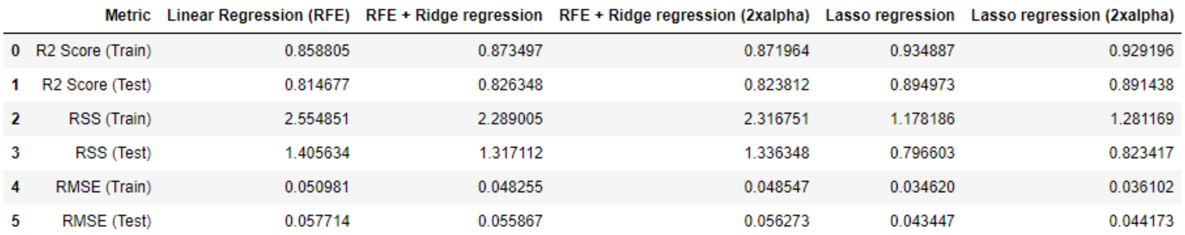
Question 1

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?

Alpha value for Ridge regression was 0.6 and Lasso Regression 0.0001.

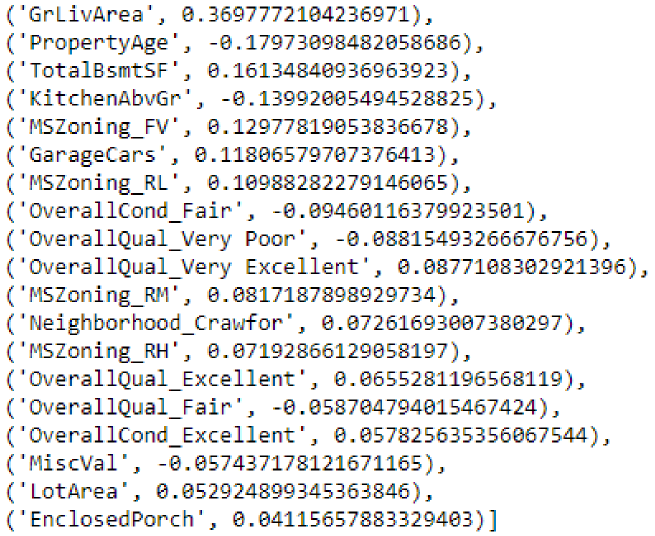
After doubling the alpha values for Ridge regression, there was a slight change in coefficients and the R2 scores were also slightly decreased.



	Metric	Linear Regression (RFE)	RFE + Ridge regression	RFE + Ridge regression (2xalpha)	Lasso regression	Lasso regression (2xalpha)
0	R2 Score (Train)	0.858805	0.873497	0.871964	0.934887	0.929196
1	R2 Score (Test)	0.814677	0.826348	0.823812	0.894973	0.891438
2	RSS (Train)	2.554851	2.289005	2.316751	1.178186	1.281169
3	RSS (Test)	1.405634	1.317112	1.336348	0.796603	0.823417
4	RMSE (Train)	0.050981	0.048255	0.048547	0.034620	0.036102
5	RMSE (Test)	0.057714	0.055867	0.056273	0.043447	0.044173

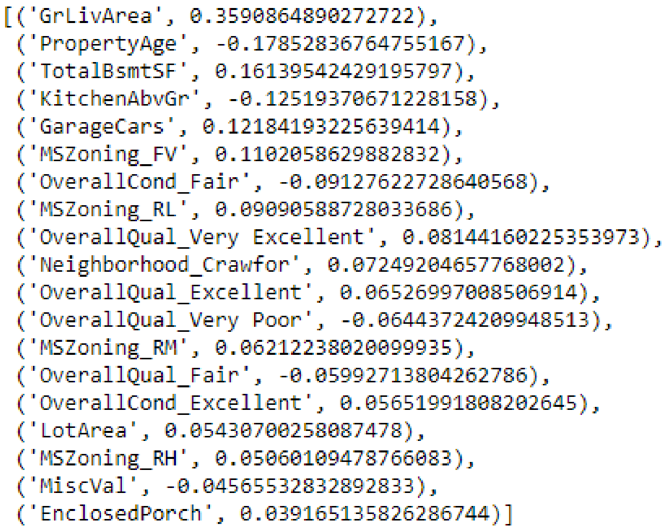
Also, in terms of features, for ridge regression there is only change in the values of the coefficients. The importance remained same.

