Holly Cohan,Yesh Onipede, Yu Xia

Week 6 Report: Model Training, Hyperparameter Tuning, and Model Evaluation

The objective of this week’s assignment is to execute one modeling method with three different hyperparameter settings of the method. We have imputed missing values and conducted standardization and PCA, which is very useful for dealing with high-dimensional data. For this week's iteration, we imputed missing values for validation and testing data, which we avoided last week due to concerns about data leakage. Since the missing values are based on calculated Uber fare breakdowns, the nature of the data means this shouldn’t pose a problem now.

Ordinary Linear Regression

The first modeling method we attempted on our data is ordinary linear regression. The OLS regression model performed poorly, with extremely high error metrics (MSE and RMSE) and a negative R^2, indicating that the model does not explain the variance in the data. A negative R^2 suggests that the model fits the data worse than a horizontal line (no relationship between the features and target variable).

- MSE: 5.561 x 1024
- RMSE: 2.36 x 10^{12}
- R^2: -5.24 x 10^{20}

Ordinary Linear Regression using PCA

In contrast, the OLS model using PCA components shows much better performance. The significantly lower MSE and RMSE, coupled with an R^2 of 0.245, suggest that the PCA-transformed model explains about 24.5% of the variance in the data. While this isn't a perfect fit, it is a marked improvement over the basic OLS model. However, the loss remains quite high, as the RMSE of 89.49 indicates significant error, especially considering how far off this is from typical Uber and Lyft ride prices.

- MSE: 8009.13
- RMSE: 89.49
- R^2: 0.245

Linear Regression with Lasso Regularization (L1)

- We then trained three lasso regression models with varying regularization strengths (alpha) of 0.001, 0.01, 0.1, 1, 10, and 100. We used 5-fold cross validation to determine the best alpha value, and we used the best alpha of 0.1 to train the model to calculate predictions for both the training and validation sets and calculated mean squared error and R-squared.
- By adding an L1 penalty, it shrinks some coefficients to zero, effectively removing irrelevant features from the model. This helps to reduce overfitting and simplify the model without needing to manually reduce the feature set. In this case, Lasso performed reasonably well, though the

errors (RMSE: 89.49) indicate room for improvement. The model explains 21.43% of the variance on the training set and 24.5% on the test set.

  - Training Metrics:
    - MSE: 6871.08
    - RMSE: 82.89
    - R^2: 0.2143
  - Testing Metrics:
    - MSE: 8008.44
    - RMSE: 89.49
    - R^2: 0.2450

Linear Regression with Ridge Regularization (L2)

- We then trained a Ridge regression model, exploring various regularization strengths by adjusting the alpha parameter. We used alpha values of 0.001, 0.01, 0.1, 1, 10, and 100 and performed 5-fold cross-validation to identify the best alpha value. The optimal alpha was found to be 0.1, and we used it to train the model and calculate predictions for both the training and validation sets. We then computed the mean squared error (MSE) and R-squared values to evaluate the model's performance.
- Ridge regression adds an L2 penalty, which shrinks coefficients to prevent large values but does not reduce any of them to zero, unlike Lasso regression. This regularization helps mitigate overfitting and improves model generalization by stabilizing the coefficients. While Ridge regression also simplifies the model by controlling the size of the coefficients, it retains all features in the model, unlike Lasso, which removes irrelevant ones.
  - Training Metrics:
    - MSE: 6866.59
    - RMSE: 82.86
    - R^2: 0.2148
  - Testing Metrics:
    - MSE: 8013.63
    - RMSE: 89.52
    - R^2: 0.2446

Generating polynomial features with no regularization

- We experimented with generating polynomial features of degree 2 and trained a linear regression model without regularization. The Polynomial Regression model performed extremely well on the training set, with an MSE of 2.73, RMSE of 1.65, and an R^2 of 0.9997, indicating near-perfect performance. However, the validation set performed poorly, with an MSE of 4.97 x 10^{22}, RMSE of 2.23 x 10^{11}, and a drastically negative R^2 of -6.10 x 10^{18}, clearly showing severe overfitting due to the model's inability to generalize.
  - Training Metrics:
    - MSE: 6866.5889
    - RMSE: 82.8649
    - R^2: 0.2148

    - Testing Metrics:
        - MSE: 8013.6327
        - RMSE: 89.5189
        - R^2: 0.2446

Polynomial features with Ridge Regression

- To address this overfitting, we introduced Ridge regression combined with polynomial features and experimented with different regularization strengths. The best alpha value was found to be 0.1. The Ridge Regression model with this alpha value performed excellently on both the training and validation sets. For the training set, the MSE was 2.73, RMSE was 1.65, and R^2 was 0.9997, while the validation set showed strong results with an MSE of 9.16, RMSE of 3.03, and R^2 of 0.9989. This demonstrated that Ridge regression effectively regularized the model, reducing overfitting and significantly improving validation performance. We did not experiment with Lasso regression due to its computational complexity when dealing with high-dimensional polynomial features.
    - Training Metrics:
        - MSE: 2.7311
        - RMSE: 1.6526,
        - R^2: 0.9997
    - Validation Metrics:
        - MSE: 9.1598
        - RMSE: 3.0265
        - R^2: 0.9989

Conclusion

In high-dimensional data scenarios, linear regression models without techniques like PCA (Principal Component Analysis) or regularization tend to struggle due to the curse of dimensionality. As the number of features increases, models can become overly complex, leading to overfitting. This means that while the model may perform exceptionally well on training data, it often fails to generalize to validation or test data, resulting in poor predictive performance and inflated error metrics.

L1 (Lasso) and L2 (Ridge) regularization methods are critical in addressing these challenges. Incorporating these regularization techniques into models allows for better handling of high-dimensional data by controlling complexity and enhancing generalization. In addition, adding polynomial terms significantly reduced loss, highlighting that a simple linear model is insufficient for capturing the complexities inherent in the data. The introduction of polynomial features allowed the models to better fit the underlying patterns and relationships, which linear regression alone could not adequately represent. This underscores the need for more flexible modeling approaches when dealing with non-linear relationships for our datasets.