# Assignment – 2 – Advanced Regression

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*** –

*The optimal value of alpha for ridge and lass expression:*

| *Type* | *Alpha* |
|---|---|
| ***Ridge*** | *10* |
| ***Lasso*** | *0.001* |

*Changes after doubling the alpha*

| Type | Alpha | R2 Score Train | R2 Score Test | Most Important Vars |
|---|---|---|---|---|
| **Ridge** | 10 | 0.9338 | 0.9084 | Neighborhood_Crawfor<br>GrLivArea<br>OverallQual<br>SaleCondition_Normal<br>Functional_Typ |
| | 20 | 0.9296 | 0.9090 | GrLivArea<br>OverallQual<br>Neighborhood_Crawfor<br>SaleCondition_Normal<br>Functional_Typ |
| **Lasso** | 0.001 | 0.9222 | 0.9064 | GrLivArea<br>SaleCondition_Partial<br>Neighborhood_Crawfor<br>OverallQual<br>SaleCondition_Normal |
| | 0.002 | 0.9138 | 0.9036 | GrLivArea<br>SaleCondition_Partial<br>Neighborhood_Crawfor<br>OverallQual<br>SaleCondition_Normal |

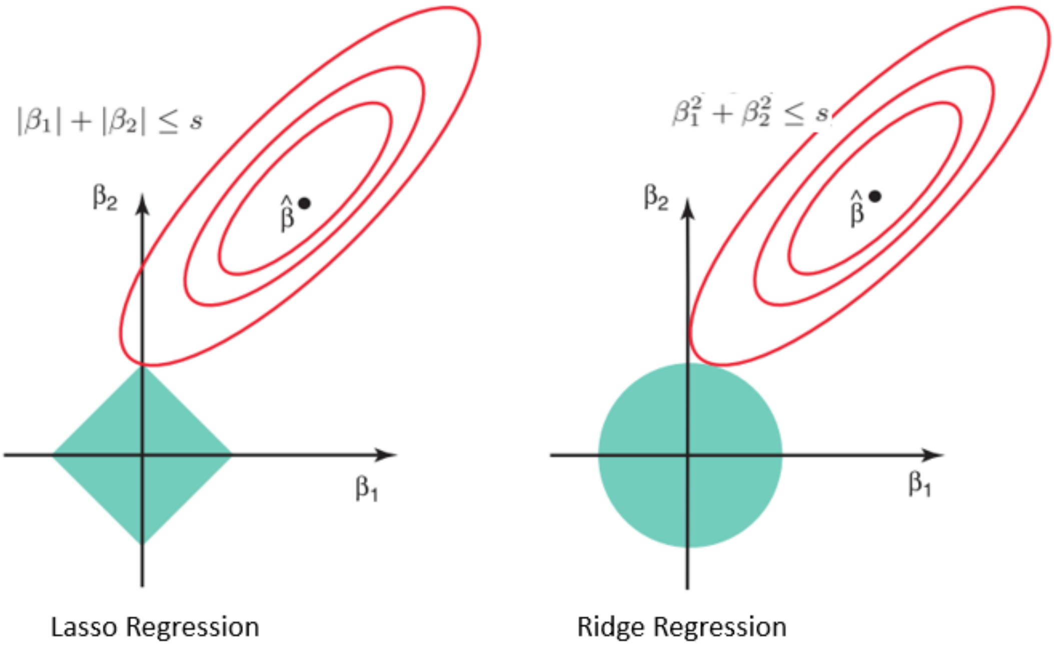
2. You have determined the optimal value of lambda for ridge and lasso regression during the assignment. Now, which one will you choose to apply and why?

**Answer** –

*These are two commonly used regularised regression methods - Ridge regression and Lasso regression. Both these methods are used to make the regression model simpler while balancing the 'bias-variance' trade-off.*
*Both Ridge and Lasso regularize the coefficients by reducing them in value, essentially causing shrinkage of the coefficients. Ridge and Lasso perform different measures of shrinkage which depends on the value of hyperparameter, λ. In the process of shrinkage, Lasso shrinks some of the variable coefficients to 0, thus performing variable selection.*

***It is better to use Lasso than Ridge when comparing complexity because Lasso will select the best features from the existing variables while Ridge will keep all the variables while reducing the coefficient of variables. And also Lasso regression model is more simpler.***

***Thus Lasso was chosen over Ridge Regression.***



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*** –

***Original 5 most important predictor variables:***

1. *GrLivArea*
2. *SaleCondition_Partial*
3. *Neighborhood_Crawfor*
4. *OverallQual*
5. *SaleCondition_Normal*

*After removing these variables and creating another model, we got following five most important predictors :*

