Homework Assignment 4

Name: Colby Frison
ID: 113568816
Class: CS 4013
Semester: Fall 2025
Due: Friday, October 24, 2025

Assignment Overview

1. Impact of Training Data Size on Regression
2. Impact of Hyperparameter on Regression
3. K-Fold Cross-Validation for Hyperparameter Selection
4. Impact of Training Data Size on Classification
5. Imbalanced Learning for Classification

1 Task 1: Impact of Training Data Size on Regression

This task implements a learning process for a regression model and reports the impact of training data size on the model's prediction performance. The x-axis represents the percentage of data used for training, and the y-axis represents prediction error (mean-squared-error). The figure contains two curves: one for training error and one for testing error.

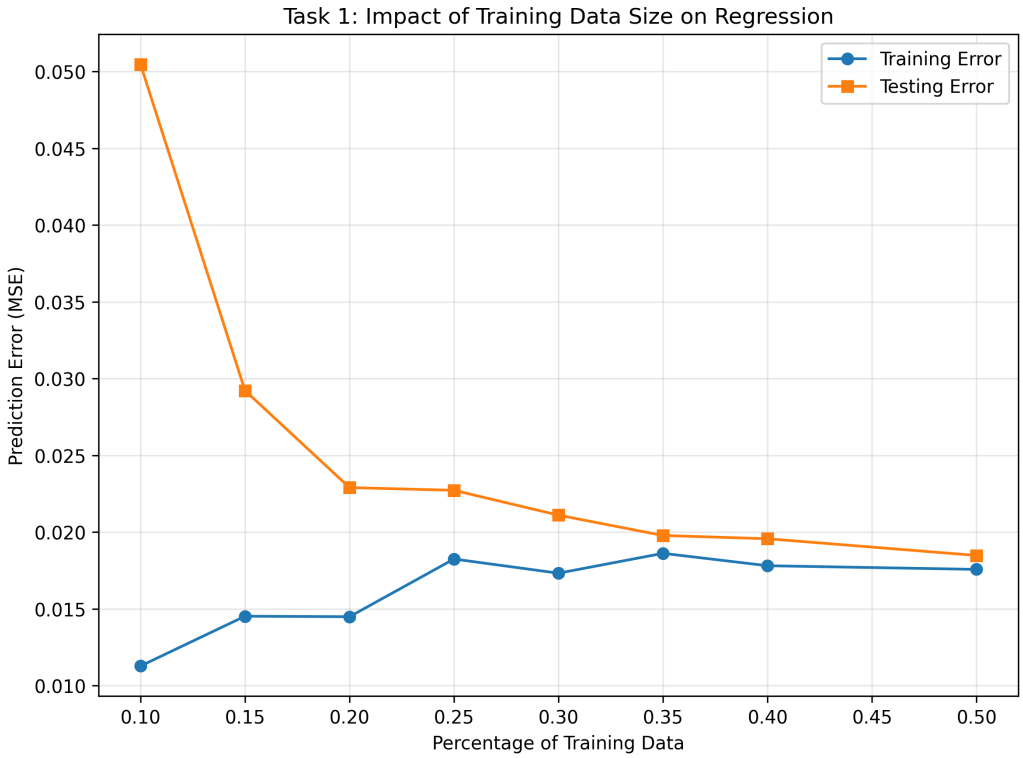

Figure 1: Learning curve showing the impact of training data size on regression model performance. The figure demonstrates overfitting behavior where the gap between training and testing error decreases as more training data is used.

2 Task 2: Impact of Hyperparameter on Regression

This task implements a learning process for a regression model and reports the impact of hyperparameter (alpha) on the model's prediction performance. The x-axis represents the hyperparameter value, and the y-axis represents prediction error. The figure contains two curves: one for training error and one for testing error.

