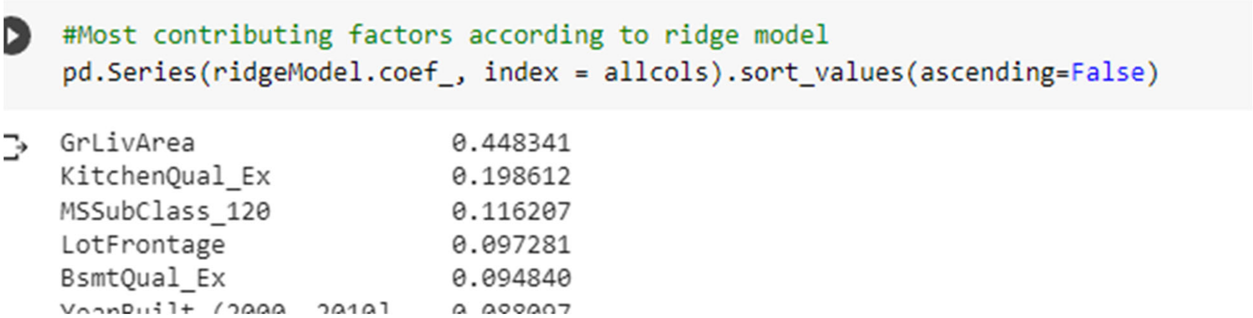
1) **Which variables are significant in predicting the price of a house**

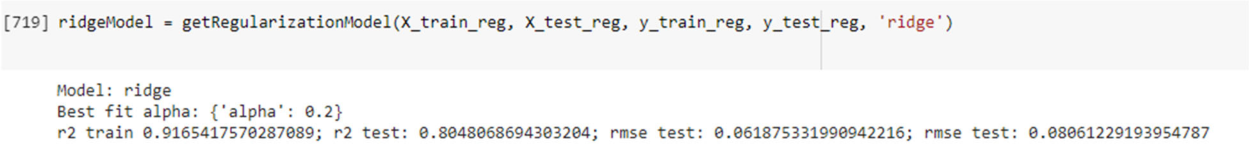


```
#Most contributing factors according to ridge model
pd.Series(ridgeModel.coef_, index = allcols).sort_values(ascending=False)
```

```
GrLivArea                 0.448341
KitchenQual_Ex            0.198612
MSSubClass_120            0.116207
LotFrontage               0.097281
BsmtQual_Ex               0.094840
YearBuilt_(2000, 2010]    0.088007
```

- **Above grade (ground) living area square feet**
- **Excellent kitchen quality**

2) **How well those variables describe the price of a house.**

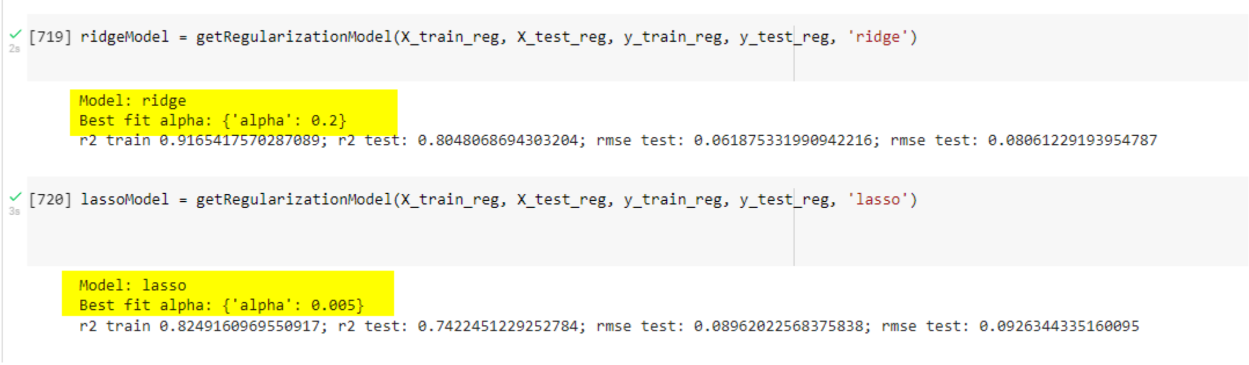


```
[719] ridgeModel = getRegularizationModel(X_train_reg, X_test_reg, y_train_reg, y_test_reg, 'ridge')
```

```
Model: ridge
Best fit alpha: {'alpha': 0.2}
r2 train 0.9165417570287089; r2 test: 0.8048068694303204; rmse test: 0.061875331990942216; rmse test: 0.08061229193954787
```

For ridge model r2 score for train is .90 and test is 0.81. Hence we say the model is pretty accurate

