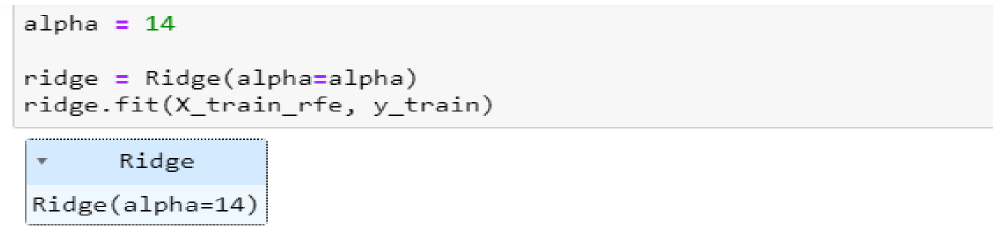
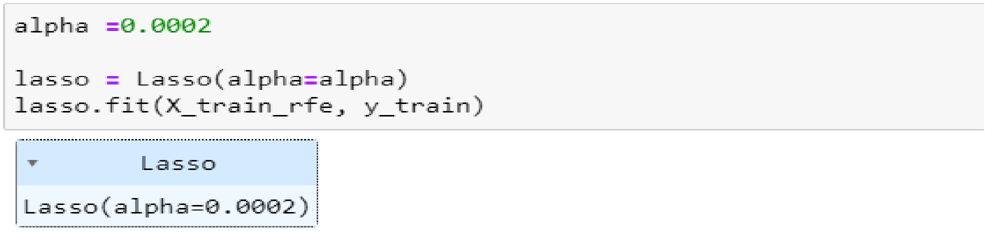
1. What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

- The optimal alpha value for ridge is 7. So double its value is 14
- The optimal alpha value for lasso is .0001. So double it's value is .0002



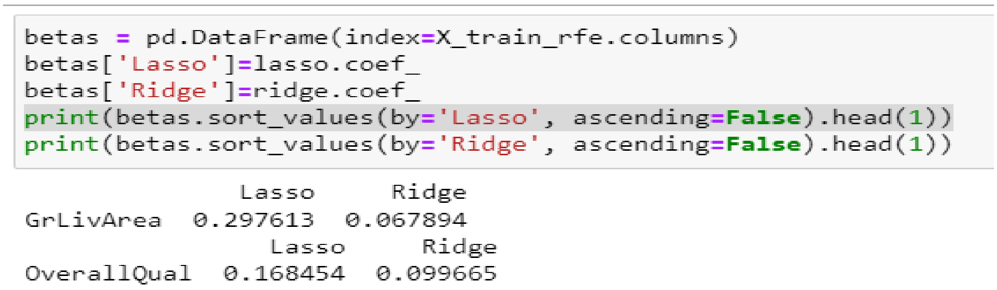
```
alpha = 14

ridge = Ridge(alpha=alpha)
ridge.fit(X_train_rfe, y_train)
```

```
▾     Ridge
Ridge(alpha=14)
```



```
alpha =0.0002

lasso = Lasso(alpha=alpha)
lasso.fit(X_train_rfe, y_train)
```

```
▾       Lasso
Lasso(alpha=0.0002)
```



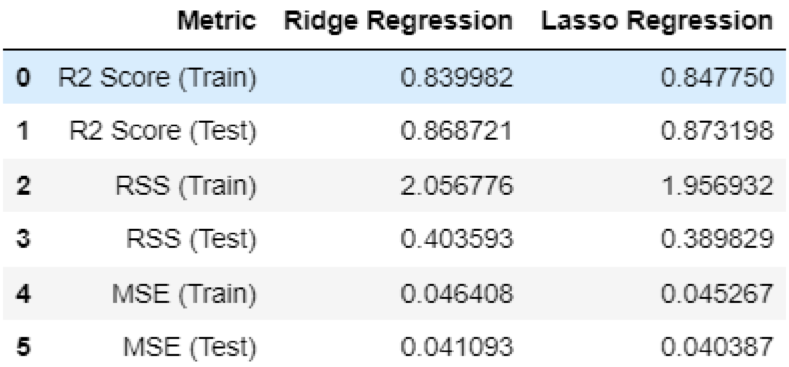
```
betas = pd.DataFrame(index=X_train_rfe.columns)
betas['Lasso']=lasso.coef_
betas['Ridge']=ridge.coef_
print(betas.sort_values(by='Lasso', ascending=False).head(1))
print(betas.sort_values(by='Ridge', ascending=False).head(1))
```

```
              Lasso     Ridge
GrLivArea  0.297613  0.067894
                Lasso     Ridge
OverallQual  0.168454  0.099665
```

After doubling alpha metrics

- For **lasso model**, alpha=0.0002, the **best predictor is GrLivArea**. This predictor describes the square feet of the property
- For **ridge model**, alpha=14, the **best predictor is OverallQual**. This predictor overall quality of the property

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?

