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?

Optimal value of alpha:

• Optimal alpha (lambda) value for Ridge Regression model is: 20

• Optimal alpha (lambda) value for Lasso Regression model is: 500

Effect of choosing double the value of optimal alpha:

Before explaining the second part of the question, let's see the cost functions of Ridge and Lasso.

Cost function for Ridge:

$$L_{hridge}(\hat{\beta}) = \sum_{i-1}^{n}(y_i - x_i'\hat{\beta})^2 + \lambda \sum_{j-1}^{m} w_j \hat{\beta}_j^2.$$

Cost function for Lasso:

$$L_{lasso}(\hat{\beta}) = \sum_{i-1}^{n}(y_i - x_i'\hat{\beta})^2 + \lambda \sum_{j-1}^{m} |\hat{\beta}_j|.$$

So, here it can be seen that in both the cases penalty term increases with higher value of beta co□efficient. Ridge imposes more aggressive penalty as it uses sum of square of all beta coefficients (L2 norm) as shrinking penalty. Where Lasso uses sum of absolute values of all beta coefficients (L1 norm) as shrinking penalty. In both equations these norms are multiplied by lambda or alpha. This alpha is a hyperparameter and its optimal value can be obtained by performing cross validation. Value of alpha can be any number >= 0. If we increase the value of alpha then shrinking penalty will be higher, so Ridge and Lasso both will try to shrink values of beta coefficients towards zero, so our model will be simpler. That means it will increase the bias where variance will be reduced. If we increase the value of alpha to a very large number, then all coefficients of Lasso become 0 and for Ridge coefficients become close to zero (as they cannot be exact 0 in Ridge). That means the model will have very high bias and low variance and it may result in underfitting. That means model will fail to learn the underlying data pattern in training dataset. If we reduce the value of alpha then

shrinking penalty will be lower, so model bias will reduce, and variance will increase. Now if we put value of alpha as 0, then the cost function of both Ridge and Lasso become OLS cost function (i.e., RSS) and we will get exact same model as we get using OLS. So, reducing value of alpha will reduce the effect of shrinking penalty may lead to possible overfitting for very low or close to zero value of alpha. So, we need to find the optimal value of alpha by performing hyperparameter tunning.

From the above equations it is clear that once we double the alpha the coefficients(betas) are going to shrink.

Below is the result explaining the same:

Ridge Before

```
GrLivArea          22702.201150837
OverallQual        10778.040152010
YearBuilt          10526.286844199
TotalBsmtSF         9537.274410841
TotRmsAbvGrd        7736.950090833
BsmtFinSF1          7523.644642894
BsmtExposure_Gd     6931.728818765
KitchenQual         6640.634487866
MSSubClass_60       6367.448679206
OverallCond         6192.442719838
dtype: float64
```

Ridge After:

```
GrLivArea          19375.887941036
OverallQual        10809.410794998
YearBuilt           9902.195217719
TotalBsmtSF         8527.675168471
TotRmsAbvGrd        7631.170553250
BsmtFinSF1          7440.801495025
KitchenQual         6812.199309351
BsmtExposure_Gd     6699.451030748
Neighborhood_StoneBr    6011.840273296
OverallCond         5962.989638737
dtype: float64
```

Lasso Before:

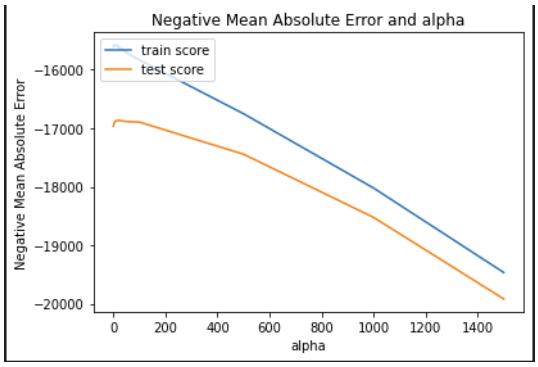
```
GrLivArea              23858.018589398
OverallQual            10733.906436942
YearBuilt               7927.886389835
BsmtFinSF1              7652.977984802
TotalBsmtSF             7547.986101693
SaleType_New            6482.498877295
KitchenQual             6423.326167538
Neighborhood_NridgHt    6216.461376297
OverallCond             6004.891166129
BsmtExposure_Gd         5884.348041558
dtype: float64
```

Lasso After:

