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 of lambda for Ridge Regression = **8**
- Optimal value of lambda for Lasso = **0.0001**

For Ridge :

```
OverallQual_8          1.01
GrLivArea              1.01
OverallQual_9          1.01
Neighborhood_Crawfor   1.00
Functional_Typ         1.00
OverallCond_9          1.00
TotalBsmtSF            1.00
OverallCond_8          1.00
2ndFlrSF               1.00
BsmtCond_Gd            1.00
```

For Lasso :

```
GrLivArea              1.01
OverallQual_8          1.01
TotalBsmtSF            1.01
Functional_Typ         1.00
KitchenAbvGr_1         1.00
Neighborhood_Crawfor   1.00
GarageArea             1.00
OverallQual_9          1.00
Foundation_PConc       1.00
BsmtFinSF1             1.00
```

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 –

The model we will choose to apply will depend on the use case.

If our primary goal is feature selection, then we will use **Lasso**.

If reduction of coefficient magnitude is one of our prime goals, then we will use **Ridge Regression**.

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 –

```
2ndFlrSF             0.01
1stFlrSF             0.01
Functional_Typ       0.01
TotalBsmtSF          0.01
Exterior1st_BrkFace  0.00
```

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 –

To ensure that a model is robust and generalizable, there are several strategies and considerations to keep in mind:

Train with Sufficient and Diverse Data: A robust and generalizable model requires a diverse and representative training dataset. The dataset should cover a wide range of scenarios and variations that the model is likely to encounter in real-world situations. It is important to ensure that the training data adequately represents the target population and includes relevant edge cases or outliers.

Cross-Validation: Cross-validation techniques, such as k-fold cross-validation, can help assess the model's performance on different subsets of the data. By splitting the data into multiple folds and training and evaluating the model on different combinations, it provides a better estimate of how the model will perform on unseen data.

Regularization: Regularization techniques, such as L1 or L2 regularization, help prevent overfitting by adding a penalty term to the model's loss function. This encourages the model to be less sensitive to individual data points and reduces its complexity, making it more robust and generalizable.

Feature Engineering: Thoughtful feature engineering can help improve the model's ability to generalize. It involves selecting relevant features, removing noise or redundant information, and transforming variables to enhance their predictive power. Careful feature selection and transformation can improve the model's ability to generalize beyond the training data.

Hyperparameter Tuning: Selecting appropriate hyperparameters for the model is crucial for achieving good generalization. Hyperparameters, such as learning rate, regularization strength, or the number of layers in a neural network, can significantly impact the model's performance. Hyperparameter tuning techniques, such as grid search or random search, can help find the optimal set of hyperparameters that balance model complexity and performance.

Evaluation on Unseen Data: Finally, it is important to evaluate the model's performance on unseen data, such as a separate validation or test dataset. This provides insights into how well the model generalizes to new examples and helps identify any issues or limitations that need to be addressed.