HOUSE PRICING CASE STUDY

minhngc4795@gmail.com

Assignment-based Subjective Questions

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 value of alpha:
  - Ridge Regression: 0.8
  - Lasso Regression: 50
- If we double the value of alpha for both the Ridge and Lasso regression models, the penalty term applied to the regression coefficients will be larger. This indicates that the regularization terms' impact on model fit will continue to increase, eventually leading to:
  - Higher regression coefficient decreasing towards zero
  - Lasso model feature set sparsity was increased.
  - Regression coefficients with smaller values may be more interpretable.
  - More bias in the model as regularization increases, but perhaps reduced variance and greater performance on unknown data.
  - Potentially limiting the model's ability to overfit on the training dataset.
- The Ridge (top 10 importants) - there are some new features appeared in top 10 when changing the lambda

```python
betas.sort_values(by=["Ridge"], ascending = False).head(10)
```

✓ 0.8s

	Linear	Ridge	Ridge Double	Lasso	Lasso Double
RoofMatl_WdShngl	130727.892425	113999.934660	99289.847315	125138.294154	115860.303026
OverallQual	117904.687302	109893.060031	103261.322639	118346.876964	118872.242372
LotArea	134316.905892	76146.866866	56658.036191	78568.429766	52135.179179
BsmtFinSF1	87268.545974	74655.325124	64695.730390	73976.627166	55317.723199
GarageArea	75793.632218	74149.540854	72027.992511	75665.354729	76270.077479
Neighborhood_StoneBr	69715.821338	66426.958507	63472.030911	64475.080412	61950.106475
2ndFlrSF	49739.774947	48975.019231	48731.742633	48469.777869	47834.798743
Neighborhood_NridgHt	48987.501762	47708.370416	46605.691735	47717.371297	46622.556700
BedroomAbvGr	44680.615293	45170.672905	43507.746041	43229.038305	41463.576283
SaleType_Con	57429.157275	40685.686436	31534.848974	30514.828807	3391.775650

```python
betas.sort_values(by=["Ridge Double"], ascending = False).head(10)
```

✓ 0.1s

	Linear	Ridge	Ridge Double	Lasso	Lasso Double
OverallQual	117904.687302	109893.060031	103261.322639	118346.876964	118872.242372
RoofMatl_WdShngl	130727.892425	113999.934660	99289.847315	125138.294154	115860.303026
GarageArea	75793.632218	74149.540854	72027.992511	75665.354729	76270.077479
BsmtFinSF1	87268.545974	74655.325124	64695.730390	73976.627166	55317.723199
Neighborhood_StoneBr	69715.821338	66426.958507	63472.030911	64475.080412	61950.106475
LotArea	134316.905892	76146.866866	56658.036191	78568.429766	52135.179179
2ndFlrSF	49739.774947	48975.019231	48731.742633	48469.777869	47834.798743
Neighborhood_NridgHt	48987.501762	47708.370416	46605.691735	47717.371297	46622.556700
BedroomAbvGr	44680.615293	45170.672905	43507.746041	43229.038305	41463.576283
Neighborhood_NoRidge	35143.632653	35728.125338	35847.215152	35553.497825	35408.529246

- The Lasso (top 10 importants) - no new features, just changing the value in top 10 important features

```
betas.sort_values(by=["Lasso"], ascending = False).head(10)
```

[97] ✓ 0.5s

	Linear	Ridge	Ridge Double	Lasso	Lasso Double
RoofMatl_WdShngl	130727.892425	113999.934660	99289.847315	125138.294154	115860.303026
OverallQual	117904.687302	109893.060031	103261.322639	118346.876964	118872.242372
LotArea	134316.905892	76146.866866	56658.036191	78568.429766	52135.179179
GarageArea	75793.632218	74149.540854	72027.992511	75665.354729	76270.077479
BsmtFinSF1	87268.545974	74655.325124	64695.730390	73976.627166	55317.723199
Neighborhood_StoneBr	69715.821338	66426.958507	63472.030911	64475.080412	61950.106475
2ndFlrSF	49739.774947	48975.019231	48731.742633	48469.777869	47834.798743
Neighborhood_NridgHt	48987.501762	47708.370416	46605.691735	47717.371297	46622.556700
BedroomAbvGr	44680.615293	45170.672905	43507.746041	43229.038305	41463.576283
Neighborhood_NoRidge	35143.632653	35728.125338	35847.215152	35553.497825	35408.529246

```
betas.sort_values(by=["Lasso Double"], ascending = False).head(10)
```

