Subjective Questions:

Question 1

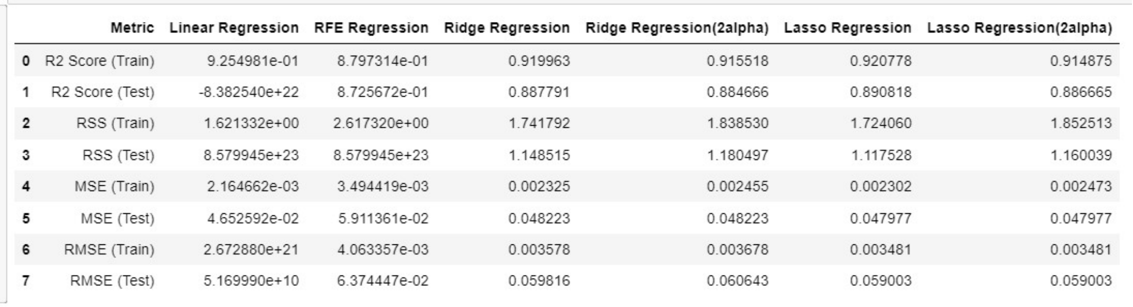
What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented

Answer:

Optimal value for alpha for ridge : 2.0

Optimal value for alpha for Lasso: 0.0001

With current alpha for ridge(2.0) ,lasso (0.0001), with alpha double the times for ridge and lasso, please find the below metrics (column no 5 and 7)



	Metric	Linear Regression	RFE Regression	Ridge Regression	Ridge Regression(2alpha)	Lasso Regression	Lasso Regression(2alpha)
0	R2 Score (Train)	9.254981e-01	8.797314e-01	0.919963	0.915518	0.920778	0.914875
1	R2 Score (Test)	-8.382540e+22	8.725672e-01	0.887791	0.884666	0.890818	0.886665
2	RSS (Train)	1.621332e+00	2.617320e+00	1.741792	1.838530	1.724060	1.852513
3	RSS (Test)	8.579945e+23	8.579945e+23	1.148515	1.180497	1.117528	1.160039
4	MSE (Train)	2.164662e-03	3.494419e-03	0.002325	0.002455	0.002302	0.002473
5	MSE (Test)	4.652592e-02	5.911361e-02	0.048223	0.048223	0.047977	0.047977
6	RMSE (Train)	2.672880e+21	4.063357e-03	0.003578	0.003678	0.003481	0.003481
7	RMSE (Test)	5.169990e+10	6.374447e-02	0.059816	0.060643	0.059003	0.059003

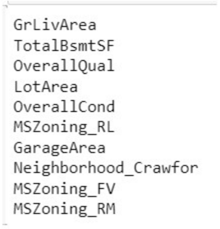
With double the value of alpha, for both ridge and lasso , R2 score starts decreasing, RSS starts increasing , MSE starts increasing, RMSE starts increasing (Increase in error rate)

That is error rate started to increase gradually. For a better model RMSE should be as low as possible. Hence with double the value of alpha error rate started increasing .

Higher the value of alpha , more the regularization and model complexity would be moving towards simple model which will result in underfitting which cause more bias(error) on training data

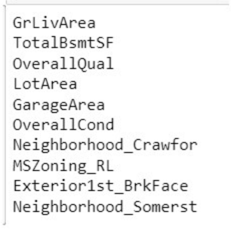
Below is the top features with optimal alpha(col 1, 3) and double alpha(col 2, 4) using ridge and lasso

Ridge Top Features



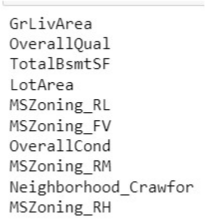
```
GrLivArea
TotalBsmtSF
OverallQual
LotArea
OverallCond
MSZoning_RL
GarageArea
Neighborhood_Crawfor
MSZoning_FV
MSZoning_RM
```

Ridge Top Fea-Double alpha



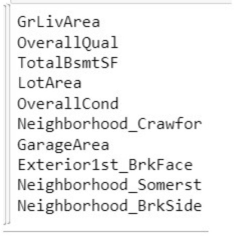
```
GrLivArea
TotalBsmtSF
OverallQual
LotArea
GarageArea
OverallCond
Neighborhood_Crawfor
MSZoning_RL
Exterior1st_BrkFace
Neighborhood_Somerst
```