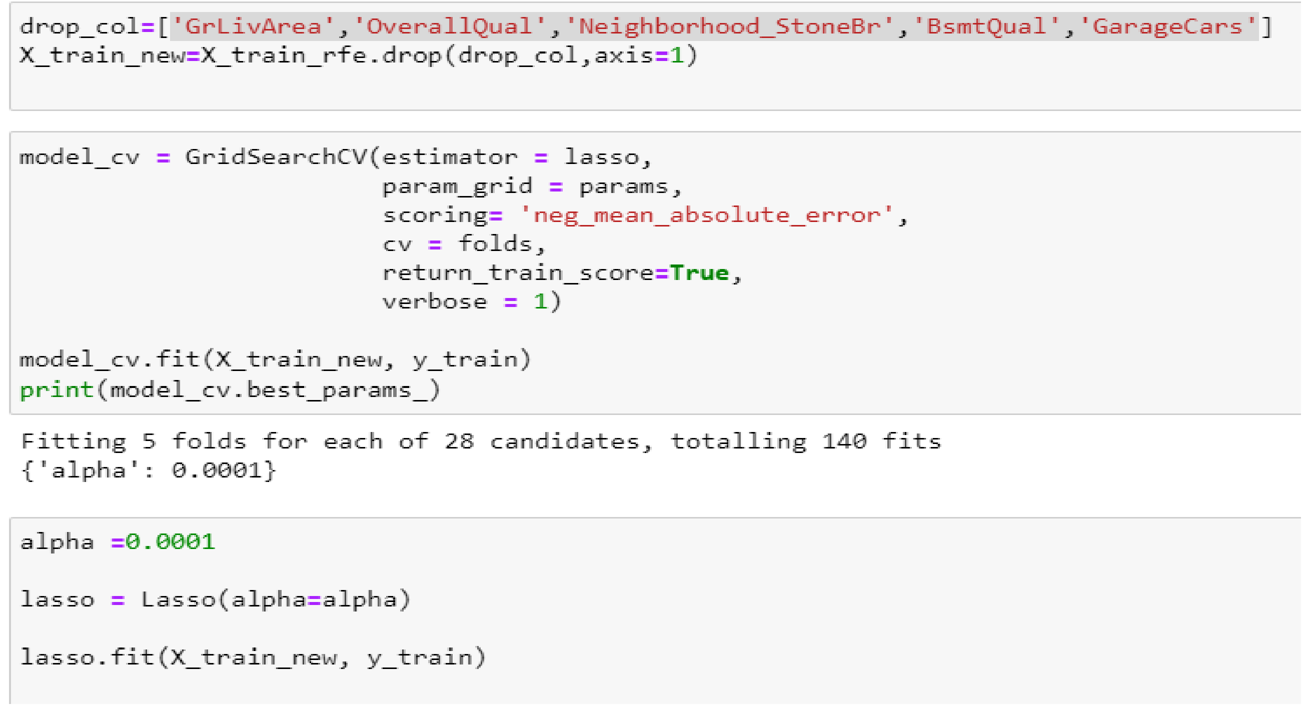


	Metric	Ridge Regression	Lasso Regression
0	R2 Score (Train)	0.839982	0.847750
1	R2 Score (Test)	0.868721	0.873198
2	RSS (Train)	2.056776	1.956932
3	RSS (Test)	0.403593	0.389829
4	MSE (Train)	0.046408	0.045267
5	MSE (Test)	0.041093	0.040387

- Upon comparison of both models, lasso has slightly better r2 score, RSS and MSE in both training and test data
- So **Lasso is the better model**

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?

- In the initial Lasso model, the most important predictor variables are 'GrLivArea', 'OverallQual', 'Neighborhood_StoneBr', 'BsmtQual', 'GarageCars'.
- After removing those predictors and rebuilding the lasso model, the 5 most important predictors are **1stFlrSF, 2ndFlrSF, ExterQual, KitchenQual, Neighborhood_NoRidge**

