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?

Answer:

The optimal alpha values for Ridge and Lasso were found to be 10 and 316.23, respectively.

The alpha parameter in Ridge and Lasso regressions is a regularization term that controls the magnitude of the coefficients in the model. Regularization is an approach to prevent overfitting by adding a penalty term to the loss function that the model seeks to minimize.

For Ridge regression (L2 regularization), the regularization term is the sum of the squares of the coefficients, multiplied by alpha. If we increase alpha, the penalty for large coefficients becomes more severe. This means the model will be less likely to assign a large weight to any single feature, potentially leading to less overfitting. However, if alpha is extremely high, the model might become exceedingly simple and underfit the data, leading to higher bias.

For Lasso regression (L1 regularization), the regularization term is the sum of the absolute values of the coefficients, multiplied by alpha. Increasing alpha in Lasso regression has the same general effect of reducing overfitting. However, a distinctive feature of Lasso regression is that it can shrink some coefficients to exactly zero, effectively performing feature selection. If we increase alpha substantially, an increasing number of features might be "lassoed" out of the model, rendering it overly simple and again leading to underfitting.

In general, if we were to choose double the value of alpha for both Ridge and Lasso regression the model would become more regularized, reducing the risk of overfitting. Furthermore, the coefficients of the model would generally become smaller. In the case of Lasso, some coefficients might become exactly zero, reducing the complexity of the model. The model might become less sensitive to changes in individual features, which can be beneficial should we expect the model to be robust to small changes in the input data. However, if alpha is increased substantially, the model might become exceedingly simple, and its predictive performance on both the training set and the test set could decrease, resulting in underfitting.

To determine the most important predictors after doubling the alpha for both ridge and lasso, the following code snippet has been executed:

```
pd.set_option('display.max_rows', None)

alpha_ridge = 10 * 2
alpha_lasso = 316.23 *2

models = {'Ridge': Ridge(alpha=alpha_ridge), 'Lasso': Lasso(alpha=alpha_lasso)}

for name, model in models.items():
    model.fit(X_train, y_train)

    # Creating a DataFrame to store the model coefficients
    coef_df = pd.DataFrame()
    coef_df['Variable'] = X_train.columns  # Ensuring all features are included
```

```
    coef_df['Coefficient'] = model.coef_

    # Filtering the coefficients that are exactly zero (for Lasso)
    coef_df = coef_df[coef_df['Coefficient'] != 0]

    # Sorting the DataFrame by the absolute value of the coefficients
    coef_df = coef_df.reindex(coef_df.Coefficient.abs().sort_values(ascending=False).index)

    print(f'\n{name} Regression with alpha={model.alpha}')
    print('Most important predictors after doubling the alpha value:')
    print(coef_df)
```

Most important predictors after doubling the alpha value for Ridge Regression:

Variable	Coefficient
TotRmsAbvGrd_10	31411.124004
Neighborhood_NoRidge	30437.572472
OverallQual_10	29223.799243
GrLivArea	28852.825893
2ndFlrSF	27659.579475
OverallQual_9	25411.504777
FullBath	25288.228965
GarageCars	24166.736710
Fireplaces	21851.055221

Most important predictors after doubling the alpha value for Lasso Regression:

Variable	Coefficient
GrLivArea	161363.102489
OverallQual_10	55307.145951
OverallQual_9	52452.445017
GarageCars	45211.871426
Neighborhood_NoRidge	35257.910874
Fireplaces	31772.047014
TotRmsAbvGrd_10	27060.303544
OverallQual_8	24931.754638
BsmtExposure_Gd	18901.705641

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?

Answer:

The following performance table has been obtained as output upon evaluation of the models built:

Performance Table

	Regression	Dataset	RSS	R2	Adj. R2	MSE	NRMSE
0	Linear	Train	2.852138e+12	0.553005	0.549026	2.793475e+09	-52853.331582
1	Linear	Test	1.542873e+12	0.452632	0.441122	3.522541e+09	-59350.995782
2	Ridge	Train	1.750575e+12	0.725645	0.723203	1.714569e+09	-41407.356141
3	Ridge	Test	7.596290e+11	0.730505	0.724838	1.734313e+09	-41645.081050
4	Lasso	Train	1.289688e+12	0.797877	0.796077	1.263161e+09	-35540.979030
5	Lasso	Test	5.527885e+11	0.803886	0.799762	1.262074e+09	-35525.682431

Overall, lower values of RSS, MSE, and NRMSE and higher values of R2 and Adj. R2 indicate better model performance.