3) **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?**



```
[719] ridgeModel = getRegularizationModel(X_train_reg, X_test_reg, y_train_reg, y_test_reg, 'ridge')
```

```
Model: ridge
Best fit alpha: {'alpha': 0.2}
r2 train 0.9165417570287089; r2 test: 0.8048068694303204; rmse test: 0.061875331990942216; rmse test: 0.08061229193954787
```

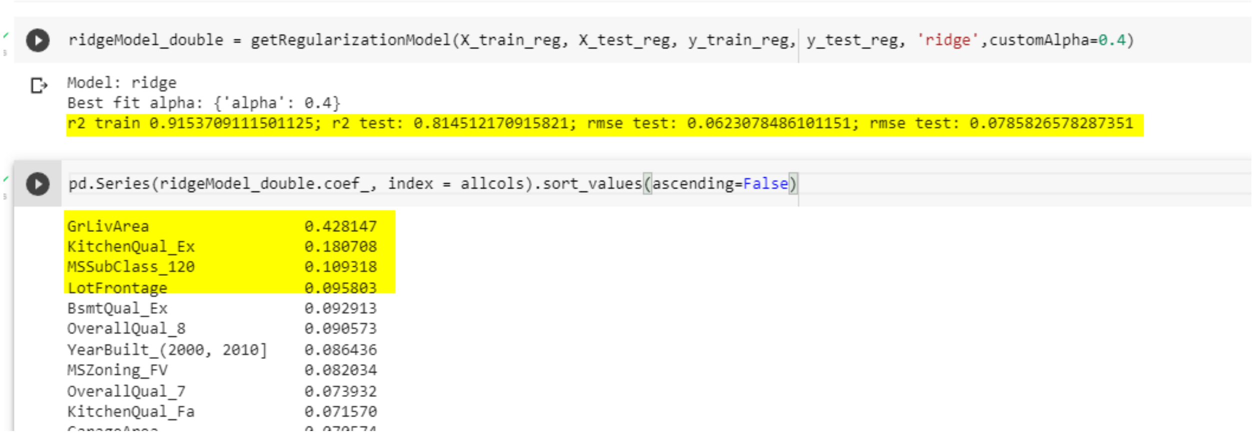
```
[720] lassoModel = getRegularizationModel(X_train_reg, X_test_reg, y_train_reg, y_test_reg, 'lasso')
```

```
Model: lasso
Best fit alpha: {'alpha': 0.005}
r2 train 0.8249160969550917; r2 test: 0.7422451229252784; rmse test: 0.08962022568375838; rmse test: 0.0926344335160095
```

Ridge Alpha: 0.2

Lasso Alpha: 0.005

Ridge double alpha:



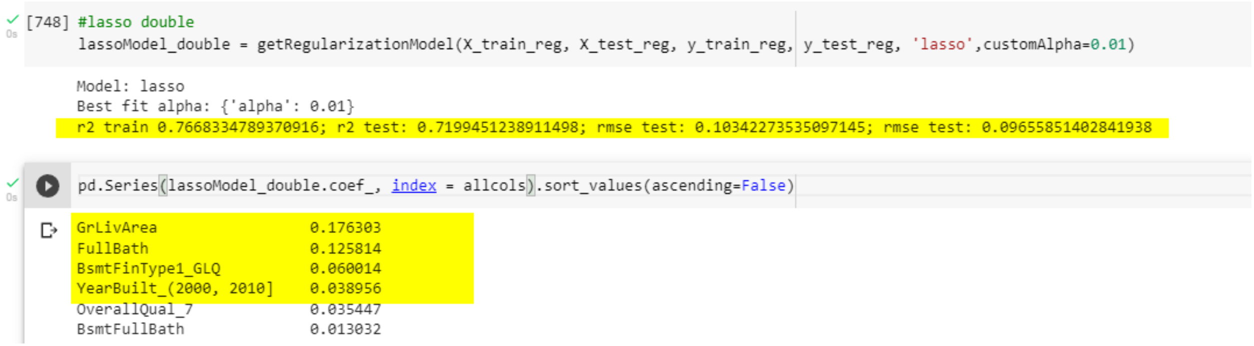
```
ridgeModel_double = getRegularizationModel(X_train_reg, X_test_reg, y_train_reg, y_test_reg, 'ridge',customAlpha=0.4)

Model: ridge
Best fit alpha: {'alpha': 0.4}
r2 train 0.9153709111501125; r2 test: 0.814512170915821; rmse test: 0.0623078486101151; rmse test: 0.0785826578287351
```

```
pd.Series(ridgeModel_double.coef_, index = allcols).sort_values(ascending=False)

GrLivArea               0.428147
KitchenQual_Ex          0.180708
MSSubClass_120          0.109318
LotFrontage             0.095803
BsmtQual_Ex             0.092913
OverallQual_8           0.090573
YearBuilt_(2000, 2010]  0.086436
MSZoning_FV             0.082034
OverallQual_7           0.073932
KitchenQual_Fa          0.071570
GarageArea              0.070574
```

