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 1 – optimal value of ridge and lasso were 50 and 100 respectively. On doubling alpha ridge regression would increase the penalty, reduce the coefficient magnitude and trying to further generalize the model by trading off train and test score. Lasso also similarly tries to do this but in case of lasso it would try to make more features coefficient 0.

New top 10 features after doubling alpha.

Ridge

1. BsmtFullBath
2. 2ndFlrSF
3. 1stFlrSF
4. BsmtFinSF2
5. Fireplaces
6. Neighborhood_Timber
7. LowQualFinSF
8. WoodDeckSF
9. OverallQual_10
10. OverallCond_2

Lasso

1. 1stFlrSF
2. GarageArea
3. BsmtFullBath
4. EnclosedPorch
5. OverallCond_2
6. OverallQual_10
7. BedroomAbvGr
8. OverallQual_9
9. Exterior1st_CBlock
10. GarageType_Basment

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: Lasso regression, as it is also performing feature elimination and difference between test and train score is lower as compared to 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: Next 5 important predictors are

Exterior1st_CBlock

OverallQual_10

Neighborhood_NridgHt

OpenPorchSF

3SsnPorch

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: Model should be as simple as possible but not too simple. This means a simple model is more generalizable, but it trades off accuracy for simplicity. A complex model with a lot of features is more accurate but not very generalizable and very sensitive to input variations. This is also known as bias variance trade-off.

The bias is known as the difference between the prediction of the values by the model and the correct value. Being high in biasing gives a large error in training as well as testing data. It recommended that an algorithm should always be low-biased to avoid the problem of underfitting. By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data in the data set. Such fitting is known as the underfitting. This happens when the hypothesis is too simple or linear in nature.

The variability of model prediction for a given data point which tells us the spread of our data is called the variance of the model.

If the algorithm is too simple (hypothesis with linear equation) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex (hypothesis with high degree equation) then it may be on high variance and low bias. In the latter condition, the new entries will not

perform well. Well, there is something between both of these conditions, known as a Trade-off or Bias Variance Trade-off. This tradeoff in complexity is why there is a tradeoff between bias and variance.