Original



```
('GrLivArea', 0.3697772104236971),
('PropertyAge', -0.17973098482058686),
('TotalBsmtSF', 0.16134840936963923),
('KitchenAbvGr', -0.13992005494528825),
('MSZoning_FV', 0.12977819053836678),
('GarageCars', 0.11806579707376413),
('MSZoning_RL', 0.10988282279146065),
('OverallCond_Fair', -0.09460116379923501),
('OverallQual_Very Poor', -0.08815493266676756),
('OverallQual_Very Excellent', 0.0877108302921396),
('MSZoning_RM', 0.0817187898929734),
('Neighborhood_Crawfor', 0.07261693007380297),
('MSZoning_RH', 0.07192866129058197),
('OverallQual_Excellent', 0.0655281196568119),
('OverallQual_Fair', -0.05870479401546742 4),
('OverallCond_Excellent', 0.05782563535606754 4),
('MiscVal', -0.05743717812167116 5),
('LotArea', 0.05292489934536384 6),
('EnclosedPorch', 0.04115657883329403)]
```

Doubled

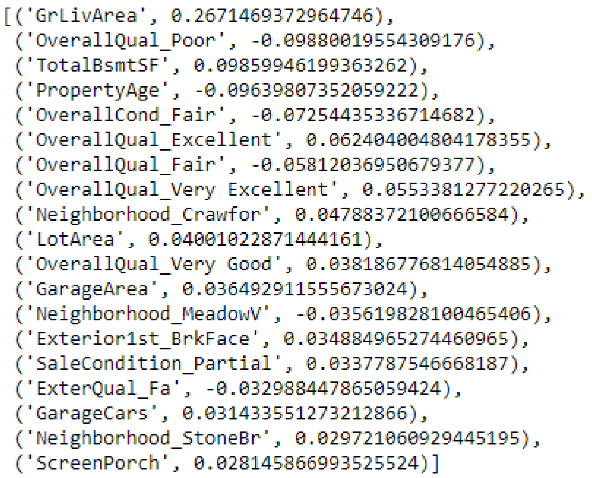


```
[('GrLivArea', 0.3590864890272722),
 ('PropertyAge', -0.17852836764755167),
 ('TotalBsmtSF', 0.16139542429195797),
 ('KitchenAbvGr', -0.12519370671228158),
 ('GarageCars', 0.12184193225639414),
 ('MSZoning_FV', 0.1102058629882832),
 ('OverallCond_Fair', -0.09127622728640568),
 ('MSZoning_RL', 0.09090588728033686),
 ('OverallQual_Very Excellent', 0.08144160225353973),
 ('Neighborhood_Crawfor', 0.07249204657768002),
 ('OverallQual_Excellent', 0.06526997008506914),
 ('OverallQual_Very Poor', -0.06443724209948513),
 ('MSZoning_RM', 0.06212238020099935),
 ('OverallQual_Fair', -0.05992713804262786),
 ('OverallCond_Excellent', 0.05651991808202645),
 ('LotArea', 0.05430700258087478),
 ('MSZoning_RH', 0.05060109478766083),
 ('MiscVal', -0.04565532832892833),
 ('EnclosedPorch', 0.03916513582628674 4)]
```

In case of Lasso though, most of the features changed their coefficients and also some new parameters were added.

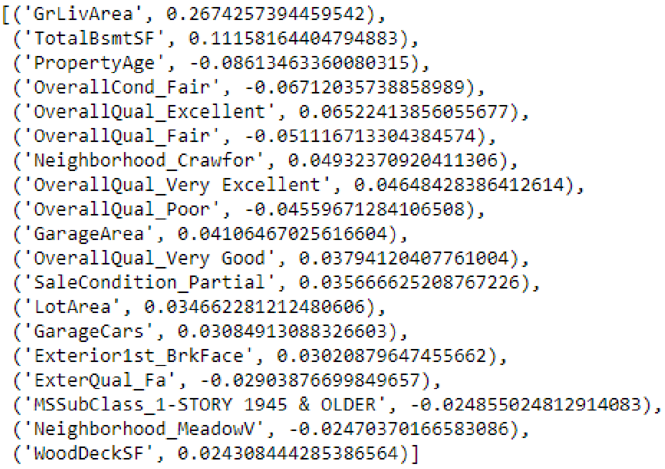
P.S. instead of comparing all the parameters only 19 parameters were considered for Lasso.

