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?

The most optimal value for ridge regression is 7 and that for lasso regression is 0.0005.

When we increase alpha in ridge and lasso regression, it basically means that we are giving a little more importance to the regularization term when compared to the error term in the cost function. When we increase the term, we allow the model to make a little more error which making it more generalizable. In other words, we increase the bias and decrease the variance of the model.

Specifically increasing alpha by a factor of two would cause a slight decline in the train scores as the model attempts to generalize better. The original train scores for ridge and lasso are 0.897 and 0.895. The new train scores are 0.893 and 0.885 respectively.

While most of the important variables continue to remain so, few are replaced with new variables.

	Coefficient	Column		Coefficient	Column
0	-0.107257	Neighborhood_MeadowV	0	-0.070433	MSSubClass_Others
1	-0.082948	MSSubClass_Others	1	-0.069816	Neighborhood_MeadowV
2	-0.066063	Neighborhood_Edwards	2	-0.059610	Neighborhood_Edwards
3	-0.059518	BsmtFinType1_Unf	3	-0.051371	BsmtFinType1_Unf
4	-0.047123	HeatingQC_Others	4	-0.043332	MSZoning_RM
5	-0.043828	Neighborhood_BrDale	5	-0.037023	HeatingQC_Others
6	-0.041935	MSSubClass_50	6	-0.030634	MSSubClass_50
7	-0.039727	Neighborhood_IDOTRR	7	-0.030455	Neighborhood_BrDale
8	-0.039545	MSZoning_RM	8	-0.029749	Neighborhood_IDOTRR
9	-0.034665	MSSubClass_60	9	-0.029036	GarageType_Others

	Coefficient	Column		Coefficient	Column
106	0.052456	SaleCondition_Alloca	106	0.063008	BedroomAbvGr_4
107	0.053682	Neighborhood_NoRidge	107	0.074334	Neighborhood_NoRidge
108	0.059394	FullBath_Others	108	0.075441	FullBath_Others
109	0.059982	BedroomAbvGr_4	109	0.084262	SaleCondition_Alloca
110	0.062519	Neighborhood_NridgHt	110	0.084657	Neighborhood_NridgHt
111	0.078724	Neighborhood_StoneBr	111	0.088643	OverallQual
112	0.081967	GarageCars_3	112	0.098205	GarageCars_3
113	0.086435	Neighborhood_Crawfor	113	0.105585	Neighborhood_Crawfor
114	0.090336	OverallQual	114	0.112747	Neighborhood_StoneBr
115	0.107506	GrLivArea	115	0.113947	GrLivArea

Here, the first image shows the original parameters and the new image shows the parameters when alpha is doubled.

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?

We have built the model by using 100 odd features. Moreover, as a pre-step, we are not performing any PCA or RFE feature selection technique. So, in order to reduce the number of relevant features, we could proceed with lasso regression if we have a liberty of having a good computational power.

For this specific case, both the ridge and lasso give similar results though. We could leverage ElasticNet which gives us the capabilities of both ridge and lasso regression. However, lasso regression would be a better choice as the range of coefficients that we get in lasso is wider than ridge and also it takes care of correlated variables (which are plenty) and reduces them to zero.

```
-0.01260757,  0.01970547,  0.08803333,  0.04459262, -0.03200092
-0.01711198, -0.00785914,  0.03862849, -0.00961308,  0.
-0.00099419, -0.        ,  0.10555163,  0.01575165,  0.02885801
 0.01550444, -0.0254457 ,  0.00966094,  0.00608183, -0.00378417
 0.00557374,  0.0019622 , -0.03487599, -0.02807956, -0.07945079
 0.00799473, -0.04421362,  0.        , -0.        , -0.0158573
 0.        , -0.        , -0.01284311,  0.04123163,  0.0713268
 0.        ,  0.12829345, -0.04868254, -0.00557716, -0.01486012
-0.11380159,  0.00466605, -0.00386283, -0.        , -0.00270097
 0.08280386,  0.09657201, -0.        , -0.        , -0.
-0.        ,  0.0482166 ,  0.13593677,  0.00875695,  0.
 0.03050914, -0.02666732,  0.03686855,  0.00834014,  0.03011275
 0.0048891 , -0.        , -0.01192898,  0.        , -0.01298878
 0.        ,  0.        ,  0.01762769,  0.        , -0.01644477
 0.        , -0.        , -0.        ,  0.        ,  0.01593221
 0.        ,  0.04725698,  0.05922438, -0.00625411, -0.01598115
-0.        , -0.        ,  0.00971035, -0.00213454, -0.
-0.00048058, -0.05249263, -0.01707786, -0.03835413, -0.01573201
 0.04437185,  0.03696498,  0.03077517,  0.07246912,  0.04027807
 0.        ,  0.01029284,  0.06057017,  0.        ,  0.02023256
 0.02766283,  0.03409671, -0.00103521, -0.00583339, -0.02993058
-0.        ,  0.00524081, -0.02320387,  0.03528543,  0.10681356
 0.        ,  0.        ,  0.10463973, -0.        ,  0.02312323
```

