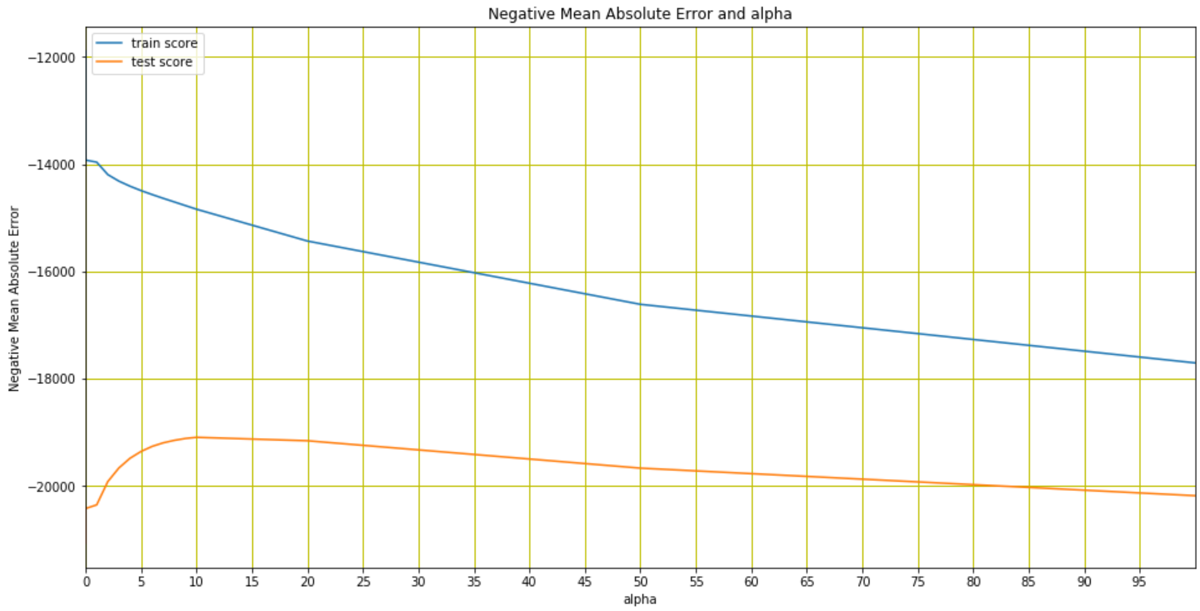
Question 1

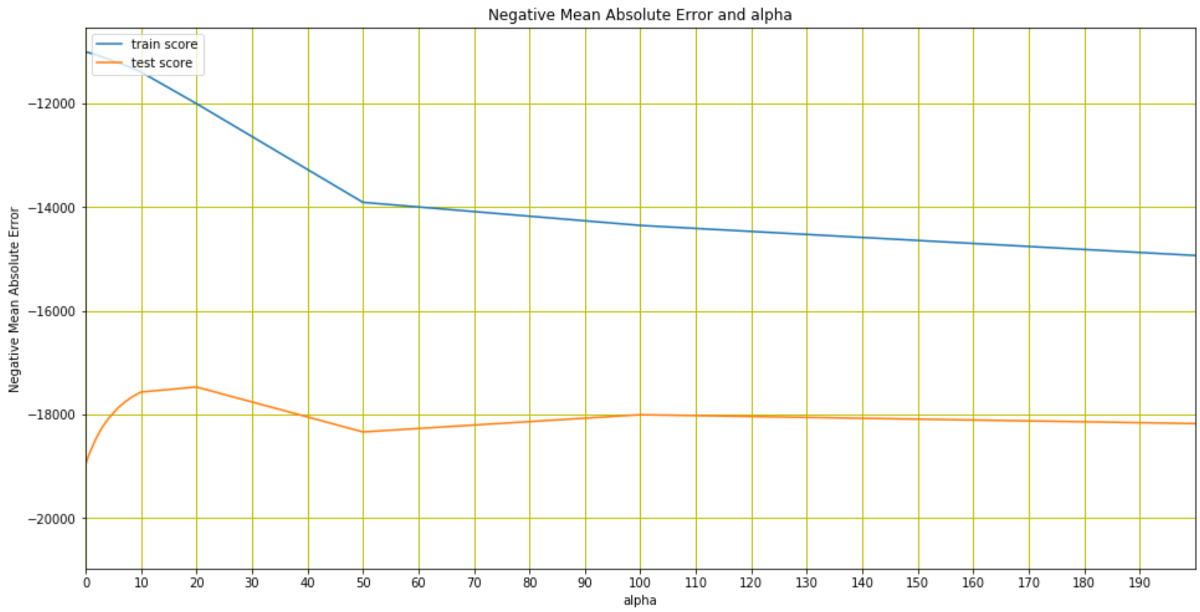
What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