1. *MSSubClass*
2. *LowQualFinSF*
3. *2ndFlrSF*
4. *FireplaceQu_Fa*
5. *MasVnrArea*

```
#Removing top 5 features of Lasso regression and building the model again
lasso_vars = ['GrLivArea','SaleCondition_Partial','Neighborhood_Crawfor','OverallQual','SaleCondition_Normal']

X_train_lasso = X_train.drop(lasso_vars,axis=1)
X_test_lasso = X_test.drop(lasso_vars,axis=1)
```
✓ 0.4s

```
lasso = Lasso(alpha=0.001)
lasso.fit(X_train_lasso,y_train)

y_train_pred_lasso = lasso.predict(X_train_lasso)
y_test_pred_lasso = lasso.predict(X_test_lasso)

print(r2_score(y_true=y_train,y_pred=y_train_pred_lasso))
print(r2_score(y_true=y_test,y_pred=y_test_pred_lasso))
```
✓ 0.1s

```
0.9112968085439112
0.8931441016549596
```

```
model_param = list(lasso.coef_)
model_param.insert(0,lasso.intercept_)
cols = X_train_lasso.columns
cols.insert(0,'const')
lasso_coef = pd.DataFrame(list(zip(cols,model_param)))
lasso_coef.columns = ['Feature','Coef']
lasso_coef.sort_values(by='Coef',ascending=False).head(10)
```
✓ 0.3s

| | Feature | Coef |
|---|---|---|
| 0 | MSSubClass | 11.840541 |
| 13 | FullBath | 0.130009 |
| 4 | MasVnrArea | 0.090594 |
| 51 | Neighborhood_MeadowV | 0.066734 |
| 5 | BsmtFinSF1 | 0.056192 |

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*** *–*

*As Per, Occam's Razor— given two models that show similar 'performance' in the finite training or test data, we should pick the one that makes fewer on the test data due to following reasons:-*

- *Simpler models are usually more 'generic' and are more widely applicable*
- *Simpler models require fewer training samples for effective training than the more complex ones and hence are easier to train.*
- *Simpler models are more robust.*

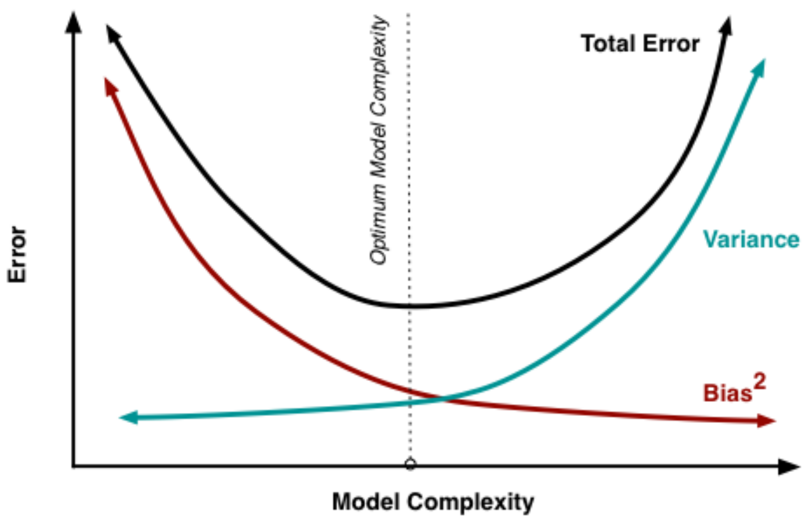
  - *Complex models tend to change wildly with changes in the training data set*
  - *Simple models have low variance, high bias and complex models have low bias, high variance*
  - *Simpler models make more errors in the training set. Complex models lead to overfitting — they work very well for the training samples, fail miserably when applied to other test samples*

*Therefore, to make the model more robust and generalizable, make it simple but not simpler, which will be of no use.*

*Regularization can be used to make the model simpler. Regularization helps to strike the delicate balance between keeping the model simple and not making it too naive to be of any use. For regression, regularization involves adding a regularization term to the cost that adds up the absolute values or the squares of the parameters of the model.*

*Also, Making a model simple leads to Bias-Variance Trade-off:*

- *A complex model will need to change for every little change in the dataset and hence is very unstable and extremely sensitive to any changes in the training data.*
- *A simpler model that abstracts out some pattern followed by the data points given is unlikely to change wildly even if more points are added or removed.*



*Bias quantifies how accurate is the model likely to be on test data. A complex model can do an accurate job prediction provided there is enough training data. Models that are too naïve, for e.g., one that gives same answer to all test inputs and makes no discrimination whatsoever has a very large bias as its expected error across all test inputs are very high. Variance refers to the degree of changes in the model itself with respect to changes in the training data. Thus accuracy of the model can be maintained by keeping the balance between Bias and Variance as it minimizes the total error as shown in the figure above.*