Subjective Question Assignment

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: As per the calculations done in the iPython Notebook:

Best alpha value for Lasso : {'alpha': 0.001}

Best alpha value for Ridge : {'alpha': 0.9}

For LASSO alpha 0.001

```
[('MSSubClass', -1.046),
 ('LotArea', 0.226),
 ('LotShape', 0.183),
 ('LandSlope', 0.171),
 ('OverallQual', 0.04),
 ('OverallCond', -0.0),
 ('MasVnrArea', 0.078),
 ('ExterQual', 0.007),
 ('ExterCond', 0.0),
 ('BsmtQual', -0.028),
 ('BsmtCond', 0.36),
 ('BsmtExposure', 0.15),
 ('BsmtFinType1', 0.016),
 ('BsmtFinSF1', 0.115),
 ('BsmtFinType2', -0.0),
 ('BsmtFinSF2', -0.054),
 ('BsmtUnfSF', 0.204),
 ('TotalBsmtSF', -0.067),
 ('HeatingQC', -0.029),
 ('CentralAir', -0.137),
 ('1stFlrSF', -0.264),
 ('2ndFlrSF', -0.122),
```

```
('LowQualFinSF', -0.139),
('GrLivArea', -0.213),
('BsmtFullBath', -0.261),
('BsmtHalfBath', -0.225),
('FullBath', 0.394),
('HalfBath', -0.244),
('BedroomAbvGr', -0.224),
('KitchenAbvGr', -0.175),
('KitchenQual', -0.208),
('TotRmsAbvGrd', 0.006),
('Fireplaces', 0.148),
('GarageFinish', -0.045),
('GarageCars', -0.0),
('GarageArea', -0.104),
('GarageQual', -2.03),
('GarageCond', -0.366),
('WoodDeckSF', -0.292),
('OpenPorchSF', -0.0),
('EnclosedPorch', -0.206),
('3SsnPorch', 0.0),
('ScreenPorch', 0.0),
('PoolArea', -0.0),
('MiscVal', 1.556),
('YearBuilt_Old', 0.155),
('YearRemodAdd_Old', -0.0),
('GarageYrBlt_Old', 0.0),
('YrSold_Old', -0.081),
('MSZoning_FV', -0.0),
('MSZoning_RH', -0.0),
('MSZoning_RL', -0.0),
('MSZoning_RM', 0.0),
('LandContour_HLS', 0.036),
('LandContour_Low', 0.04),
('LandContour_Lvl', 0.028),
('LotConfig_CulDSac', -0.0),
('LotConfig_FR2', -0.0),
('LotConfig_FR3', -0.0),
('LotConfig_Inside', 0.163),
('Neighborhood_Blueste', -0.009),
('Neighborhood_BrDale', 0.0),
('Neighborhood_BrkSide', 0.03),
('Neighborhood_ClearCr', -0.0),
('Neighborhood_CollgCr', 0.0),
('Neighborhood_Crawfor', -0.019),
```

```
 ('Neighborhood_Edwards', -0.0),
 ('Neighborhood_Gilbert', 0.343),
 ('Neighborhood_IDOTRR', 0.0),
 ('Neighborhood_MeadowV', 0.107),
 ('Neighborhood_Mitchel', 0.0)]
```

Now lets try to improve our model with the optimal value of alpha using GridSearchCV

```
folds = KFold(n_splits=10,shuffle=True,random_state=42)

hyper_param = {'alpha':[0.001, 0.01, 0.1,1.0, 5.0, 10.0,20.0]}

model = Lasso()

model_cv = GridSearchCV(estimator = model,
                        param_grid=hyper_param,
                        scoring='r2',
                        cv=folds,
                        verbose=1,
                        return
```

For Ridge alpha 0.9

	Feaure	Coef
44	MiscVal	1.552991
67	Neighborhood_Gilbert	0.435588
26	FullBath	0.342914
42	ScreenPorch	0.330342
68	Neighborhood_IDOTRR	0.304116
16	BsmtUnfSF	0.211166
1	LotArea	0.206851

	Feaure	Coef
10	BsmtCond	0.198022
41	3SsnPorch	0.194943
13	BsmtFinSF1	0.180495

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:

Looking at the Variable coefficients, I will prefer to use Lasso Regression Results because it makes multiple coefficients to Zero. And hence it automatically drops the non-important variables and let us know the important variables along with its relation which will help us in deciding the sales price of the house more effectively.

Question 3

After building the model, you realized 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:

If we would have not got the top 5 most important predicted variable we would have got below most important predicted variable:

#MiscVal : $Value of miscellaneous feature

#BsmtHalfBath : Basement half bathrooms

#LowQualFinSF : Low quality finished square feet (all floors)

#BsmtFullBath : Basement full bathrooms

#HalfBath : Half baths above grade

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: To make sure my model is robust and generalizable:

1) I will test it in on both Train and Test dataset and make sure to choose that Hyperparameter for which the accuracy (R square) on both Test and Train is close.
2) Make sure no overfitting is happening. If overfitting is there then we try to add some bias to it (using regularization techniques)
3) Also, I will try to use lesser number of independent variables to predict the dependent variable. It can be achieved by VIF or applying Regularization (preferably Lasso).

Implication of doing above is that: Accuracy of the model decreases which is not good to have.

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 widely used and are usually more 'generic'.
- 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

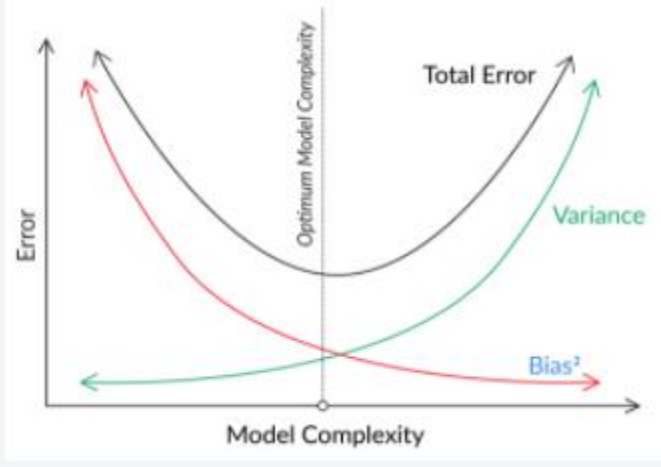
Therefore to make the model more robust and generalizable, make the model simple but not simpler which will not be of any use.
Regularization can be used to make the model simpler. Regularization helps to strike the delicate balance between keeping the model simple and not making it too naive to be of 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 below graph


Optimum Model Complexity
Total Error
Variance
Bias²
Error
Model Complexity