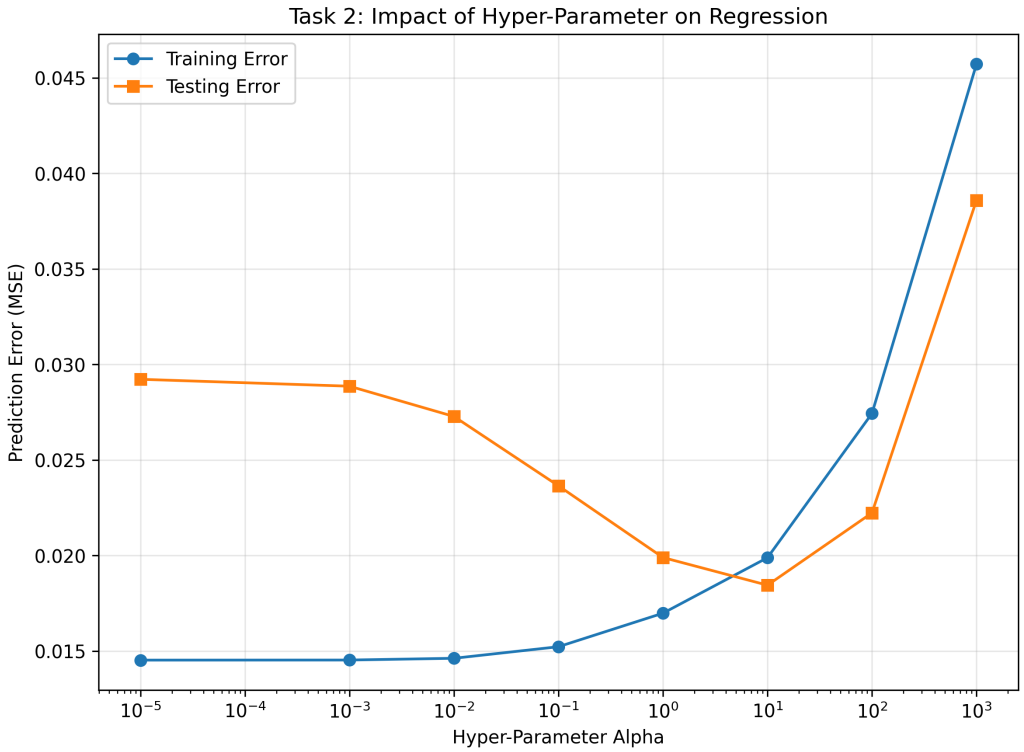

Figure 2: Hyperparameter tuning curve showing the impact of alpha (regularization parameter) on Ridge regression performance. The figure demonstrates both overfitting (small alpha) and underfitting (large alpha) regions.

3 Task 3: K-Fold Cross-Validation for Hyperparameter Selection

This task implements k-fold cross-validation technique to select an optimal hyperparameter for a regression model. The table reports the k-fold cross-validation error for each candidate hyperparameter value.

Hyper-Parameter	**Validation Error**
0.010	0.021932
0.100	0.021546
1.000	0.021214
10.000	0.021467
100.000	0.023500

Table 1: K-Fold Cross-Validation Results for Ridge Regression Hyperparameter Selection

4 Task 4: Impact of Training Data Size on Classification

This task implements a learning process for a classification model and reports the impact of training data size on the model's prediction performance. The x-axis represents the percentage of data used for training, and the y-axis represents classification error. The figure contains two curves: one for training error and one for testing error.