Lasso Top Features



```
GrLivArea
OverallQual
TotalBsmtSF
LotArea
MSZoning_RL
MSZoning_FV
OverallCond
MSZoning_RM
Neighborhood_Crawfor
MSZoning_RH
```

Lasso Top Fea-Double alpha



```
GrLivArea
OverallQual
TotalBsmtSF
LotArea
OverallCond
Neighborhood_Crawfor
GarageArea
Exterior1st_BrkFace
Neighborhood_Somerst
Neighborhood_BrkSide
```

Question 2

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

Solution :

As we know the Lasso regression adds a penalty to regression cost function as magnitude of cost function as below

$$\circ\ Lasso\ Regression\ Cost = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

With optimum value of chosen alpha , lasso helps in making some of s model co-efficients to become exactly zero which will help in feature selection for model which has high number of features

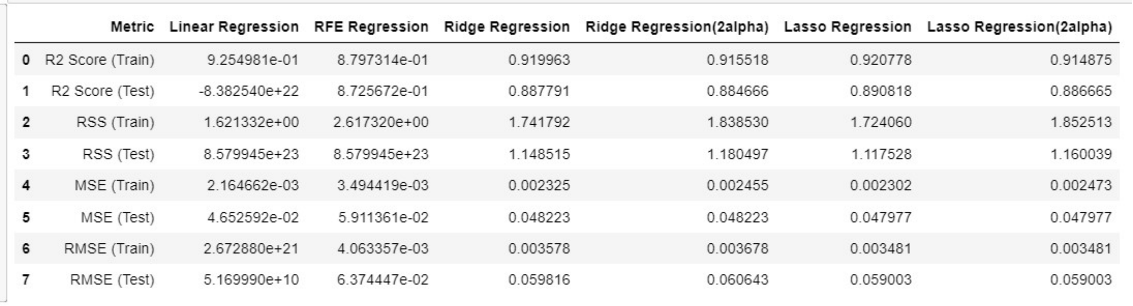
Due to this feature selection that happens in Lasso it is easier to interpret models generated by Lasso as compared to Ridge where as Ridge regression retains all variables that are present in data.

Now when the number of variables very large (happen to be the case of house prediction model) and data may have unrelated or noisy variable we may not want to keep those variables in model

Ridge regression does not do feature selection even if some of predictors are noisy. Although the accuracy of model would not be affected but model interpretation more challenging if we choose ridge

Hence for our problem statement as the number of features were high we can conclude by going with Lasso

Short summary: we will choose Lasso as it is will do reduce the co-efficient to zero which helps us feature selection where ridge will try to move the co-efficients towards zero (not exact zero) hence would be choosing lasso



	Metric	Linear Regression	RFE Regression	Ridge Regression	Ridge Regression(2alpha)	Lasso Regression	Lasso Regression(2alpha)
0	R2 Score (Train)	9.254981e-01	8.797314e-01	0.919963	0.915518	0.920778	0.914875
1	R2 Score (Test)	-8.382540e+22	8.725672e-01	0.887791	0.884666	0.890818	0.886665
2	RSS (Train)	1.621332e+00	2.617320e+00	1.741792	1.838530	1.724060	1.852513
3	RSS (Test)	8.579945e+23	8.579945e+23	1.148515	1.180497	1.117528	1.160039
4	MSE (Train)	2.164662e-03	3.494419e-03	0.002325	0.002455	0.002302	0.002473
5	MSE (Test)	4.652592e-02	5.911361e-02	0.048223	0.048223	0.047977	0.047977
6	RMSE (Train)	2.672880e+21	4.063357e-03	0.003578	0.003678	0.003481	0.003481
7	RMSE (Test)	5.169990e+10	6.374447e-02	0.059816	0.060643	0.059003	0.059003

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?

