

What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose to double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

- The optimal values are as follows:
  - For Ridge: 100
  - For Lasso: 0.001
  - We saw that, the training score was less slightly than the testing.
  - Important features:
  - GrLivArea, OverallQual, TotalBsmtSF, OverallCond and YearBuild.
- 

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  $r^2$  score for the lasso is slightly greater than the ridge regression. Hence, we would go with the lasso implementation for the final model. Further, it has the lowest residual squares.
- 

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?

- The top features are GrLivArea, OverallQual, TotalBsmtSF, OverallCond and YearBuild.
- 

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?

Ensuring that a model is robust and generalizable is crucial for its reliability and applicability to new, unseen data. Here are key considerations and practices to achieve robustness and generalizability in machine learning models:

### Data Quality

- Ensure the training data is representative of the problem space.
- Address issues such as missing data, outliers, and data imbalances.
- Use cross-validation to assess model performance across different subsets of the data.

### Feature Engineering

- Select relevant features that contribute to the model's performance.
- Avoid overfitting by limiting the complexity of the model and the number of features.

### Model Complexity

- Choose a model complexity that balances underfitting and overfitting. This can be achieved through techniques like regularization.
- Avoid overly complex models that may memorize the training data but fail to generalize.

### Hyperparameter Tuning

- Optimize hyperparameters using techniques like grid search or randomized search.
- Use cross-validation to find hyperparameter settings that generalize well.

### Cross-Validation

- Employ techniques such as k-fold cross-validation to assess model performance on different subsets of the data.

- Ensure that the model consistently performs well across multiple folds.

### Data Augmentation

- Augment training data by applying transformations like rotation, scaling, or cropping.
- This helps the model learn robust features that are invariant to certain variations.

### Regularization

- Apply regularization techniques (e.g., L1 or L2 regularization) to prevent the model from becoming too sensitive to small variations in the training data.

### Ensemble Methods

- Use ensemble methods, such as bagging or boosting, to combine predictions from multiple models. This can improve robustness by reducing the impact of individual model weaknesses.

## Implications for Accuracy

### Generalizability and Accuracy

- A model that generalizes well to new, unseen data is likely to have higher accuracy on real-world tasks.
- Overly complex models may achieve high accuracy on the training data but might fail to generalize, resulting in poor accuracy on new data.

### Robustness and Stability

- Robust models are less sensitive to variations in the input data, leading to more stable and reliable predictions.
- Robustness contributes to model stability over time, ensuring that the model performs consistently in different environments.

### Avoiding Overfitting:

- Overfit models may achieve high accuracy on the training set but perform poorly on new data. Robustness helps in avoiding overfitting and ensures that the model captures genuine patterns rather than noise in the training data.

### Real-world Applicability:

- Models that are robust and generalizable are more likely to perform well in real-world scenarios, where the data may vary from the training set.

### Adaptability to Changes:

- Robust models can adapt to changes in the data distribution, making them suitable for dynamic environments.

In summary, the pursuit of model accuracy should be accompanied by a focus on robustness and generalizability. Balancing model complexity, optimizing hyperparameters, and employing cross-validation are essential steps to achieve accurate and reliable machine learning models.