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?

Ans: 1

The optimal value of alpha in Ridge and Lasso regression is typically determined through a process called hyperparameter tuning. This involves using techniques such as cross-validation to find the alpha value that results in the best model performance for your specific dataset. The specific optimal value of alpha can vary depending on the dataset and the problem you're trying to solve.

However, as a rule of thumb, smaller values of alpha (close to zero) in both Ridge and Lasso regression result in models that are similar to ordinary linear regression, while larger values of alpha lead to more regularisation.

If you were to double the value of alpha for both Ridge and Lasso regression, the following changes would typically occur:

Ridge Regression:

- Increasing alpha in Ridge regression would increase the amount of L2 regularisation applied to the model.
- This regularisation term encourages the model coefficients to be smaller and more distributed across all predictors.

Lasso Regression:

- Increasing alpha in Lasso regression would increase the amount of L1 regularisation applied to the model.
- L1 regularisation has a sparsity-inducing effect, meaning it tends to force many coefficients to become exactly zero.
- As alpha increases, more predictors will be "zeroed out" and excluded from the model.

After doubling the alpha the variables that are not zero out remain, important for prediction.

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?

Ans 2

During the analysis I got almost the same R squared and RMSE scores for both Train and Test data. On train data I got around 95% R Squared and for test data it was around 88% in both cases of Ridge and Lasso. Lasso tends to do well if there are a small number of significant parameters and the others are close to zero. Here also I found many variable’s coefficients were zero while applying Lasso regression. So I would choose to apply Lasso compared to Ridge.

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?

Ans 3:

To determine the five most important predictor variables in a Lasso model, follow the below steps:

Train the Lasso Model:
First, train your Lasso regression model using the original dataset, including all available predictor variables.

Obtain Coefficient Values:
Retrieve the coefficient values assigned to each predictor variable by the Lasso model. These coefficient values indicate the importance of each variable in making predictions.

Sort Coefficients:
Sort the coefficient values in descending order, so that the most important variables have the largest coefficients.

Select the Top Five:
Select the top five predictor variables with the largest absolute coefficient values as the five most important predictor variables.

So after sorting the variables in descending order of their coefficients, the top five will be the most predictor variables.

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?

Ans 4:

To ensure that the model is robust and generalisable consider the following steps to follow:

Data Splitting:
Use a proper data splitting strategy into training, validation, and test sets. Typically, you'll use a majority of your data for training, a portion for validation to tune hyperparameters, and a separate portion for testing to evaluate final model performance.

Cross-Validation:
Employ techniques like k-fold cross-validation to assess the model's performance more reliably.

Feature Engineering:
Carefully select and engineer features. Feature engineering can significantly impact model performance. Remove irrelevant features, handle missing data appropriately, and create meaningful features if needed.

Regularization: Implement regularization techniques like Lasso or Ridge regularization to prevent overfitting.

Hyperparameter Tuning:
Optimise hyperparameters using a validation set or cross-validation.

Regular Monitoring:
Continuously monitor model performance in production. Models can degrade over time due to changing data distributions, so it's essential to re-evaluate and retrain them periodically.

Bias and Fairness:
Ensure that the model doesn't exhibit bias or unfair behaviour towards specific groups or demographics. Evaluate and mitigate bias in predictions to make the model more generalizable across different subpopulations.

Interpretability:
Choose models that are interpretable and provide insights into their decision-making process.

Transfer Learning:
Consider using pre-trained models or transfer learning when applicable. These models have been trained on large, diverse datasets and can often be fine-tuned for specific tasks, improving generalisation.

Implications for Model Accuracy:

Striving for robustness and generalizability may sometimes lead to a trade-off with model accuracy. When you focus on robustness, you might introduce regularisation or conservative features that could slightly reduce training accuracy.

However, the goal is to achieve a balance between accuracy and generalisation. A model that is overly complex and fits the training data perfectly may suffer from overfitting and perform poorly on new, unseen data. By ensuring robustness and generalizability, you aim to achieve better performance on unseen data, which is the ultimate objective in most real-world applications.

In some cases, even if training accuracy decreases slightly, the model's real-world performance may improve significantly. This is because a more robust model is less sensitive to noise and variations in the data, leading to more reliable and trustworthy predictions.