After removing the five most important predictor variables in lasso model, below were the next important features predicted by Lasso (refer the python notebook subjective question3 for model building ref)

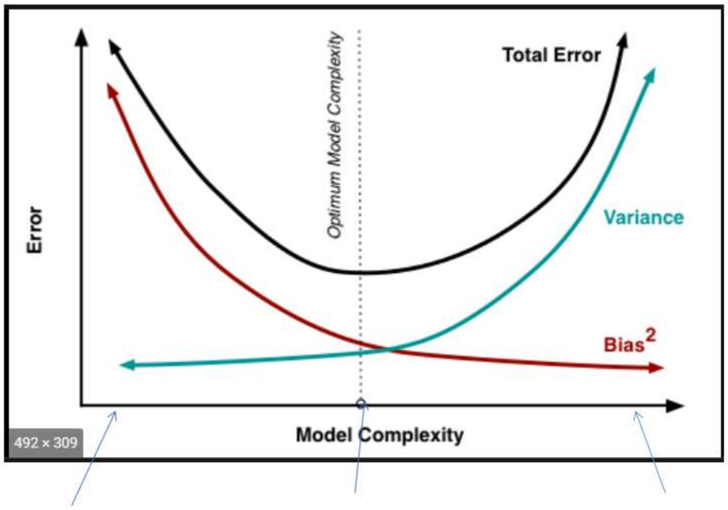
TotRmsAbvGrd
GarageArea
2ndFlrSF
Neighborhood_StoneBr
Neighborhood_Crawfor
KitchenQual

Exterior1st_BrkFace
FullBath
BsmtFinSF1
OverallCond

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?

As per Occam Razor principle model should be as simple (generalizable) as possible but robust



High Lambda/
underfitting

Optimum
Lambda

Low Lambda/
overfitting

Note: Alpha / Lambda represents one and same

Simpler model would be less complex and learns more generalized pattern so it ends up with high bias and low variance . High bias leads to more error rate . Low variance leads to ability of model to predict well on unseen data (Underfitting)
Complex model would be more complex and learns the data more well so it leads to resulting model with low bias and high variance . low bias leads to reduced error rate . High variance leads to inability of model to perform well on unseen data (Overfitting)

Hence if we see the above graph on both the ends
that is leftmost side (simple model) and right most side(complex model)-left most side - has more total error . Hence accuracy will be lower on both the sides

If we need to describe more clearly , cost function of linear regression aims at reduced RSS (residual sum of squares) which would end up in complex model (overfitting) which has its disadvantage of high variance which significantly affects the model performance on unseen data

Hence regularization came into effect which adds a penalty to cost function wrt sum magnitude of co-efficient (Lasso) or sum of squares of magnitude of co-efficient (Ridge)

If we have a model with optimized lambda or Alpha (using regularization) that is if we choose an optimum value of alpha at the intercept point of bias and variance (where the bias and variance is low) and use it as penalty for regression cost function , then resulting model should be capable enough to perform well on unseen data (more generalized) and capable enough to produce less error (low bias) which intersects the middle , we will end up in optimum model complexity (ref the fig optimum lambda in the middle) which has significantly better accuracy (As you can see at the centre we have Total error is low)

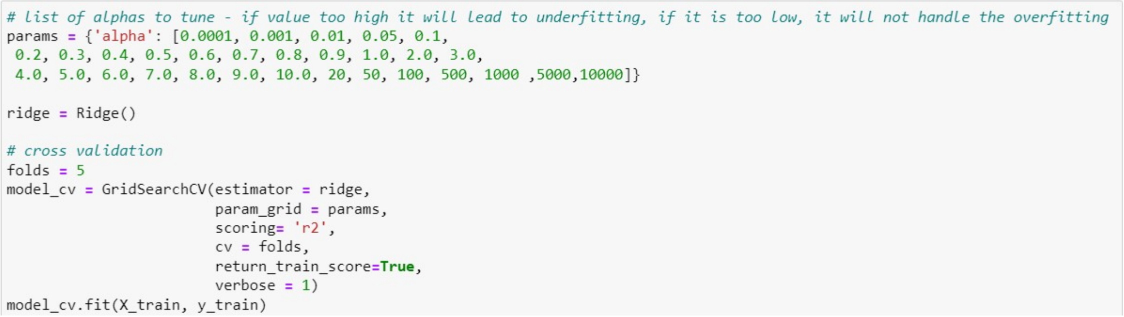
Regularization helps to choose the best value of alpha/lambda by running k folds iteratively (by testing iteratively using training and validation data for various values of alpha/lambda) and gives us various values of alpha/lamda using scoring technique for various values of lamda

