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 to 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 alpha for ridge and lasso regression are:

- Optimal Value of lambda for Ridge: 10
- Optimal Value of lambda for Lasso: 0.001

  If we double the value of alpha for both ridge and lasso: In case of ridge that will lower the coefficients and in case of Lasso there would be higher no of cases where less important features coefficients turning 0.

  The most important predictor variable after the change is implemented are those which are significant.

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 : Optimal Value of alpha for ridge and lasso regression are:

- Optimal Value of lambda for ridge: 10
- Optimal Value of lambda for Lasso: 0.001

  Both the models have a decent score however we can choose Lasso regression as it results in model parameters such that coefficients of larger no of lesser important features become zero.

  Ridge: Train :90.9 Test :87.4 and Lasso : Train :89.8 Test :86.4

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 : On running the same notebook and removing the top 5 significant variables : We found below variables as next 5 significant.

-Lasso

```
The R2 Score of the model on the test dataset is 0.8756584353259769
The MSE of the model on the test dataset is 0.020459175540738877
The most important predictor variables are as follows:
```

	Lasso Co-Efficient
OverallQual_Excellent	0.215900
Neighborhood_NridgHt	0.143994
Neighborhood_NoRidge	0.137759
OverallQual_Very Good	0.133363
Neighborhood_Somerst	0.132407

-Ridge

```
The R2 Score of the model on the test dataset is 0.8732640216996035
The MSE of the model on the test dataset is 0.02085315263783861
The most important predictor variables are as follows:
```

	Ridge Co-Efficient
OverallQual_Excellent	0.222522
Neighborhood_NridgHt	0.185098
Neighborhood_NoRidge	0.184426
Neighborhood_StoneBr	0.167035
Neighborhood_Somerst	0.161296

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 :

Here are some changes you can make to your model:

• **Use a model that's resistant to outliers.** Tree-based models are generally not as affected by outliers, while regression-based models are. If you're performing a statistical test, try a non-parametric test instead of a parametric one.

• **Use a more robust error metric.** switching from mean squared error to mean absolute difference (or something like Huber Loss) reduces the influence of outliers. I explain a bit about why this is the case at Why is the median a measure of central tendency? It doesn't have anything to do with any other values of the data set, so how does it "describe" the data set?

Here are some changes you can make to your data:

• **Winsorize your data.** Artificially cap your data at some threshold. See What are some applications of winsorization?

• **Transform your data.** If your data has a very pronounced right tail, try a log transformation.

• **Remove the outliers**. This works if there are very few of them and you're fairly certain they're anomalies and not worth predicting.

As Per, Occam's Razor— given two models that show similar 'performance' in the finite training or test data, we should pick the one that makes fewer on the test data due to following reasons:-

▪ Simpler models are usually more 'generic' and are more widely applicable

▪ Simpler models require fewer training samples for effective training than the more complex ones and hence are easier to train.

▪ Simpler models are more robust.

  o Complex models tend to change wildly with changes in the training data set o Simple models have low variance, high bias and complex models have low bias, high variance

o Simpler models make more errors in the training set. Complex models lead to overfitting — they work very well for the training samples, fail miserably when applied to other test samples

Therefore, to make the model more robust and generalizable, make the model simple but not simpler which will not be of any use.

Regularization can be used to make the model simpler. Regularization helps to strike the delicate balance between keeping the model simple and not making it too naive to be of any use. For regression, regularization involves adding a regularization term to the cost that adds up the absolute values or the squares of the parameters of the model.

Also, Making a model simple leads to Bias-Variance Trade-off:

• A complex model will need to change for every little change in the dataset and hence is very unstable and extremely sensitive to any changes in the training data.

• A simpler model that abstracts out some pattern followed by the data points given is unlikely to change wildly even if more points are added or removed.

Bias quantifies how accurate is the model likely to be on test data. A complex model can do an accurate job prediction provided there is enough training data. Models that are too naïve, for e.g., one that gives same answer to all test inputs and makes no discrimination whatsoever has a very large bias as its expected error across all test inputs are very high. Variance refers to the degree of changes in the model itself with respect to changes in the training data. Thus accuracy of the model can be maintained by keeping the balance between Bias and Variance as it minimizes the total error as shown in the below graph.