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 value for ridge regression is 10.0 and for lasso regression is 500.0

Doubling alpha in both Ridge and Lasso regression would lead to more regularization, resulting in models with smaller coefficients and potentially improved generalization performance, with Lasso also performing feature selection.

Following are the three most important predictors before and after doubling the alpha

S No.	Ridge	Lasso	Ridge (Double Alpha)	Lasso (Double Alpha)
1	OverallQual	GrLivArea	OverallQual	GrLivArea
2	GrLivArea	OverallQual	TotRmsAbvGrd	OverallQual
3	Neighborhood_NoRidge	GarageCars	Neighborhood_NoRidge	GarageCars

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:

Ridge Regression

1. Ridge regression is generally preferred when there is a possibility of multicollinearity among the predictor variables. It works well when most of the features are relevant and should be included in the model.
2. It shrinks the coefficients towards zero, but it does not force them to exactly zero unless lambda (alpha) becomes infinitely large. This means that all features are retained in the model, although some might have very small coefficients.
3. Ridge regression is effective for reducing the complexity of the model and preventing overfitting, especially when dealing with high-dimensional datasets.

Lasso Regression

1. Lasso regression, on the other hand, is preferred when there is a need for feature selection, i.e., identifying the most important predictors while simultaneously fitting the model.
2. It tends to shrink the coefficients towards zero more aggressively compared to Ridge regression and can even force some coefficients to exactly zero.
3. Lasso regression is particularly useful when dealing with high-dimensional datasets with many features, as it can automatically perform variable selection by eliminating irrelevant or less important features.

Here Lasso regression is more applicable since the number of features are very high and we need to find some of the most important features in the model.

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:

Five most important predictor variables before removal are:

1. **GrLivArea:** Above grade (ground) living area square feet
2. **OverallQual_9:** Overall material and finish of the house is "Excellent (9)" or not
3. **OverallQual_10:** Overall material and finish of the house is "Very Excellent (10)" or not
4. **GarageCars:** Size of garage in car capacity
5. **Neighborhood_NoRidge**: Whether the house is in Northridge or not

Five most important predictor variables after removal of top 5 predictor variable are:

1. **1stFlrSF**: First Floor square feet
2. **2ndFlrSF:** Second floor square feet
3. **GarageArea:** Size of garage in square feet
4. **TotRmsAbvGrd:** Total rooms above grade (does not include bathrooms)
5. **Neighborhood_NridgHt**: Whether the house is in Northridge Heights or not

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 that a model is robust and generalizable involves several key steps and considerations:

1. **Train-Test Split or Cross-Validation:** Splitting the dataset into training and testing sets, or performing cross-validation, helps evaluate the model's performance on unseen data. This ensures that the model's performance is not overly dependent on the specific data it was trained on and provides an estimate of its generalization ability.

2. **Regularization:** Regularization techniques like Ridge and Lasso regression help prevent overfitting by penalizing overly complex models. Regularization encourages simpler models that are less likely to overfit the training data, thus improving generalization to unseen data.

3. **Feature Engineering and Selection:** Careful selection and engineering of features can improve the model's ability to generalize. Removing irrelevant features and transforming variables appropriately can reduce noise in the data and enhance the model's performance on new data.

4. **Model Evaluation Metrics:** Using appropriate evaluation metrics is crucial for assessing a model's performance. Metrics like accuracy, precision, recall, F1-score, and ROC-AUC are commonly used, depending on the nature of the problem (classification or regression) and the balance between false positives and false negatives.

5. **Hyperparameter Tuning:** Optimizing hyperparameters through techniques like grid search or random search can help find the best configuration for the model, ensuring better generalization performance.

6. **Cross-Validation Strategies:** Employing robust cross-validation strategies, such as k-fold cross-validation, helps provide a more reliable estimate of the model's performance across different subsets of the data, reducing the risk of overfitting to a specific split.

Implications for Model Accuracy:

Generally, a model that is more robust and generalizable might sacrifice a bit of accuracy on the training data compared to a model that is highly tuned to fit the training data perfectly. This is because robust models prioritize avoiding overfitting, which can result in slightly lower accuracy on the training set. However, the goal of building a model is not to maximize accuracy on the training data but to generalize well to new, unseen data. A model that is robust and generalizable is more likely to perform well in real-world scenarios, where the data may differ from the training set. Therefore, it's important to prioritize generalization over maximizing training set accuracy.