Here we can see that the lasso regression achieves feature elimination which is very useful in this context as we are not doing any explicit PCA or RFE.

Marginal Parameters:

In addition to the above advantage, we also notice that,

- The difference between the train and test score for lasso is slightly better than ridge but marginally, 0.0005 vs 0.0007.
- The range of lasso coefficients is slightly better than ridge but marginally, 0.24 vs 0.22.

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?

The 5 most important factors in the current lasso model were:

[{

‘Neighborhood_StoneBr’: ’Neighborhood has Stone Brook’,

‘Neighborhood_Crawfor’: ’Neighborhood has Crawfor’,

‘Neighborhood_MeadowV’: ’Neighborhood has Meadow Village’,

‘GarageCars_3’: ’garage has capacity of 3 cars’,

'GrLivArea': ’Ground floor living area’

}]

After removal of these parameters and rebuilding the model, we get the following top 5 positively and top 5 negatively correlated variables.

	Coefficient	Column
0	-0.081991	MSZoning_RM
1	-0.059520	MSSubClass_Others
2	-0.058723	Neighborhood_Edwards
3	-0.047900	BsmtFinType1_Unf
4	-0.025450	Neighborhood_NWAmes

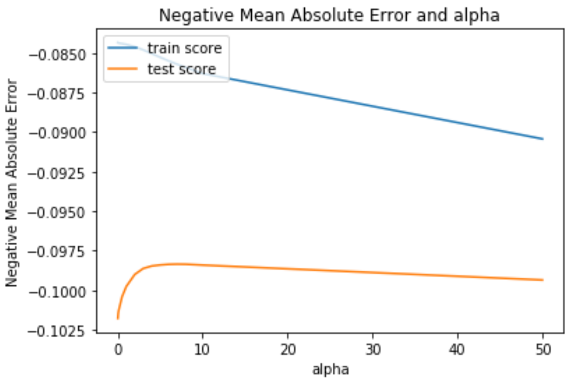
	Coefficient	Column
106	0.060921	HalfBath_1
107	0.065983	1stFlrSF
108	0.093876	BedroomAbvGr_4
109	0.114186	OverallQual
110	0.140938	FullBath_Others

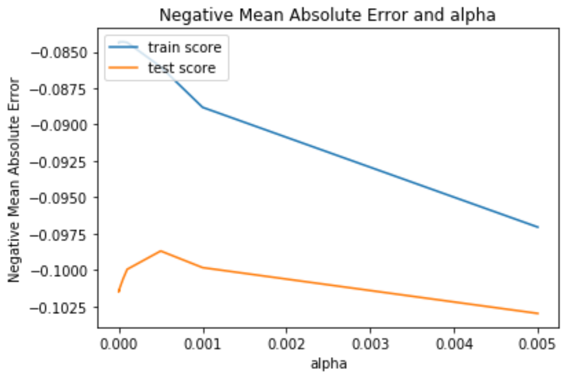
Question 4

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?

We can make sure that the model is more generalizable by putting more weightage to the regularization term. In order words, increasing the value of alpha. When that happens, the model will have a higher bias and lower variance leading to better prediction on unseen data. This achievement comes at the cost of a slight decrease in the training score. However, the test accuracy of the model is improved drastically thereby improving the overall model performance.

In our model, the hyper-parameter for ridge and lasso are 7 and 0.0005 respectively. In order to make the models more generalizable, we can increase these values. However, increasing beyond a certain point will increase the error incurred causing the model to underfit.





As we can see from the above graphs, we obtain the best value of alpha at the point where the test error is minimum. Let us call this point X.

- For alpha < X, the train error is very low (model is trying to overfit) and the test error is high.
- For alpha > X, the train error starts to increase (the model starts to generalize) but the test error increases as well.

So, for the model to be both robust and best generalizable, we keep the value of alpha at 7 and 0.0005 for ridge. If this is the case then the model accuracy will be maximum as the model does well for both seen and unseen data.