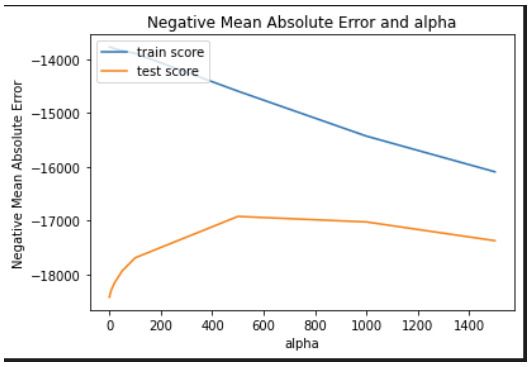
```
GrLivArea              23837.975962510
OverallQual            11631.104306013
BsmtFinSF1              7359.474995311
TotalBsmtSF             7061.048681872
SaleType_New            6631.406190798
KitchenQual             6578.963566499
YearBuilt               6284.220243573
Neighborhood_NridgHt    5797.236831175
BsmtExposure_Gd         5304.424009467
Neighborhood_StoneBr    5263.985423388
dtype: float64
```

We can also see that this relation in a better way by using this graph below:

Ridge



Lasso



As we can see in both the graphs the test score rises initially and then dips with increase in alpha, this explains the concept stated above, if the alpha Is too large then the bias is high, and the model is underfitting. And if the alpha is too low then the variance is too high, and the model is overfitting.

We can also see the change in R2Score and MSE by doubling alpha:

	Metric	Ridge_20	Ridge_40	Lasso_500	Lasso_1000
0	R2 train	0.912379900	0.911099246	0.917573379	0.908857234
1	R2 test	0.887587323	0.888414440	0.893419314	0.894588133
2	RSS train	496558817274.792968750	503816511028.242431641	467126439685.103027344	516522396195.594970703
3	RSS test	259347245137.338836670	257439003579.719329834	245892261190.603607178	243195679450.121765137
4	MSE train	22250.249357824	22412.264104311	21580.761152687	22693.114898669
5	MSE test	24558.768866260	24468.252025066	23913.227885108	23781.743997412

As you can see in both the cases the training score for R2 decreased a bit, although the change is very minimal, and we can see drastic changes if we go higher.

The most important predictor variables for ridge and lasso after the changes are:

Ridge: (GrLivArea)

```
GrLivArea              19375.887941036
OverallQual            10809.410794998
YearBuilt               9902.195217719
TotalBsmtSF             8527.675168471
TotRmsAbvGrd            7631.170553250
BsmtFinSF1              7440.801495025
KitchenQual             6812.199309351
BsmtExposure_Gd         6699.451030748
Neighborhood_StoneBr    6011.840273296
OverallCond             5962.989638737
dtype: float64
```

Lasso: (GrLivArea)

```
GrLivArea              23837.975962510
OverallQual            11631.104306013
BsmtFinSF1              7359.474995311
TotalBsmtSF             7061.048681872
SaleType_New            6631.406190798
KitchenQual             6578.963566499
YearBuilt               6284.220243573
Neighborhood_NridgHt    5797.236831175
BsmtExposure_Gd         5304.424009467
Neighborhood_StoneBr    5263.985423388
dtype: float64
```

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?

As per Occam's Razor a model should not be unnecessarily complex.

	Metric	Linear Regression	Ridge Regression	Lasso Regression
0	R2 Score (Train)	9.136031e-01	9.123799e-01	9.175734e-01
1	R2 Score (Test)	8.828700e-01	8.875873e-01	8.934193e-01
2	RSS (Train)	4.896266e+11	4.965588e+11	4.671264e+11
3	RSS (Test)	2.702306e+11	2.593472e+11	2.458923e+11
4	MSE (Train)	2.209439e+04	2.225025e+04	2.158076e+04
5	MSE (Test)	2.506877e+04	2.455877e+04	2.391323e+04

As we can see that both Ridge and Lasso are giving good results but as seen Lasso is performing slightly better on the Test set, and it has eliminated many features and has 65 features so choosing Lasso makes more sense here, since both the models are comparable but Lasso has better Test performance.

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?

Initially top 5 variables were:

```
GrLivArea      23837.975962510
OverallQual    11631.104306013
BsmtFinSF1      7359.474995311
TotalBsmtSF     7061.048681872
SaleType_New    6631.406190798
dtype: float64
```