Hence by choosing an optimum value of alpha we are making a model which is wise enough to learn the underlying patterns and able to perform better on unseen data

If Lambda is Zero : There is no regularization we will end up in model which would overfit
If Lambda is optimum: Fitted model would be close to actual data
If Lambda : Higher: Model starts underfitting

We can make sure that the model is robust and generalizable by choosing an optimum value of alpha/lambda

Robustness: By measuring the R2 score and RMSE for the model on test data(unseen data) . R2 score should be higher and RMSE should be lower. To achieve this we can pass scoring param as R2 for grid search cross validation with folds to run iteratively on training and validation data

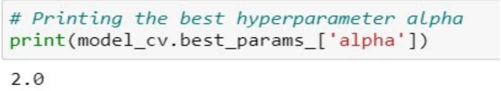


```
# list of alphas to tune - if value too high it will lead to underfitting, if it is too low, it will not handle the overfitting
params = {'alpha': [0.0001, 0.001, 0.01, 0.05, 0.1,
 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 2.0, 3.0,
 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 20, 50, 100, 500, 1000 ,5000,10000]}

ridge = Ridge()

# cross validation
folds = 5
model_cv = GridSearchCV(estimator = ridge,
                        param_grid = params,
                        scoring= 'r2',
                        cv = folds,
                        return_train_score=True,
                        verbose = 1)
model_cv.fit(X_train, y_train)
```

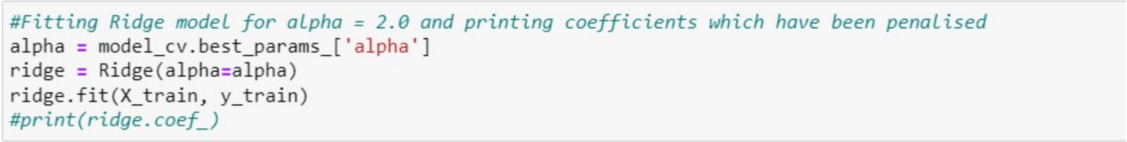
Optimum alpha:



```
# Printing the best hyperparameter alpha
print(model_cv.best_params_['alpha'])
```

```
2.0
```

Model is built using best alpha for the given scoring technique



```
#Fitting Ridge model for alpha = 2.0 and printing coefficients which have been penalised
alpha = model_cv.best_params_['alpha']
ridge = Ridge(alpha=alpha)
ridge.fit(X_train, y_train)
#print(ridge.coef_)
```

In this way with by choosing optimum alpha/lamda which ends up in less error

Generalization : Is Measured by ability of model to learn underlying patterns without overfitting so it can perform well on unseen data . Simpler models were always more generalizable .

We are making sure by choosing an optimum value of alpha/lambda helps us avoiding overfitting and helps to make the model more generalizable as regularization introduces a penalty of sum of magnitues of co-efficients (lasso) or sum of squares of magnitude of co-efficients(ridge)

$$Lasso\ Regression\ Cost = \sum_{i=1}^{n}(y_i \ - \ \hat{y}_i)^2 + \ \lambda \sum_{j=1}^{p} |\beta_j|$$

$$Ridge\ Regression\ Cost = \sum_{i=1}^{n}(y_i \ - \ \hat{y}_i)^2 + \ \lambda \sum_{j=1}^{p} \beta_j^2$$

If the chosen lamda/alpha is very low it will lead to underfitting which will . As we are choosing optimum value of lambda/alpha we are making sure model is more generalizable

Accuracy : Accuracy of model is measured by closeness of the predicted values to the actual values using 3 metrics like R2 score, RSS -residual sum of error , MSE mean square error and RMSE – root mean square error

Higher the R2 score more good is model ,
Smaller the RSS closer is model to fit the data,
MSE – smaller the value better is model
RMSE – smaller the value better is model