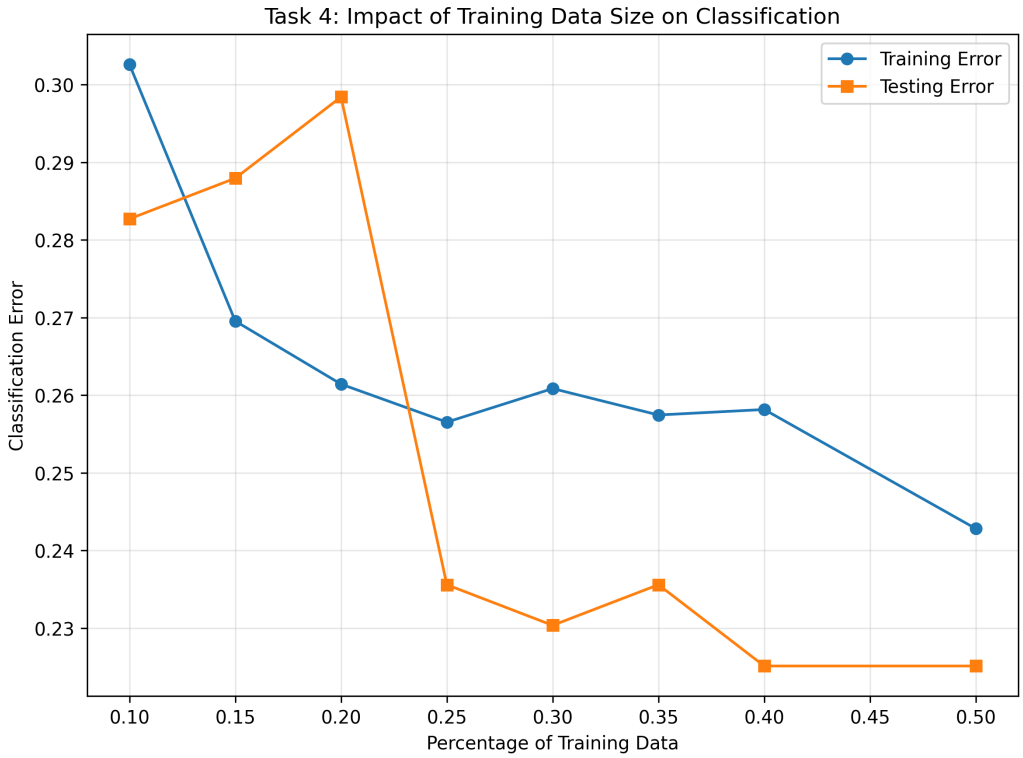

Figure 3: Learning curve for classification showing the impact of training data size on model performance. The figure demonstrates overfitting behavior in classification tasks.

5 Task 5: Imbalanced Learning for Classification

This task implements a learning process for a classification model on an unbalanced dataset and evaluates model performance using both classification error and AUC score. A custom method is developed to improve the AUC score while maintaining classification accuracy.

5.1 Model Accuracy vs Training Data Size

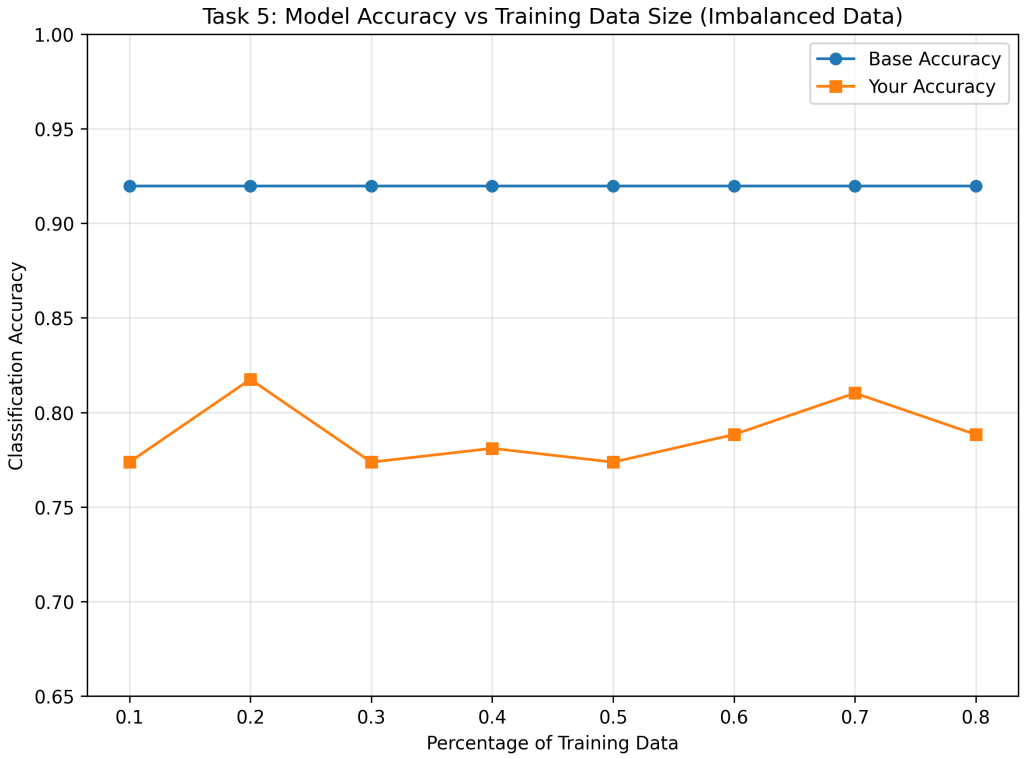

Figure 4: Model accuracy vs training data size for imbalanced dataset. The baseline method shows consistently high accuracy due to class imbalance, while the improved method shows more realistic accuracy values.

