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 1

Optimal value of lambda for Ridge Regression = 10

Optimal value of lambda for Lasso = 0.001

If we increase the alpha value by doubling it, we are increasing the strength of the regularization:

1. For Ridge Regression (alpha = 20): Doubling the alpha value will cause the coefficients to shrink more towards zero compared to the optimal value. This will lead to a model that potentially has higher bias and lower variance, making it less complex and reducing overfitting. However, the coefficients will never be exactly zero; they will just become increasingly smaller as alpha increases.

2. For Lasso Regression (alpha = 0.002) Lasso regression not only penalizes the magnitude of the coefficients but can also drive some of them all the way to zero, performing feature selection. Doubling the alpha value would further compress the coefficients towards zero and may result in additional coefficients being set to zero. This can make the model more interpretable by identifying a smaller subset of predictor variables but can also introduce more bias if relevant predictors are shrunk excessively or discarded.

In terms of the most important predictor variables after the change is implemented, for lasso regression, it would still be the variables with non-zero coefficients, but there may be fewer of them compared to the optimal lasso model with the original alpha value. For ridge regression, the variables with the largest coefficients (farthest from zero) would still be considered the most important, even though all coefficients are shrunken.

It's important to note that changing alpha should be done thoughtfully, as doubling the value arbitrarily without a proper model re-evaluation might lead to worse predictive performance. The real impact can only be assessed by applying the new alpha value and evaluating the model's performance, often done through techniques like cross-validation. The regressions should be rerun, new models evaluated, and the importance of the predictor variables reassessed. This ensures that decisions are data-driven and that the new model continues to be aligned with the company's predictive goals.

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:2 The model we will choose to apply will depend on the use case.

1. Lasso Regression is suitable when we are dealing with numerous features and we suspect that only a few of them actually influence the output variable. Given that Lasso can zero out the coefficients of less important features, it is a valuable technique for feature selection as well as for creating simpler and more interpretable models.

2. Ridge Regression is beneficial when we have correlated features, or we are otherwise concerned about overfitting, and we want to regularize the coefficients (i.e., keep them relatively small) to ensure the model's robustness and generalizability. Unlike Lasso, Ridge does not perform feature selection; it will shrink the coefficients but not set any of them to zero.

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 After dropping our top 5 lasso predictors, we get the following new top 5 predictors:-

1. **2ndFlrSF**
2. **Functional_Typ**
3. **1stFlrSF**
4. **MSSubClass_70**
5. **Neighborhood_Somerst**

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: . Here are several steps and considerations to ensure that the model is robust and generalizable:

1. **Quality of the Data**: Make sure that the data used for training the model is high quality, relevant, and representative of the problem space. It should contain all the important features that influence the target variable, and it is free from errors or outliers that might unduly influence the model.

2. **Feature Selection**: Include features that have a strong and justifiable relationship with the output variable, while avoiding multicollinearity as it can distort the model's performance. You can use techniques like Variance Inflation Factor (VIF) to detect multicollinearity.

3. **Split the Data**: Divide your dataset into training and testing sets to validate the performance of the model on unseen data. A common split might be 70% for training and 30% for testing. You may also want to consider a validation set or employ cross-validation techniques for better generalization.

4. **Cross-Validation**: Implement k-fold cross-validation to ensure that the model’s performance is consistent across different subsets of the data. This will help in reducing the model’s variance and making it more generalizable.

5. **Resampling Methods**: Use resampling methods like bootstrapping to estimate the model accuracy and get more insight into how the model will perform on an independent dataset.

6. **Regularization Techniques**: Apply regularization methods such as Ridge (L2) or Lasso (L1) to prevent overfitting by penalizing large coefficients and thus help the model to generalize better.

7. **Model Complexity**: Check if the model is too complex (overfitting) or too simple (underfitting). Adjusting the complexity can be done by controlling the number of predictors or by tuning hyperparameters in the model.

8. **Diagnostic Plots**: Analyze residual plots, QQ (Quantile-Quantile) plots, and other diagnostic graphs to identify any patterns that suggest the model is not capturing certain aspects of the relationship between variables.

9. **Performance Metrics**: Use appropriate performance metrics like R-squared, Adjusted R-squared, RMSE (Root Mean Square Error), MAE (Mean Absolute Error), etc., and compare these across different models to choose the best fitting one.

10. **Model Updating**: Periodically update the model with new data to ensure that it remains current and to maintain its predictive accuracy over time.

The implications of making a model robust and generalizable are mainly centered around the trade-off between bias and variance, often referred to as the bias-variance tradeoff:

- A model that is too complex might fit the training data very well (low bias) but perform poorly on new, unseen data (high variance). This is known as overfitting.
- A model that is too simple might not perform well even on the training data (high bias) but its performance is consistent across different datasets (low variance). This is known as underfitting.

At the heart of model validation is finding a sweet spot where the model has enough complexity to capture the underlying trends of the data but not so much that it becomes tailored to the idiosyncrasies of the training dataset. This balance ensures that the model will perform consistently in real-world situations (high generalizability) and that your decisions based on its predictions will be reliable and accurate.