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?

In the context of ridge regression, as we observe the curve plotting negative mean absolute error against alpha, we notice a decreasing trend in the error term as alpha increases from 0. However, the train error exhibits an increasing trend with higher alpha values. Upon reaching an alpha value of 1, the test error reaches its minimum, leading us to select an alpha value of 2 for our ridge regression.

Regarding lasso regression, I've opted for a very small alpha value of 0.001. As we increment alpha, the model intensifies its penalization efforts, aiming to drive more coefficients towards zero. Initially, the negative mean absolute error stands at 0.4, corresponding to the alpha value.

Doubling the alpha value for our ridge regression, we arrive at an alpha value of 10. This results in the model applying more stringent penalties, thereby promoting greater generalization and simplicity. However, this heightened penalty leads to increased errors for both test and train datasets, as depicted in the graph.

Similarly, increasing the alpha value for lasso regression intensifies the penalization, driving more variable coefficients towards zero. This process also coincides with a decrease in the R-squared value, indicating a reduction in model fit.
The most important variable after the changes has been implemented for ridge regression are as
follows:-
1. MSZoning_FV
2. MSZoning_RL
3. Neighborhood_Crawfor
4. MSZoning_RH
5. MSZoning_RM
6. SaleCondition_Partial
7. Neighborhood_StoneBr
8. GrLivArea
9. SaleCondition_Normal
10. Exterior1st_BrkFace

The most important variable after the changes has been implemented for lasso regression are as
follows:-
1. GrLivArea
2. OverallQual

3. OverallCond
4. TotalBsmtSF
5. BsmtFinSF1
6. GarageArea

7. Fireplaces
8. LotArea
9. LotArea
10. LotFrontage

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?

Regularizing coefficients is essential for improving prediction accuracy while also reducing variance and enhancing model interpretability. In ridge regression, the regularization strength, denoted by lambda, is determined through cross-validation. By penalizing the square of the coefficient magnitudes, ridge regression aims to minimize the residual sum of squares, thereby shrinking coefficients with larger values. Increasing lambda decreases model variance while maintaining bias constant. Unlike lasso regression, ridge regression retains all variables in the final model.

In contrast, lasso regression also employs lambda as a tuning parameter, determined via cross-validation, to penalize the absolute value of coefficient magnitudes. As lambda increases, lasso regression progressively shrinks coefficients towards zero, potentially eliminating some variables entirely. Lasso regression facilitates variable selection, treating variables with zero coefficients as non-contributors to the model. With small lambda values, lasso regression approximates simple linear regression, but as lambda increases, shrinkage occurs, leading to variable exclusion.

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?

Those 5 most important predictor variables that will be excluded are :-
1. GrLivArea
2. OverallQual
3. OverallCond
4. TotalBsmtSF
5. GarageArea

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?

The aim is to simplify the model as much as possible, even at the expense of reduced accuracy, to enhance its robustness and generalizability. This concept is closely tied to the Bias-Variance trade-off, where a simpler model tends to have higher bias but lower variance, ultimately leading to greater generalizability. This trade-off implies that a robust and generalizable model will exhibit consistent performance across both training and test datasets, with minimal changes in accuracy between them.

Bias refers to the error in the model resulting from its inability to capture the complexities present in the data. High bias indicates that the model struggles to learn intricate details, leading to poor performance on both training and testing datasets.

On the other hand, variance represents the error in the model stemming from its tendency to excessively fit the training data. High variance implies that the model performs exceptionally well on the training data, having been trained extensively on it, but fares poorly on unseen testing data.

Achieving a balance between bias and variance is crucial to prevent both overfitting and underfitting of the data. This balance ensures that the model captures the essential patterns in the data without becoming overly complex or overly simplistic, thus enhancing its ability to generalize to new, unseen data.