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 Best Estimator alpha value for Ridge regression is 0.3 and the Best Estimator alpha value for Lasso regression is 0.001.

Ridge:

Features with alpha = 0.3

	Features	Coefficient	Coef
248	PoolQC_Good	-1.2226	1.2226
104	Condition2_PosN	-0.5436	0.5436
14	GrLivArea	0.4803	0.4803
11	1stFlrSF	0.4602	0.4602
49	MSZoning_FV	0.3964	0.3964

Regression	R2_Score	MeanSquareError	RootMeanSquareError
Ridge	0.950260	0.008126	0.090144
Ridge	0.823815	0.026518	0.162842

Features with alpha DOUBLED = 0.06

	Features	Coefficient	Coef
248	PoolQC_Good	-0.8534	0.8534
104	Condition2_PosN	-0.4322	0.4322
14	GrLivArea	0.4187	0.4187
11	1stFlrSF	0.3907	0.3907
49	MSZoning_FV	0.3167	0.3167

DataSet	Regression	R2_Score	MeanSquareError	RootMeanSquareError
Train	Ridge	0.946262	0.008779	0.093697
Test	Ridge	0.849713	0.022620	0.150398

$$y = \sum_{i=1}^{n} (y_i - y_i')^2 + \lambda \sum_{j=1}^{P} \beta_j^2$$

Lasso:

Features with alpha = 0.001

	Features	Coefficient	Coef
14	GrLivArea	0.5570	0.5570
248	PoolQC_Good	-0.2901	0.2901
74	Neighborhood_Crawfor	0.1093	0.1093
35	MSSubClass_Subclass_160	-0.0934	0.0934
84	Neighborhood_NridgHt	0.0902	0.0902

DataSet	Regression	R2_Score	MeanSquareError	RootMeanSquareError
Train	Lasso	0.907256	0.015152	0.123092
Test	Lasso	0.879486	0.018138	0.134679

Features with alpha DOUBLED = 0.002

	Features	Coefficient	Coef
14	GrLivArea	0.1110	0.1110
2	OverallQual	0.0910	0.0910
24	GarageCars	0.0751	0.0751
183	BsmtExposure_GoodExposure	0.0740	0.0740
74	Neighborhood_Crawfor	0.0736	0.0736

DataSet	Regression	R2_Score	MeanSquareError	RootMeanSquareError
Train	Lasso	0.881111	0.019423	0.139366
Test	Lasso	0.870981	0.019419	0.139350

$$y = \sum_{i=1}^{n} (y_i - y_i')^2 + \lambda \sum_{j=1}^{P} |\beta_j|$$

With doubling of alpha value, the absolute value of coefficients has drastically reduced. This is because the Lasso and Ridge regression models add an error correction which would supress the coefficients of the features.

As the value of λ is increased/doubled, it will increase the penalty levied on the coefficients β_j .

For a Ridge regression, as the coefficients values tend to become 0 but not reach 0 value, the order and the features listed will not change.

However, for Lasso regression, as the coefficients' values at become 0 thereby a certain feature being eliminated. Due to this, when the value of λ is increased/doubled, the coefficients are suppressed for individual features and hence the features listed are re-shuffled based on the ranking.

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?

Looking at the following values:

```
In [87]: final_results = ridge_dataframe.append(lasso_dataframe)
         final_results.reset_index()
         final_results.set_index(['DataSet', 'Regression'])
         final_results
```

Out[87]:

	DataSet	Regression	R2_Score	MeanSquareError	RootMeanSquareError
0	Train	Ridge	0.950260	0.008126	0.090144
1	Test	Ridge	0.823815	0.026518	0.162842
0	Train	Lasso	0.907256	0.015152	0.123092
1	Test	Lasso	0.879486	0.018138	0.134679

We observe that the Ridge Regression has higher R2 score during the training dataset while its R2 score for test dataset is low (in comparison with the train set). While the Lasso regression has a slightly lesser R2 score on Training Dataset when compared with that of Ridge. Also, we observe there is significantly less difference between the R2 values of test and the train datasets of Lasso (in comparison with that of Ridge). We can infer that the model built with Ridge Regression is less efficient when compared with that of Lasso. Also, the MeanSquareError values are higher for Test Data in Ridge Regression while they are low in Lasso regression.