Answer 1:

In ridge and Lasso regression, we use **GridSearchCV** to automatically perform 5-fold cross-validation with a range of different regularization parameters in order to find the optimal value of alpha. Optimal value of alpha in Ridge and Lasso is the value that gives lowest possible bias and variance.



Optimal value of alpha for ridge regression is 20



Optimal value of alpha for lasso regression is 100

Important predictors for alpha = 100

```
(-95688.766, 'Condition2_PosN'),
 (93983.954, 'RoofMatl_WdShngl'),
 (51003.943, 'Neighborhood_StoneBr'),
 (49604.526, 'OverallQual_9'),
 (45705.69, 'OverallQual_10'),
```

If we choose double the value of alpha, it will lead to higher regularization which means value of coefficients will decrease and the model will get simpler. Too simple models lead to underfitting as the model won't be able to learn the pattern properly. The variance will decrease but the bias will go up.

The most important predictor variables after taking double the value of alpha i.e 200

```
(65613.686, 'RoofMatl_WdShngl'),
 (48039.214, 'OverallQual_9'),
 (45509.275, 'Neighborhood_StoneBr'),
 (32325.584, 'Neighborhood_NridgHt'),
 (30364.388, 'OverallQual_10'),
```

As we can see by increasing the value of alpha, the weight of coefficients have reduced.

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 2:

Ridge Results

For alpha = 20:

```
Train Score:          0.8902909542799077
Test Score:           0.8758721048848186
```

LASSO Results

For alpha = 100

```
Train Score:          0.8962154960505715
Test Score:           0.8805324738207608
```

Out of the two, we will choose Lasso regression for following reasons:

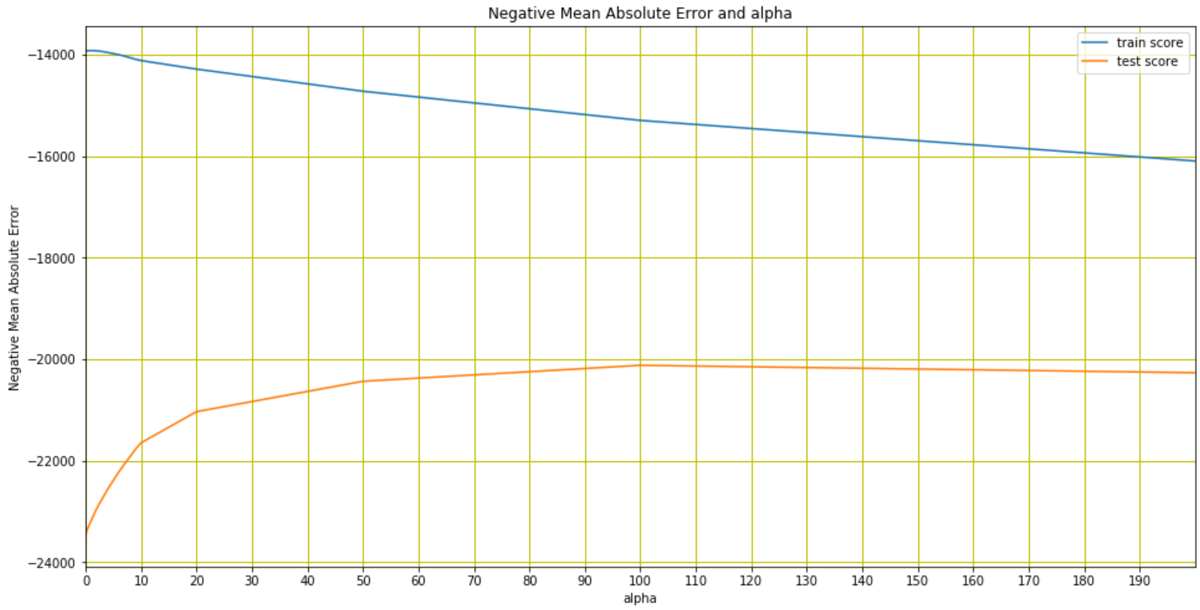
1. Lasso has also helped in the feature selection as it shrinks the coefficients of insignificant variables to zero.
2. Lesser features mean model is more robust.
3. The training and test scores produced by Lasso are slightly better than those produced by Ridge with an additional benefit of feature elimination.

