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?

ANS: The optimal value of Alpha for ridge regression is 10 and that of lasso is 0.001; if we double the value for both ridge and lasso we end with 20 the alpha value for ridge and 0.1 for lasso, the top 5 important features of ridge and lasso are as follows:

For ridge:

In [149]:
```
important_features.sort_values(by = "Coefficient_ridge_20", ascending = False)[["Columns", "Coefficient_ridge_20"]]
```

Out[149]:

	Columns	Coefficient_ridge_20
13	GrLivArea	0.271700
76	Neighborhood_StoneBr	0.235332
70	Neighborhood_NridgHt	0.209115
10	TotalBsmtSF	0.164648
2	OverallQual	0.161652
...	...	...
27	YearBuilt_Age	-0.102842
33	MSSubClass_1-STORY_PUD_(Planned Unit Developme...	-0.106456
59	Neighborhood_CollgCr	-0.115181
68	Neighborhood_NWAmes	-0.115261
38	MSSubClass_2-STORY_PUD-1946_AND_NEWER	-0.143925

GrLivArea, Neighborhood_StoneBr, Neighborhood_NridgHt, TotalBsmntSF, OverallQual

For lasso:

```
In [178]: important_features.sort_values(by = "Coefficient_lasso_0.002", ascending = False)[["Columns", "Coefficient_lasso_0.002"
```

Out[178]:

	Columns	Coefficient_lasso_0.002
76	Neighborhood_StoneBr	0.476044
131	BsmtFinType1_No_Basement	0.372103
70	Neighborhood_NridgHt	0.347769
13	GrLivArea	0.302416
10	TotalBsmtSF	0.182061
...	...	...
120	MasVnrType_BrkFace	-0.090082
27	YearBuilt_Age	-0.111662
68	Neighborhood_NWAmes	-0.118537
33	MSSubClass_1-STORY_PUD_(Planned Unit Developme...	-0.186222
38	MSSubClass_2-STORY_PUD-1946_AND_NEWER	-0.234206

151 rows × 2 columns

Neighborhood_StoneBr, BsmtFinType1_No_Basement, Neighborhood_NridgHt, GrLivArea, MSSubClass_2-STORY_PUD-1946_AND_NEWER

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?

ANS: I will go with Lasso regression since the accuracy for both ridge and lasso are the same, the reason I went with lasso specifically is because of its feature selection which it does automatically and by that I mean is it makes the coefficient of certain betas 0, so that we know that that particular feature isn't that important and we can safely ignore them; this particular feature isn't available with Ridge; although we can rule the coefficient out which are extremely close to Zero, though it may not be the best practice there is, since manually removing the variables may affect other coefficients.

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?

ANS: The five most important predictor variables for alpha = 0.001 before removing them from the model are Neighborhood_StoneBr, BsmtFinType1_No_Basement, Neighborhood_NridgHt, GrLivArea, MSSubClass_2-STORY_PUD-1946_AND_NEWER

After dropping them we found the following 5 most important predictor variables
For alpha = 0.001 we get the following variables

```
In [172]: new_features.sort_values(by = "lasso_Coefficients", ascending = False)
```

Out[172]:

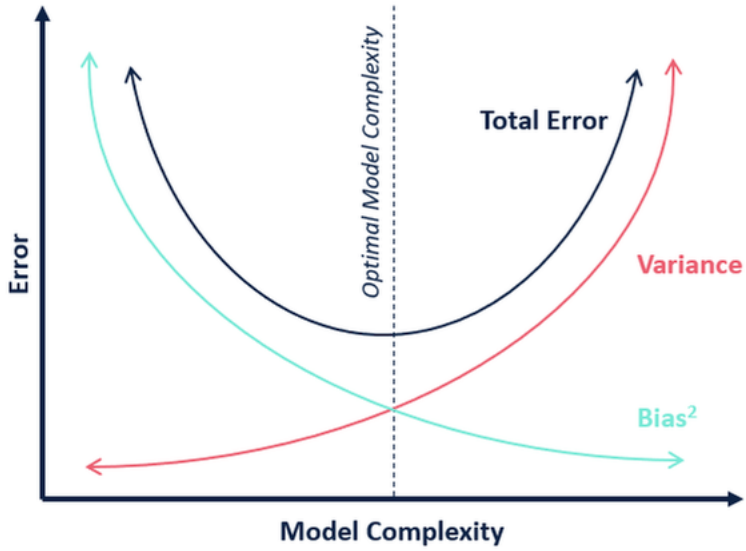
	Columns	lasso_Coefficients
121	Foundation_Slab	0.496390
10	TotalBsmtSF	0.292006
12	2ndFlrSF	0.207857
2	OverallQual	0.194456
134	GarageType_No_Garage	0.185285
...	...	...
72	Neighborhood_Somerst	-0.181897
63	Neighborhood_Mitchel	-0.240633
66	Neighborhood_NWAmes	-0.243529
57	Neighborhood_CollgCr	-0.279136
60	Neighborhood_Gilbert	-0.289362

Foundation_Slab, TotalBsmtSF, Neighborhood_Gilbert, Neighborhood_CollgCr, Neighborhood_NWAmes

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?

ANS: There are countless different combinations of data transformation and hyper parameter tuning to achieve robustness and all in all a very generalisable model.For a complex dataset we can perform data cleaning and imputation but more importantly we need to understand the data, impute all the missing value and proceed ahead with model building, for a complex model it will have high variance and low bias which will cause our model to overfit thus rendering it useless, we can introduce data transforamtions such as log, inverse etc on our predictor and response variables use polynomial regression of degree n or use a non linear model altogether, for a model which underfits we can use different sampling techniques such as SMOTE and KNN-SMOTE to increase our data so that it can learn the trend, in the end there’s there is always a tradeoff between bias and variance



From right to left we can see the tradeoff for bias and variance as the model complexity decreases, there is an increase in bias and to the far left we see an underfitted model, somewhere between we have the optimal Model complexity which may have our desired goal.

Accuracy is measured for both training and test, the accuracy of the training data will be extremely high in a complex model but will perform poorly on the test set and in case of an underfitted model the training error will be high as it will fail to learn any pattern thus will fail to achieve our desired performance, so striking the balance between Bias and variance is important so that our model can achieve our desired accuracy