The current dataset consists of only some known data, and in general this is prone to be changing as per the trends in the market. Hence, eliminating the features from the list may not be an optimal choice given as the new trends may cause previously less important feature to gain importance now.

As Ridge regression would shrink the coefficients to 0 but never make them totally 0; whereas the Lasso regression would shrink the coefficients and make a few to 0; thus, making a feature selection. Hence, for the given dataset, the Ridge regression is preferred if we foresee any changes coming through the trends in the data. If not, then a Lasso regression can be picked up as it has more efficiency of prediction results.

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 initial set of 5 predictor variables are:

```
lasso_df.head()
```

	Features	Coefficient	Coef
14	GrLivArea	0.5570	0.5570
248	PoolQC_Good	-0.2901	0.2901
74	Neighborhood_Crawfor	0.1093	0.1093
35	MSSubClass_Subclass_160	-0.0934	0.0934
84	Neighborhood_NridgHt	0.0902	0.0902

Assuming the same alpha value being used here (0.01) for Lasso.

Now, after removing these features from the Training data, we get the new set as follows.

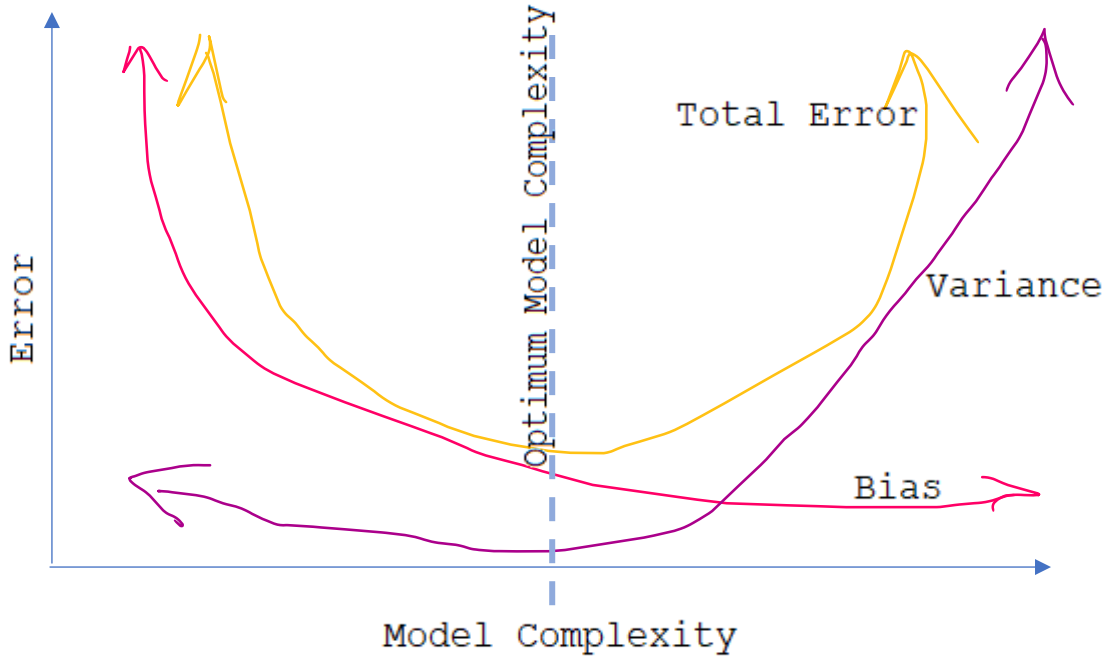
	Features	Coefficient	Coef
11	1stFlrSF	0.3004	0.3004
12	2ndFlrSF	0.0936	0.0936
129	Exterior1st_BrkFace	0.0897	0.0897
106	BldgType_Twnhs	-0.0876	0.0876
80	Neighborhood_NoRidge	0.0829	0.0829

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?

The robustness and generalizability of a model can be achieved by regularization. The regularization helps in penalizing the model coefficients to achieve optimum complexity of the model.

For us to achieve the optimum complexity, we must pick the model with a right balance between the Bias & Variance. A simpler model will not include all the features/patterns and hence has low variance. While a complex model understands and learns more on the train data in a way that it fails to perform well on the test data. Variance is evaluated on the training data. Bias helps to qualify the accuracy of the model on the test data.


Error
Optimum Model Complexity
Total Error
Variance
Bias
Model Complexity