Original



```
[('GrLivArea', 0.2671469372964746),
 ('OverallQual_Poor', -0.09880019554309176),
 ('TotalBsmtSF', 0.09859946199363262),
 ('PropertyAge', -0.09639807352059222),
 ('OverallCond_Fair', -0.07254435336714682),
 ('OverallQual_Excellent', 0.062404004804178355),
 ('OverallQual_Fair', -0.05812036950679377),
 ('OverallQual_Very Excellent', 0.0553381277220265),
 ('Neighborhood_Crawfor', 0.04788372100666584),
 ('LotArea', 0.04001022871444161),
 ('OverallQual_Very Good', 0.03818677681405488 5),
 ('GarageArea', 0.03649291155567302 4),
 ('Neighborhood_MeadowV', -0.03561982810046540 6),
 ('Exterior1st_BrkFace', 0.03488496527446096 5),
 ('SaleCondition_Partial', 0.0337787546668187),
 ('ExterQual_Fa', -0.03298844786505942 4),
 ('GarageCars', 0.03143355127321286 6),
 ('Neighborhood_StoneBr', 0.02972106092944519 5),
 ('ScreenPorch', 0.02814586699352552 4)]
```

Doubled



```
[('GrLivArea', 0.2674257394459542),
 ('TotalBsmtSF', 0.11158164404794883),
 ('PropertyAge', -0.08613463360080315),
 ('OverallCond_Fair', -0.06712035738858989),
 ('OverallQual_Excellent', 0.06522413856055677),
 ('OverallQual_Fair', -0.05111671330438457 4),
 ('Neighborhood_Crawfor', 0.04932370920411306),
 ('OverallQual_Very Excellent', 0.04648428386412614),
 ('OverallQual_Poor', -0.04559671284106508),
 ('GarageArea', 0.04106467025616604),
 ('OverallQual_Very Good', 0.03794120407761004),
 ('SaleCondition_Partial', 0.03566662520876722 6),
 ('LotArea', 0.03466228121248060 6),
 ('GarageCars', 0.03084913088326603),
 ('Exterior1st_BrkFace', 0.03020879647455662),
 ('ExterQual_Fa', -0.02903876699849657),
 ('MSSubClass_1-STORY 1945 & OLDER', -0.02485502481291408 3),
 ('Neighborhood_MeadowV', -0.02470370166583086),
 ('WoodDeckSF', 0.02430844428538656 4)]
```

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?

The lambda value for:

Ridge: 0.6

Lasso: 0.0001

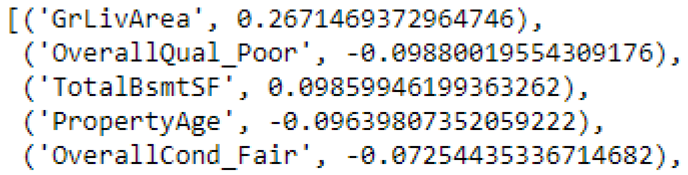
In both the models we got good results. However, RFE + Ridge seems to be more consistent. There is not much difference in R2 score of Train and test data set. Whereas in case of Lasso, the difference is visible.

Therefore, I would go with RFE + Ridge.

Question 3

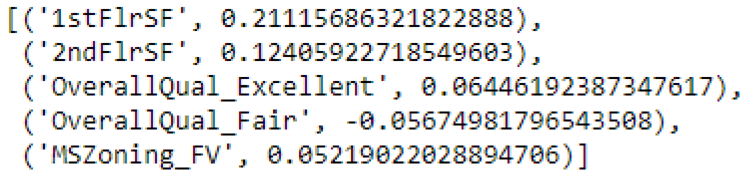
After building the model, you realised that the five most important predictor variables in the lasso model are not available in the incoming data. You will now have to create another model excluding the five most important predictor variables. Which are the five most important predictor variables now?

