverse enough to capture a broad range of scenarios and should be representative of the target population. This can help ensure that the model is not overfitting to specific examples in the data and is able to generalize to new, unseen examples.
2. **Regularization**: Regularization techniques like dropout, weight decay, and early stopping can help prevent overfitting and improve model generalization.
3. **Cross-validation**: Cross-validation can be used to evaluate the model's performance on multiple subsets of the data, which can help assess its generalization performance.
4. **Hyperparameter tuning**: Proper hyperparameter tuning can improve model performance and generalization.
5. **Data augmentation**: Data augmentation can be used to artificially increase the size of the training data and expose the model to a wider range of scenarios.

The implications of a model being robust and generalizable are that it is more likely to perform well on new, unseen data. This is because the model has learned to capture the underlying patterns in the data, rather than just memorizing specific examples. Additionally, a robust and generalizable model is less likely to be impacted by changes in the input data distribution, which is important in real-world applications where the data may evolve over time.

However, improving the robustness and generalization of a model may come at the expense of accuracy on the training data. For example, regularization techniques like dropout can reduce overfitting but may also reduce accuracy on the training set. Similarly, hyperparameter tuning may involve a tradeoff between improving generalization and reducing accuracy on the training set. Therefore, it's important to find a balance between model accuracy and generalization when building machine learning models.