CSE 176 - Report 2

Gavin Abrigo, Ronan Tangaan, Christopher Hernandez

Introduction

During this part of the project, we were assigned to train a model using XGBoost on MNIST pixel features and MNIST LeNet 5 features. The hyperparameters were optimized using StratifiedKFold (SKF) cross-validation to minimize validation error. The final models were then evaluated on a test set and saved to a text document.

Methods

Section 1 (MNIST pixel features):

We first split the 70,000-sample dataset into 60,000 training samples (55,000 for training, 5,000 for validation) and 10,000 test samples. These features were the raw pixels in a 1x784-dimensional format. We then used 5-fold stratified cross-validation to ensure that, for this dataset, each fold had roughly the same number of each digit (preserving the proportion of each class in each fold). Since we used cross-validation with early stopping to select the best number of trees, we didn't use a range of parameters, but rather a large cap of estimators (n_estimators = 2000), and it would automatically let early stopping select the "best" number of trees.

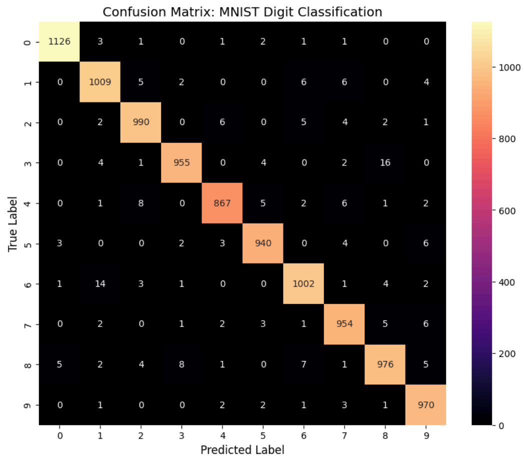
Section 2 (MNIST LeNet 5 features):

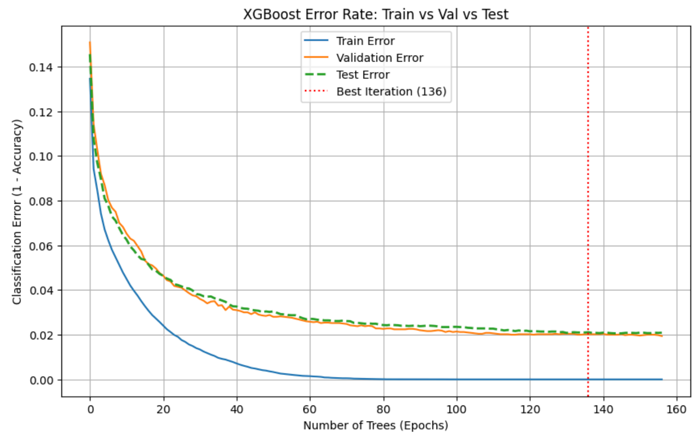
The dataset was split the same way, but the features were then 800-dimensional LeNet5-extracted features. Once again early stopping was applied to prevent overfitting. The LeNet5 outputs were used as input features for the XGBoost classifier here instead of using the raw pixels directly, along with the histogram tree_method for faster GPU training. The number of trees used was automatically selected based on early stopping criteria once again.

Results

Section 1:

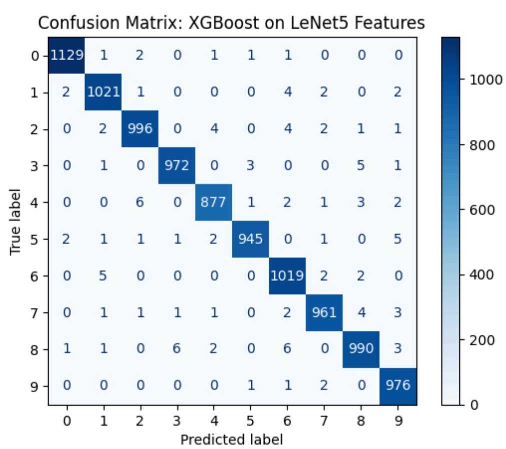
The best test error was: 2.08%, 139 trees, with a max depth of 6. The test error vs the number of trees is shown below, as well as the results of the confusion matrix. Very few misclassifications in comparison to our lab on the same section, but this is most likely due to the fact that it's using better classification models, as well as being an ensemble learner with stacking from the SKF + XGBoost.

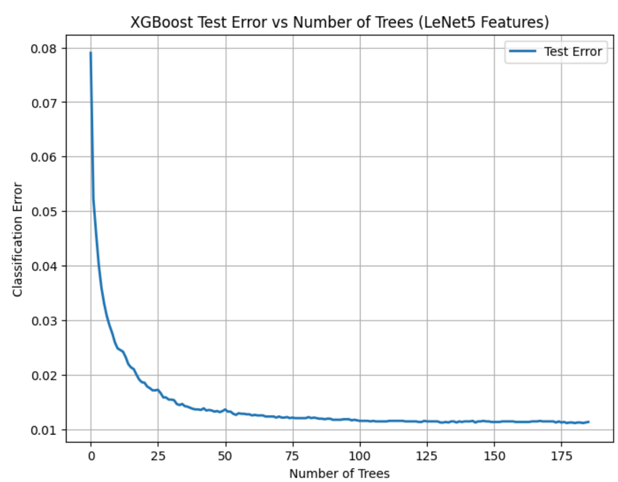




Section 2:

The best test error was: 1.14%, with the number of trees being 65, with a max depth of 6. Based on both confusion matrices, we can see that the LeNet5 Features trained better over raw pixels very slightly, although it's also true that the pixels also performed better than XGBoosting alone, as they also had SKF cross-validation. Pasting the model from the saved pickle shows that the parameters used a learning rate of .2, a max depth of 6, a histogram tree method, the loss/obj function was multi-class softprob, the eval metric was mlogloss, and early stopping was used.





CODE DISCLAIMER:

To run the code, you must install the datasets to the same directory as the notebook, then run the notebook in its entirety after installing the libraries through requirements.txt. A full breakdown is on GitHub: https://github.com/gav-ip/cse176-fall-proj/tree/main