[98] ✓ 0.6s

	Linear	Ridge	Ridge Double	Lasso	Lasso Double
OverallQual	117904.687302	109893.060031	103261.322639	118346.876964	118872.242372
RoofMatl_WdShngl	130727.892425	113999.934660	99289.847315	125138.294154	115860.303026
GarageArea	75793.632218	74149.540854	72027.992511	75665.354729	76270.077479
Neighborhood_StoneBr	69715.821338	66426.958507	63472.030911	64475.080412	61950.106475
BsmtFinSF1	87268.545974	74655.325124	64695.730390	73976.627166	55317.723199
LotArea	134316.905892	76146.866866	56658.036191	78568.429766	52135.179179
2ndFlrSF	49739.774947	48975.019231	48731.742633	48469.777869	47834.798743
Neighborhood_NridgHt	48987.501762	47708.370416	46605.691735	47717.371297	46622.556700
BedroomAbvGr	44680.615293	45170.672905	43507.746041	43229.038305	41463.576283
Neighborhood_NoRidge	35143.632653	35728.125338	35847.215152	35553.497825	35408.529246

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?

- I will choose the Lasso regression due to the fact that they are usually preferred when dealing with a large number of features and some of them may not be relevant to the outcome, as it can perform feature selection and shrink the coefficients of less important features to zero.
- Beside that, the evaluation metrics on the test set when comparison between models, the Lasso model performs the best

	Metric	Linear Regression	Ridge Regression	Ridge Double Regression	Lasso Regression	Lasso Doulbe Regression
0	R2 Score (Train)	8.615700e-01	8.590932e-01	8.557855e-01	8.572467e-01	8.514316e-01
1	R2 Score (Test)	8.202198e-01	8.213665e-01	8.206915e-01	8.226401e-01	8.207266e-01
2	RSS (Train)	8.402355e+11	8.552686e+11	8.753456e+11	8.664766e+11	9.017727e+11
3	RSS (Test)	5.639493e+11	5.603523e+11	5.624696e+11	5.563569e+11	5.623596e+11
4	MSE (Train)	2.868717e+04	2.894266e+04	2.928039e+04	2.913168e+04	2.971910e+04
5	MSE (Test)	3.584163e+04	3.572715e+04	3.579458e+04	3.559955e+04	3.579108e+04

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 the filter top 5 features, the optimal lambda of Lasso change from 50 to 100, and the top 5 features importance updated to

```
Top 10 Important Predictor Variables:
Index(['BedroomAbvGr', 'Neighborhood_NridgHt', '2ndFlrSF', 'LotArea',
       'BsmtFinSF1', 'Neighborhood_StoneBr', 'GarageArea', 'Condition2_PosN',
       'RoofMatl_WdShngl', 'OverallQual'],
      dtype='object')
```

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?

I used the following ways to validate that the model is robust and generalisable:

- Splitting the data into training and testing sets, and using cross-validation techniques to evaluate the performance of the model on unseen data.
- Regularizing the model to avoid overfitting and improve its ability to generalize to new data.
- Assessing the impact of different hyperparameters and tuning them using grid search or other optimization methods.
- Ensuring that the training data is representative of the population and the problem domain, and that any biases or confounding factors are appropriately accounted for.

In the evaluation section, I have applied multiple metrics to measure the bias and variances of the model to ensure that the result reflect correctly the generalize cases in real-world and not being bias by any factors