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 3:

After removing the top 5 predictor variables, the Lasso regressin gives the following result:



Optimal alpha = 100

```
Train score:          0.8819444424065811
Test score:           0.8718802260258496
```

Most significant predictors are:

```
(-113196.821, 'Condition2_Norm'),
 (-35694.852, 'MasVnrType_None'),
 (-34767.405, 'HouseStyle_SLvl'),
 (-31509.02, 'LotShape_IR3'),
 (-30551.442, 'MasVnrType_BrkFace'),
```

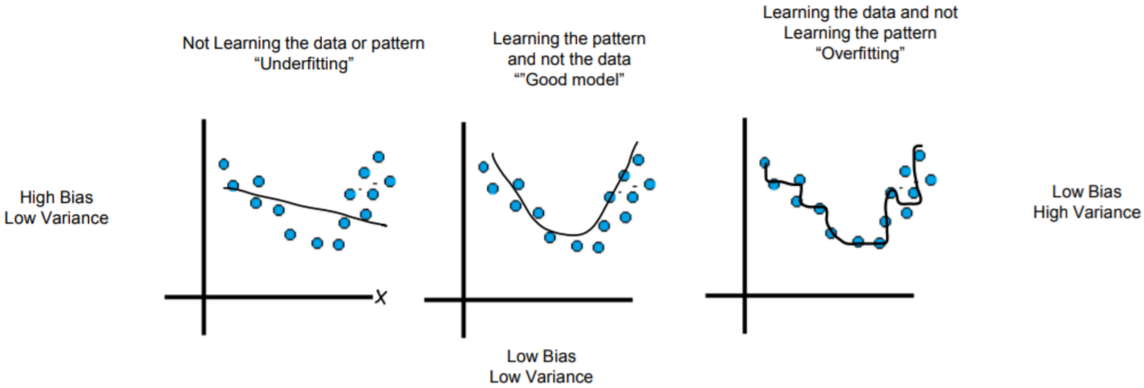
Question 4

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 4:

A regression model has to be as simple as possible, but no simpler. Simpler models are more generalizable and require fewer training samples for effective training than the more complex ones. However, Linear regression tries to minimize the error (e.g. MSE or OLS) without taking into account complexity of the model which often result in arbitrarily complex coefficients.

A problem with too simple model is that it doesn't learn the pattern well. As a result it is too naïve to produce meaningful results. This is also called underfitting. It's like a student who only learns the basic concepts and do not attempt to understand the application of those concepts. We can see in first graph how a simple linear equation misses the trend in the data. Underfitting is easier to detect as the model with perform poorly in training itself and adding more training data will address the issue.



On the other hand, complex models have the tendency to learn all the data and thus cause overfitting. Overfitting means instead of learning the pattern, the model fits through all the data points resulting in a wavy pattern, as seen in the third graph. Though it will give very good accuracy in the training but it will fail miserably when used on unseen test data. It is like a student who learns answers to all the questions and does not understand the concepts. Student will do very well when asked the same questions but unable to answer questions that (s)he didn't read earlier.

The solution to prevent overfitting is Regularization, which is to add a penalty to different parameters of the model to create an optimally complex model, i.e. a model which is as simple as possible while performing well on the training data.

Through regularization, we make a deliberate attempt to bring down the complexity and try to strike the delicate balance between keeping the model simple, yet not making it too naïve to be of any use. Hence the model will be less likely to fit the noise of the training data and improve the generalization of the model.

In regularized regression, the objective function has two parts - the **error term** and the **regularization term**.

The regularization term can be:

- Sum of squares of the coefficients or Ridge regression
- Sum of absolute value of the coefficients or Lasso regression

λ (also called the regularization rate) is the tuning parameter that decides how much we want to penalize the flexibility of our model.

1. λ = 0, it means no regularisation is performed.
2. λ is high, means more regularisation. Model will be simple, but we run the risk of underfitting the data. The model won't learn enough about the training data to make useful predictions.
3. λ is low, means less regularisation. Model will be more complex, and we run the risk of overfitting your data. The model will learn too much about the particularities of the training data, and won't be able to generalize to new data.