Original top 5 features:



```
[('GrLivArea', 0.2671469372964746),
 ('OverallQual_Poor', -0.09880019554309176),
 ('TotalBsmtSF', 0.09859946199363262),
 ('PropertyAge', -0.09639807352059222),
 ('OverallCond_Fair', -0.07254435336714682),
```

After removing above, new top 5 features



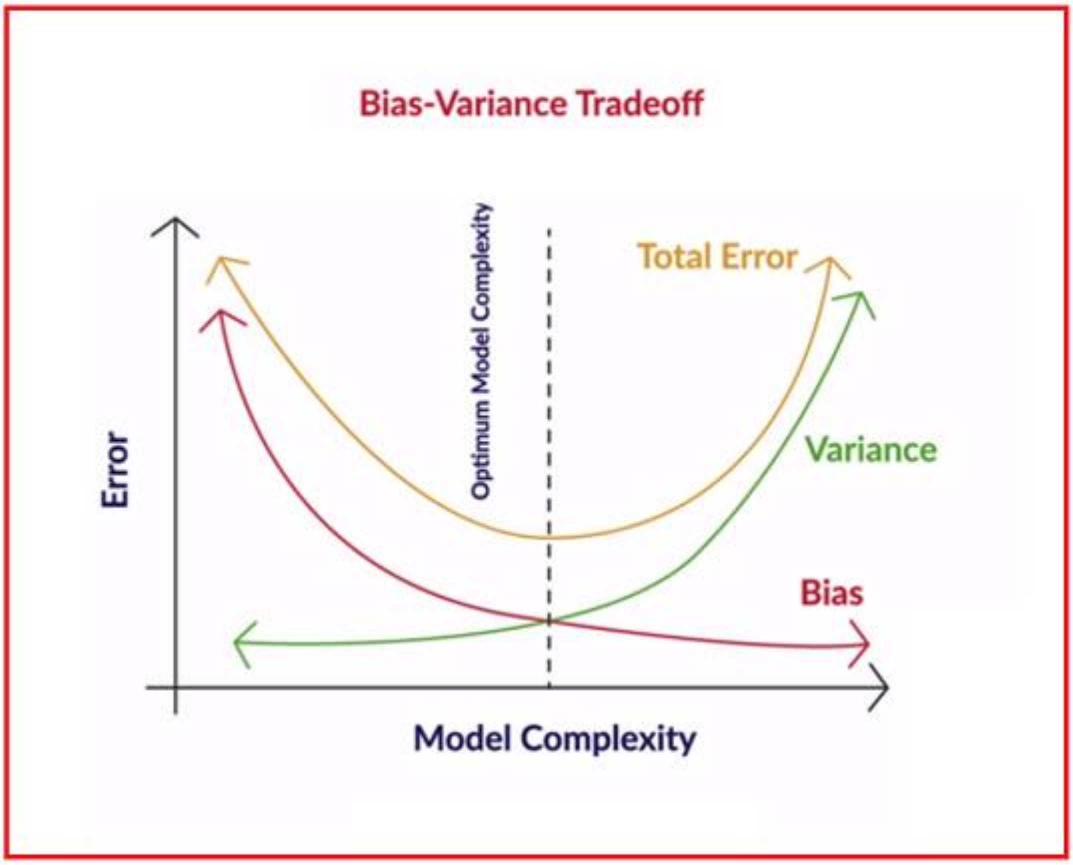
```
[('1stFlrSF', 0.21115686321822888),
 ('2ndFlrSF', 0.12405922718549603),
 ('OverallQual_Excellent', 0.06446192387347617),
 ('OverallQual_Fair', -0.05674981796543508),
 ('MSZoning_FV', 0.05219022028894706)]
```

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?

Robustness of model depends on how well it can handle outliers. Therefore, while doing EDA it is important that outliers are removed, only the ones relevant to the dataset should be retained. Some of the ways of identifying / removing outliers are using quantiles (typically 0.95) or standard deviations (3-5). Any value above these can be considered as an outlier. It is also important to ensure that while removing outliers not too much of important data is getting lost. This ensures accuracy of the model.

A model can be made generalized by keeping it as simple as possible. It is important to keep optimum number of features in the model keeping in mind the Bias-Variance trade off.



As the number of features increase the model starts learning the data and thus overfits. Similarly, if there are too less features, the model cannot react to complex real-world data.