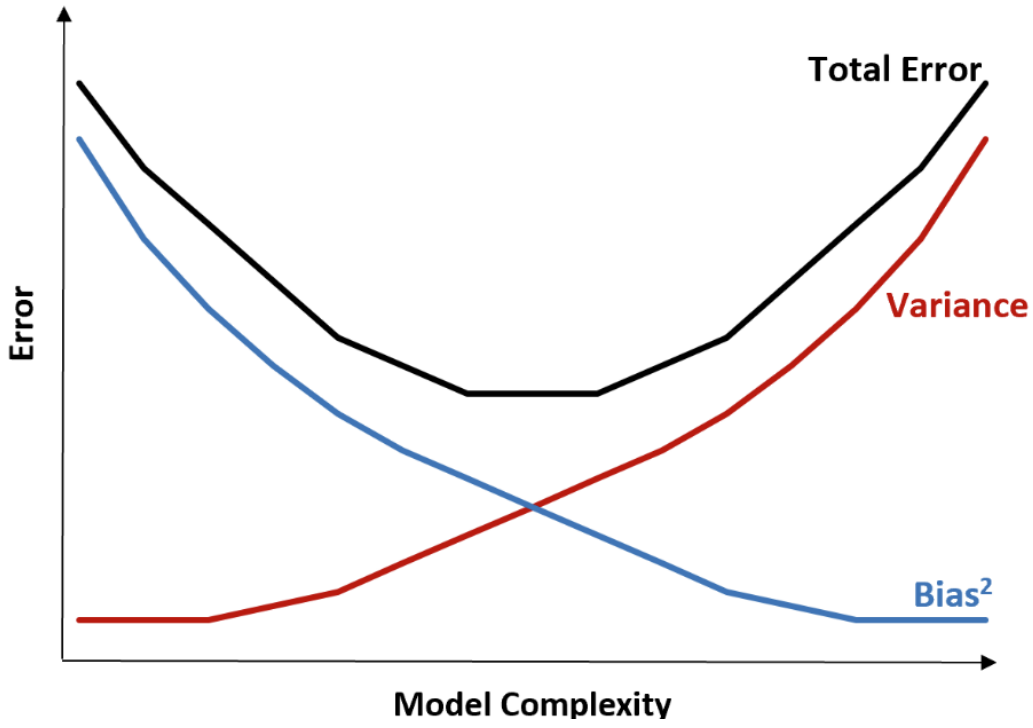
Since SaleType is a dummy variable we need to get rid of the entire category here, after removing the top5 features were:

```
1stFlrSF                 24163.027405885
2ndFlrSF                 19093.972727323
YearBuilt                 8584.979716613
SaleCondition_Partial     7993.459725429
KitchenQual               7550.447119591
dtype: float64
```

Question 4

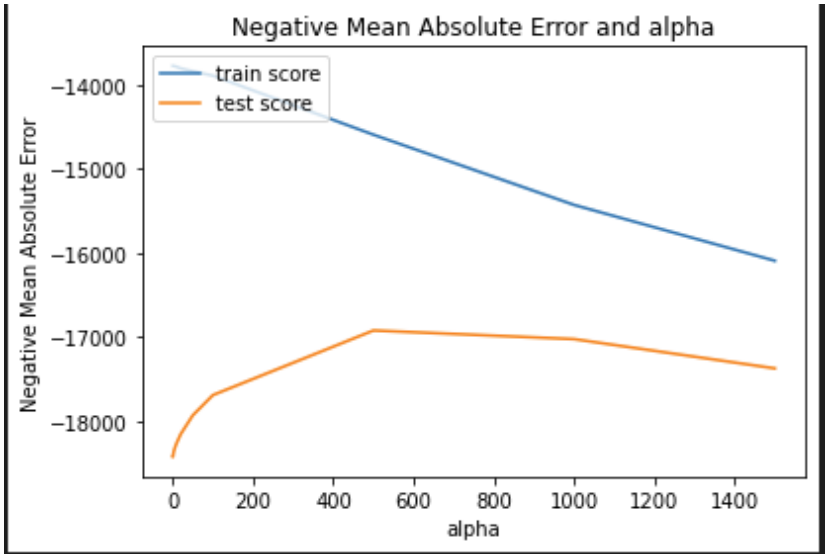
How can you make sure that a model is robust and generalisable? What are the implications of the same for the accuracy of the model and why?

Bias v/s variance trade-off plays an important role in explaining a model. Bias stands for how generalizable a model is, and variance is defined as the change in the model when the data changes. A good model should always be generalizable for new data at the same time it should not vary much with any new unseen data. Which means it should have low bias, low variance.



Overfitting: when a model is complex enough to understand whole training data, it doesn't perform very well on test data this problem is called overfitting. More accuracy implies model should be more complex so some bias needs to be there so that it is more generalizable. To avoid overfitting, regularization techniques are used.

Regularization: it is the process of penalizing models for using features. So this would add an extra alpha coefficient Ridge and lasso are two regularization techniques for a linear model. When model becomes generalizable, it performs better on test data while it drops it's performance in train data.



With change in alpha(increase in generalization) until a point, test score increases and it falls after a certain point. So any value until this point with good train & test score is considered as a good alpha value, this decision should be taken based on business priorities by considering test-train scores trade-off.