SUBJECTIVE QUESTIONS

I. 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?

- Optimal value of alpha for Ridge Regression = 10
- Optimal value of alpha for Lasso Regression = 0.0001

When we double the value of alpha in Ridge and Lasso regression, the regularization effect on the model will increase.

In Ridge regression, doubling the value of alpha will increase the penalty term in the loss function, which will result in a decrease in the magnitude of the coefficients. This decrease in magnitude will lead to a simpler model that is less prone to overfitting.

In Lasso regression, doubling the value of alpha will result in more coefficients being shrunk to zero. This is because Lasso regression has a sparsity-inducing property, where it tends to push coefficients towards zero and result in a sparse model. A higher value of alpha increases this effect and results in even more coefficients being set to zero.

Our metrics for these models will change accordingly as below (please refer to part VIII in Jupyter Notebook - Advanced Regression - Surprise Housing Case Study – Anh Ngo):

Metric	Ridge Regression	Lasso Regression
R2 Score (Train)	0.923953	0.949175
R2 Score (Test)	0.893315	0.775896
RSS (Train)	9.691137	6.476856
RSS (Test)	5.746857	12.071901
MSE (Train)	0.009492	0.006344
MSE (Test)	0.013121	0.027561
RMSE (Train)	0.097426	0.079647
RMSE (Test)	0.114546	0.166016

Changes in Ridge Regression metrics:

- R2 score of train set decreased from 0.94 to 0.92
- R2 score of test set remained unchanged at 0.89

Changes in Lasso metrics:

- R2 score of train set slightly decreased from 0.954 to 0.949
- R2 score of test set increased from 0.74 to 0.78

For Ridge regression, the decrease in R2 score for the train set from 0.94 to 0.92 suggests that the model is now fitting the training data slightly worse than before. This is expected as a higher value of alpha leads to increased regularization, which can result in a simpler model that may not fit the training data as well. However, the R2 score for the test set remained unchanged at 0.89, indicating that the model's ability to generalize to unseen data has not been affected.

For Lasso regression, the slight decrease in R2 score for the train set from 0.954 to 0.949 suggests that the model is fitting the training data slightly worse than before, but the increase in the R2 score for the test set from 0.74 to 0.78 suggests that the model's ability to generalize to unseen data has improved. This is expected as increasing the value of alpha in Lasso regression results in more coefficients being set to zero, which can help in reducing overfitting and improving the model's generalization ability.

The most important predictor variables:

- OverallQual_9
- OverallCond_9
- OverallQual_8
- Neighborhood_Crawfor
- OverallCond_8
- SaleType_ConLD
- Condition2_PosA
- Neighborhood_ClearCr
- OverallCond_7
- OverallQual_7

II. 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 choice of the model depends on the specific problem we are trying to solve and the evaluation metric that is most important for our use case.

In general, a higher R2 score and a lower value of MSE and RMSE indicate better performance of the model. Based on the given metrics, we can see that the Lasso Regression model outperforms the Ridge Regression model on all the evaluation metrics except for the RSS on the training set.

Metric	Ridge Regression	Lasso Regression
R2 Score (Train)	0.923953	0.949175
R2 Score (Test)	0.893315	0.775896
RSS (Train)	9.691137	6.476856
RSS (Test)	5.746857	12.071901
MSE (Train)	0.009492	0.006344
MSE (Test)	0.013121	0.027561
RMSE (Train)	0.097426	0.079647
RMSE (Test)	0.114546	0.166016

However, to identify the most important features, the Ridge Regression model might be more appropriate as it only shrinks the coefficients towards zero, while Lasso Regression can lead to some coefficients being set to exactly zero.

In general, the final choice of the model depends on specific requirements and priorities.

- If we have too many variables and primary goal is feature selection, then we will use Lasso.
- If we don't want to get too large coefficients and reduction of coefficient, we should go with Ridge Regression.

III. 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?

The detailed codes are provided in the part VIII in Jupyter Notebook - Advanced Regression - Surprise Housing Case Study – Anh Ngo.

To find new five most important predictor variables, we will:

1. Drop the current top 5 features in Lasso model
2. Build the model again
3. Find new optimal value of alpha (0.0001)
4. Fitting the model on training data
5. Make Predictions
6. Checking metrics again

```
R-Squared (Train) = 0.9517510395833402
R-Squared (Test) = 0.7369689453879364
RSS (Train) = 6.148625042395704
RSS (Test) = 14.168795718770149
MSE (Train) = 0.006022159688928211
MSE (Test) = 0.032348848672991204
RMSE (Train) = 0.07760257527252695
RMSE (Test) = 0.17985785685643874
```

7. Spot our new top five as below:

```
#Top 5 coefficients of Lasso

betas['Lasso'].sort_values(ascending=False)[:5]
```

```
Condition2_PosA    0.363338
SaleType_ConLD     0.167936
2ndFlrSF           0.086431
MSSubClass_70      0.081918
1stFlrSF           0.078386
```

Our new five most important predictor variables are:

1. Condition2_PosA (Proximity to various conditions is Adjacent to postive off-site feature)
2. SaleType_ConLD (Contract Low Down Sale)
3. 2ndFlrSF (Second floor square feet)
4. MSSubClass_70 (2-STORY 1945 & OLDER dwelling)
5. 1stFlrSF (First Floor square feet)

IV. 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?

A model is robust and generalizable when any variation in the data does not affect its performance much. It is expected to perform well on unseen data, while a model that overfits the training data will perform poorly on unseen data. In other words, the model should not be too complicated.

To make sure a model is robust and generalizable, we have to ensure it doesn't overfit. Overfitting occurs when a model is too complex and captures noise in the training data, leading to poor performance on unseen data. This is because an overfitting model has very high variance and a smallest change in data affects the model prediction heavily. Such a model will identify all the patterns of a training data, but fail to pick up the patterns in unseen test data.

There are several steps we can take to ensure that a model is robust and generalizable:

1. Use a diverse and representative dataset: A diverse dataset ensures that the model has exposure to a wide range of scenarios, while a representative dataset ensures that the model is not biased towards any particular subset of the population.
2. Regularization: Regularization techniques can prevent the model from overfitting the training data, ensuring that the model learns generalizable patterns.
3. Cross-validation: it helps in evaluating the model's performance on different subsets of data, which ensures that the model is robust and not overfitting to a particular set of data.
4. Hyperparameter Tuning: Hyperparameters control the model's behavior during training. Optimizing the hyperparameters can improve the model's generalization ability.
5. Testing: testing the model on a holdout dataset can provide an estimate of its generalization ability.

While it is possible to achieve high accuracy on the training data by overfitting, this does not guarantee high accuracy on new data. A model with good generalization ability will have a lower chance of making errors on new data, and this can lead to more accurate predictions in the real world. In general, we must find strike some balance between model accuracy and complexity. This can be achieved by Regularization techniques like Ridge Regression and Lasso.