Lasso double alpha:



```
[748] #lasso double
      lassoModel_double = getRegularizationModel(X_train_reg, X_test_reg, y_train_reg, y_test_reg, 'lasso',customAlpha=0.01)

      Model: lasso
      Best fit alpha: {'alpha': 0.01}
      r2 train 0.7668334789370916; r2 test: 0.7199451238911498; rmse test: 0.10342273535097145; rmse test: 0.09655851402841938
```

```
pd.Series(lassoModel_double.coef_, index = allcols).sort_values(ascending=False)

GrLivArea               0.176303
FullBath                0.125814
BsmtFinType1_GLQ        0.060014
YearBuilt_(2000, 2010]  0.038956
OverallQual_7           0.035447
BsmtFullBath            0.013032
```

Increasing alpha double changed the predictors for both the models.

Increasing alpha for ridge increased test r2score and decreased rmse test.

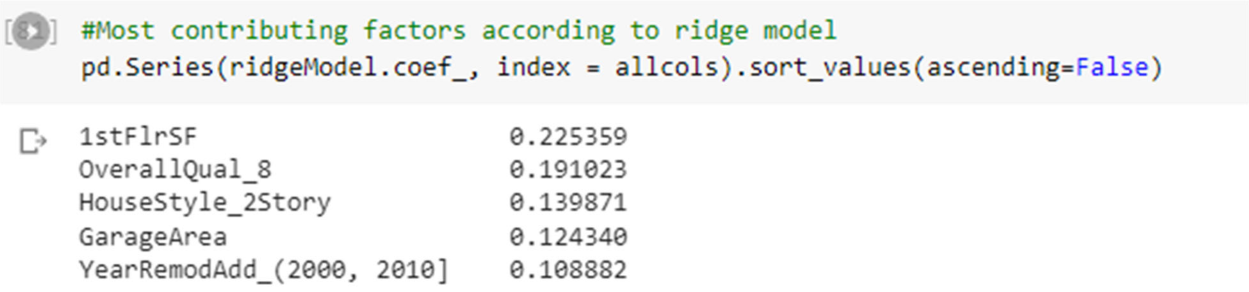
Increasing alpha for lasso decreased r2 score and increased rmse test.

4) **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?**

Ridge would be preferred over lasso because of higher r2 score for both train and test.

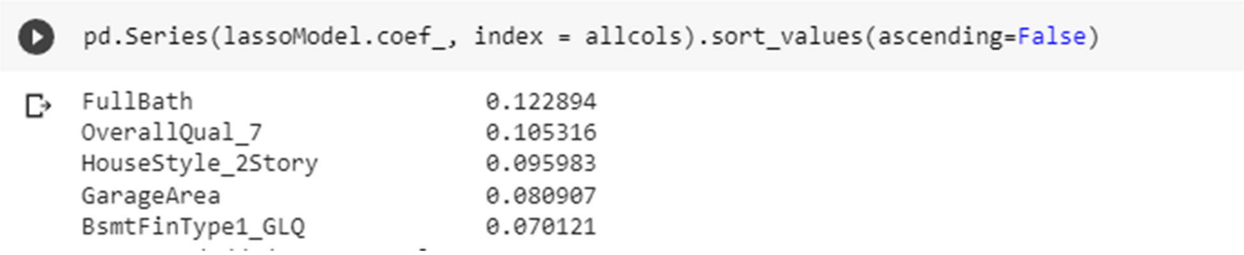
5) **After building the model, you realised that the five most important predictor variables in the lasso model are not available in the incoming data. You will now have to create another model excluding the five most important predictor variables. Which are the five most important predictor variables now?**

- **1^{st} Floor Square Footage**
- **Very good overall material and finish**
- **2 story house**
- **Garage Area**
- **Year Remodeled between 2000 and 2010**



```
#Most contributing factors according to ridge model
pd.Series(ridgeModel.coef_, index = allcols).sort_values(ascending=False)
```

```
1stFlrSF                    0.225359
OverallQual_8               0.191023
HouseStyle_2Story           0.139871
GarageArea                  0.124340
YearRemodAdd_(2000, 2010]   0.108882
```



```
pd.Series(lassoModel.coef_, index = allcols).sort_values(ascending=False)
```

```
FullBath                 0.122894
OverallQual_7            0.105316
HouseStyle_2Story        0.095983
GarageArea               0.080907
BsmtFinType1_GLQ         0.070121
```

6) **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?**
   - **Little or no multi-collinearity between predictor variables**
   - **Error terms are normally distributed**
   - **No specific pattern to error terms**
   - **Errors should be Homoscedastic**
   - **Proper balance between bias and variance by picking optimal alpha value (usng regularization)**
   - **Check BIC for validating the accuracy of model**