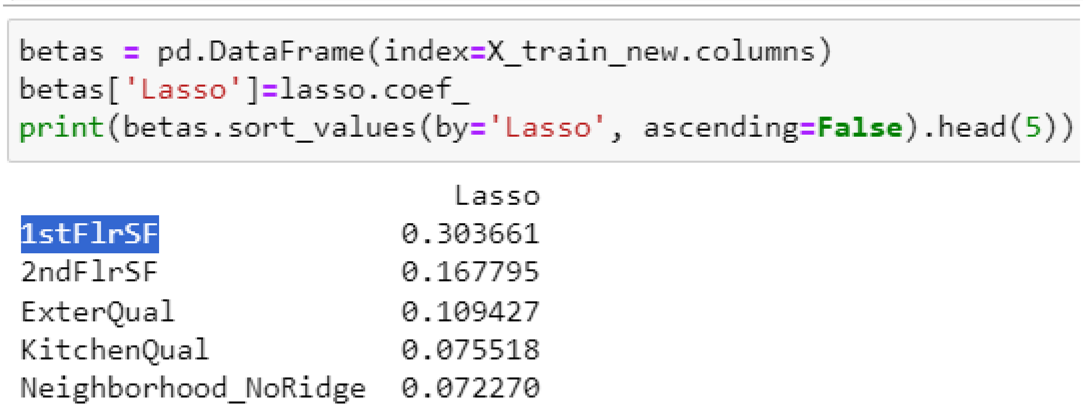


```
drop_col=['GrLivArea','OverallQual','Neighborhood_StoneBr','BsmtQual','GarageCars']
X_train_new=X_train_rfe.drop(drop_col,axis=1)
```

```
model_cv = GridSearchCV(estimator = lasso,
                        param_grid = params,
                        scoring= 'neg_mean_absolute_error',
                        cv = folds,
                        return_train_score=True,
                        verbose = 1)

model_cv.fit(X_train_new, y_train)
print(model_cv.best_params_)
```

```
Fitting 5 folds for each of 28 candidates, totalling 140 fits
{'alpha': 0.0001}
```

```
alpha =0.0001

lasso = Lasso(alpha=alpha)

lasso.fit(X_train_new, y_train)
```

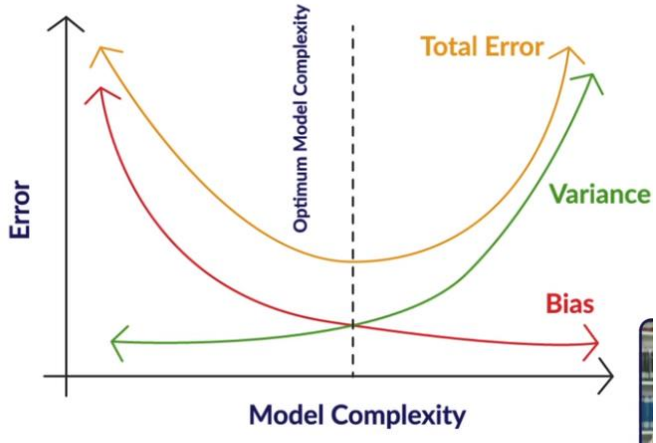


```
betas = pd.DataFrame(index=X_train_new.columns)
betas['Lasso']=lasso.coef_
print(betas.sort_values(by='Lasso', ascending=False).head(5))
```

```
                         Lasso
1stFlrSF              0.303661
2ndFlrSF              0.167795
ExterQual             0.109427
KitchenQual           0.075518
Neighborhood_NoRidge  0.072270
```

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 first step of a good model is thorough EDA(Exploratory data Analysis), which includes understanding the data, handling missing values, outliers, deriving new data
- As per Occam's Razor, choose a simpler/generic model or make the model as simple as possible without losing much data.
- Complex model has high variance and simplest model has high bias. Both are not good.
- Bias in a model is high when it does not perform well on the training data itself, and variance is high when the model does not perform well on the test data.
- Find a balanced generic model as given below



Bias-Variance Tradeoff

- Using cross validation, to ensure that none of the data is left out and making sure that the model doesn't peek into test data
    - types of cross-validation are as follows:
        - K-fold cross-validation
        - Leave one out (LOO) cross-validation → calculate n-1 features
        - Leave P-out (LPO) cross-validation→for p samples with n data points will give nCp train-test split
        - Stratified K-Fold cross-validation-> used when there is imbalance in data(eg. 98% good cust, 2% bad cust)
- Regularisation would prevent making a complex model. It uses hyperparameter to fine tune the complexity of the model
  - Ridge Regression
      - standardise the data whenever working with Ridge regression
      - Disadvantages
          - No feature selection
          - So Interpreting model is hard
  - Lasso Regression
      - As lambda increases, variance decreases and bias increases
      - Eliminates unnecessary feature by making it's coefficient 0
- To get a robust and generalizable model
    - Thorough EDA

- Balanced model between variance and bias
- Cross validating the train data
- Regularising the data