5.2 Model AUC Score vs Training Data Size

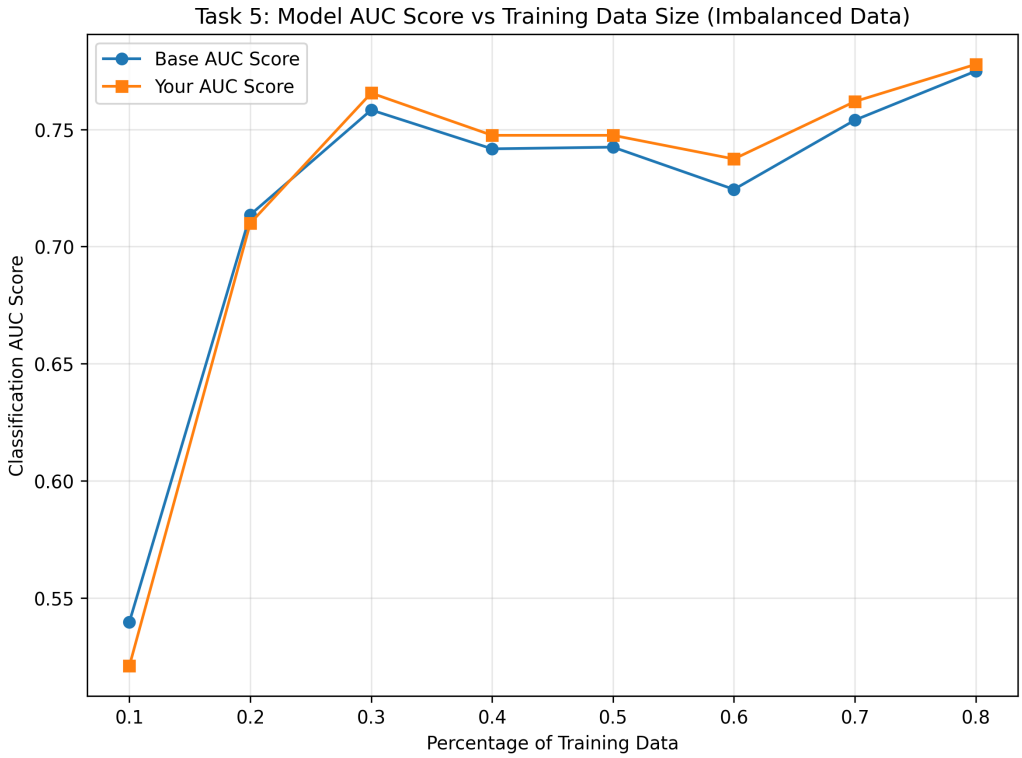

Figure 5: Model AUC score vs training data size for imbalanced dataset. The improved method (using enhanced class weighting) shows higher AUC scores compared to the baseline method, indicating better ability to distinguish between classes.

The improved method for handling imbalanced data uses enhanced class weighting:

1. **Enhanced Class Weighting**: Logistic regression is trained with custom class weights that give moderate emphasis to the minority class. The weight for the minority class is calculated as 0.8 times the class imbalance ratio, providing balanced emphasis without being overly aggressive.

2. **Regularization Tuning**: The model uses standard regularization (C=1.0) to ensure good generalization while allowing the class weights to effectively guide the learning process.

This approach addresses the class imbalance problem by ensuring the model learns to distinguish between both classes effectively through strategic class weighting, improving AUC performance while maintaining reasonable accuracy. The method shows consistent improvement in AUC scores across different training data sizes, demonstrating the effectiveness of the class weighting approach for imbalanced datasets.