Given the results, the Lasso Regression model appears to be the most appropriate model to apply to this data. Its performance on the training set is strong, suggesting a good fit to the data, and its performance on the test set is also the best among the three models, indicating that it generalizes well to unseen data. Furthermore, the Lasso model's ability to perform feature selection by forcing some coefficients to zero is beneficial for the problem at hand owing to interpretability.

However, the choice between Ridge and Lasso would depend on additional considerations. For instance, if the aim is to perform feature selection and create a simpler model with fewer features, the Lasso model may be preferred because it can reduce some coefficients to zero, effectively excluding those features from the model. On the other hand, if all features are considered important and should be included in the model, then Ridge may be the preferred choice. In other words, ridge Regression works well if there are numerous parameters of comparable values (i.e., when most predictors impact the response). Ridge Regression tends to have better predictive accuracy than the Linear Regression when multicollinearity is present among the predictor variables. Ridge Regression is more stable and less likely to overfit than Lasso when the predictors are highly correlated.

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?

Answer:

To answer this question, the following lines of code have been executed

'''In the Lasso Regression output, the five variables with the highest absolute coefficients are
GrLivArea: 216,326.78
OverallQual_10: 120,041.59
OverallQual_9: 102,387.77
GarageCars: 59,249.79
Fireplaces: 50,927.78
These predictors have the highest coefficients, indicating that they contribute the most to predicting the dependent variable in the model.'''

'''To determine the five most important predictor variables after excluding the previous ones, we would need to rerun the Lasso regression model with the remaining variables. '''

```
# creating a list of the variables to be excluded
excluded_vars = ['GrLivArea', 'OverallQual_10', 'OverallQual_9', 'GarageCars', 'Fireplaces']

# creating a new DataFrame that does not include these variables
X_train_new = X_train.drop(excluded_vars, axis=1)
X_test_new = X_test.drop(excluded_vars, axis=1)

# creating a new Lasso model
lasso_new = LassoCV(cv=5, random_state=0)

# fitting the model
lasso_new.fit(X_train_new, y_train)

# obtaining the coefficients
coef_new = pd.Series(lasso_new.coef_, index = X_train_new.columns)

# printing the coefficients
print(coef_new.sort_values(ascending=False).head(5))
```

The five most important predictors obtained after running this code are as follows:

1stFlrSF	156815.785177
2ndFlrSF	88120.412487
Neighborhood_NoRidge	46675.245746
RoofMatl_WdShngl	44516.743009
GarageArea	42816.109409

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?

Answer:

Ensuring the robustness and generalizability of a model often involves a combination of the following:

1. **Cross-validation**: Splitting the data into multiple subsets and training/testing the model on different combinations of these subsets. This helps ensure that the model performs well on different sections of the data and is not overly tuned to specific characteristics.
2. **Regularization**: Techniques like Ridge and Lasso are used to prevent overfitting by adding a penalty term to the loss function. This discourages overly complex models by effectively reducing the number of features included in the model. The selection of an appropriate regularization parameter (alpha) is critical.
3. **Feature Selection**: This includes using statistical tests to select the most informative features, or using techniques like Lasso that can perform feature selection as part of the model training process.
4. **Evaluating performance on a separate test set**: This helps ensure the model's ability to generalize to unseen data. The model's performance on the test set is usually a good indicator of how it will perform on new, unseen data.

The implications for the accuracy of the model are as follows:

- An overfit model might have high accuracy on the training data but will likely perform poorly on unseen data because it is overly adapted to the training data and not sufficiently generalized.
- A well-generalized model will perform comparably on both training and unseen data.

Considering the performance table in the output,

- The Ridge and Lasso models show better performance (higher R2, lower RMSE) on both the training and test datasets relative to the Linear Regression model.
- The Ridge and Lasso models also show comparable performance on the training and test datasets (R2 and RMSE are close in value), suggesting that they are well-generalized.
- The Lasso model performs slightly better than the Ridge model in this case, on both training and test data, and would be the preferred model based on the results.

The best alphas for Ridge and Lasso in the output are 10.0 and 316.23, respectively, indicating that these values offer the best trade-off between bias and variance for each model.

In summary, the Lasso model appears to be the most robust and generalizable model based on the output, with high R2 values and low RMSE on both the training and test datasets. However, ongoing monitoring and validation would be needed to ensure its performance remains consistent over time and as new data is introduced.