Subjective Questions – Advanced Regression Surprise Housing Assignment part II

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?

For my analysis, optimal value of ridge is 3, while for Lasso it is 0.0001 as per the GridSearchCV.
I built the Ridge and Lasso models with all features, by doubling the Alpha value which we got earlier i.e. New alpha for Ridge = 6 and new alpha for Lasso = 0.0002. Below are the test metrics for the same.

Metric	*Ridge (alpha =3)*	*Ridge (alpha = 6)*	*Lasso (Alpha = 0.0001)*	*Lasso (Alpha = 0.0002)*
R2 Score (Test)	0.925347	0.920425	0.923935	0.920797
RSS (Test)	0.263267	0.280622	0.268244	0.279312
MSE (Test)	0.026015	0.026859	0.026260	0.026796

Table 1

As we can see in the above table that when the lambda is increased the general tendency is a slight reduction in the model accuracy (R2) and increase in error (RSS and MSE). Same trend is observed with the train values also.

After the change is implemented, it is noticed that there is no change in the top 10 predictor variables for the Ridge regression remain unchanged, while for Lasso regression there are some changes in the top 10 variables. Also the coefficients have shrinked when the alpha value is increased.

	Before	After
Ridge	**Alpha = 3** GrLivArea 0.079955 OverallQual_10 0.078361 1stFlrSF 0.075178 BsmtFinSF1 0.059433 TotalBsmtSF 0.059208 2ndFlrSF 0.054260 Neighborhood_StoneBr 0.039718 PoolQC_NoPool 0.039645 LotArea 0.034898 FullBath_3 0.034509	**Alpha = 6** GrLivArea 0.062239 OverallQual_10 0.059970 1stFlrSF 0.059606 BsmtFinSF1 0.052568 TotalBsmtSF 0.051772 2ndFlrSF 0.041007 Neighborhood_StoneBr 0.033902 FullBath_3 0.033828 LotArea 0.032810 PoolQC_NoPool 0.030416
Lasso	**Alpha = 0.0001** GrLivArea 0.294979 OverallQual_10 0.156764 TotalBsmtSF 0.094439 OverallQual_9 0.063531 PropertyAge 0.053307 BsmtFinSF1 0.046825 PoolQC_NoPool 0.040641 Neighborhood_StoneBr 0.035819 Neighborhood_Crawfor 0.035264 OverallQual_8 0.033674	**Alpha = 0.0002** GrLivArea 0.281609 OverallQual_10 0.150156 TotalBsmtSF 0.096418 OverallQual_9 0.064072 BsmtFinSF1 0.049500 PropertyAge 0.038642 OverallQual_8 0.034400 LotArea 0.032980 Neighborhood_Crawfor 0.032256 SaleCondition_Partial 0.030395

Table 2

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?

From the table 3 we can see that in terms of accuracy, best performance is given by the Ridge regression model (alpha = 3), but Lasso (alpha = 0.0001) is extremely close with an added advantage that it can reduce the number of features if required by fine tuning the alpha value which is not possible with Ridge. Hence, I will be using Lasso (with alpha 0.0001) in this case.

Metric	Ridge (alpha =3)	Ridge (alpha = 6)	Lasso (Alpha = 0.0001)	Lasso (Alpha = 0.0002)
R2 Score (Test)	0.925347	0.920425	0.923935	0.920797
RSS (Test)	0.263267	0.280622	0.268244	0.279312
MSE (Test)	0.026015	0.026859	0.026260	0.026796

Table 3

Another advantage of using Lasso is that we can bring down the number of predictor variables by increasing the alpha value without compromising too much on the error in the model. Using lower number of predictor variable without compromising too much on the errors is a great because it helps to keep the model simple and interpretable.

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?

Top 10 variables for Lasso (0.0001) models are as follows:

```
GrLivArea                0.294979
OverallQual_10           0.156764
TotalBsmtSF              0.094439
OverallQual_9            0.063531
PropertyAge              0.053307
BsmtFinSF1               0.046825
PoolQC_NoPool            0.040641
Neighborhood_StoneBr     0.035819
Neighborhood_Crawfor     0.035264
OverallQual_8            0.033674
```

Now as per the question, top 5 variable are not available and we need to build a new model. So let's drop the top 5 variables from the data and rebuild the model again. Variables to be dropped: 'GrLivArea', 'OverallQual_10', 'TotalBsmtSF', 'OverallQual_9','PropertyAge'

But we also notice that OverallQual is the categorical variable, since this variable isn't present in the incoming data, we should remove all the corresponding dummy variables related to this predictor variable and rebuild the model again.

After remodelling the new top 5 features are '1stFlrSF', '2ndFlrSF','BsmtFinSF1','PoolQC_NoPool','BsmtUnfSF'

And below are the metrices for the new model:

```
R2 train: 0.9272540125342503
R2 test: 0.9157577287725094
RSS train: 0.608234232577668
R2 test: 0.2970826825765307
MSE train: 0.0006706000359180462
MSE test: 0.0007637086955694876
```

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: A model needs to be made robust so that they are not impacted by extreme range of the incoming data i.e. outliers in the training data. The model should generalize such that the performance of the model on the unseen data is as close to the training data i.e. it shouls identify the underlying patterns in the data and also perform well on the unseen real world test data.

Outlier Analysis and treatment - Too much importance should not given to the outliers so that the accuracy predicted by the model is high. To achieve this we should make sure outliers are processed and treated properly i.e. try to remove outliers from the training data before using the same for model building. This would help increase the accuracy of the predictions made by the model. We can use the general outlier removal formula to achieve the same i.e OL [<Q1-1.5*IQR, >Q3 +1.5*IQR.]

Also to make sure that the model is generalizing well and not overfitting, we should always validate the model on the test data which the model hasn't seen i.e. the unseen data. To achieve this we must keep a set of the test data aside for validation right at the start of the whole process. When we are done with the model building and achieve a good accurancy, then we should be using the test data to test the accuaracy of the model. If the Test accuracy is very close to the Training accuracy we are good, but if the test accuracy is very low compared to the training accuracy it means the model is not generalizing and it is overfitting. We need to review the modeling process again and see how we can improve the same.