Improving MNIST Accuracy through Hyper Parameter Tuning and Feature Engineering Project Update

Aina Shafqat
2021-EE-058
Department of Electrical Engineering
University of Engineering and Technology
2021ee58@student.uet.edu.pk

Ayesha Ahmad
2021-EE-052
Department of Electrical Engineering
University of Engineering and Technology
2021ee52@student.uet.edu.pk

This progress update outlines the methodology followed, the results achieved thus far and the future direction of the project. This project is focused on optimizing the performance of a classification model, SVM, using hyper parameter tuning and feature engineering techniques on the MNIST dataset. Experiments were conducted to assess the impact of various hyper parameters on model accuracy and training time, and to identify configurations that notably improved performance metrics. Our exploration included utilizing different featurization techniques to enhance the model's discriminative power with MNIST data. Hyper parameter tuning employed the optuna library's Bayesian optimizer, while featurization involved transforming raw MNIST data into meaningful representations.

1 Methodology

1.1 Data Preprocessing

The images in the MNIST dataset were preprocessed through feature engineering. Feature engineering is essential for enhancing the discriminative power of machine learning models. Various featurization techniques were employed to extract meaningful representations from the raw data.
One such technique is the centroid method, which involves identifying the centroid or geometric center of clusters within the data. The number of clusters set were varied from 1 to 20, to generate multiple featurized datasets. Additionally, flattening was utilized to convert the multidimensional image into a one-dimensional array, facilitating easier processing by the SVM model.
Eigenvalues and eigenvectors extracted from covariance matrices of the images were also explored as features, providing insights into the variance and orientation of the data. Furthermore, image thresholding alongside flattening were employed to extract relevant features based on intensity levels. The thresholds were varied from 0 to 1, to generate multiple featurized datasets.

1.2 Model Training

The featurized datasets were classified using a Support Vector Machine (SVM) model as per the restrictions on the MNIST competition on Kaggle. SVMs are well-suited for tasks where the number of features exceeds the number of samples, making them suitable for the MNIST dataset, which consists of images represented as a matrix of pixels.

1.3 Hyperparameter Tuning

Hyper parameter tuning is a critical aspect of optimizing machine learning models for better performance. To achieve this, the optuna library's default optimizer was employed. It leverages a Bayesian optimization algorithm called the Tree-structured Parzen Estimator (TPE). This algorithm efficiently explores the hyper parameter space to identify configurations that yield the best objective

Preprint. Under review.

function value. The validation accuracy was set as the objective function value to be maximized. The maximum number of iterations set for the hyper parameter tuning were 100 trials. This implies that the SVM was trained 100 times with different configurations for one sset of features.
The hyperparameters of the SVM tuned included 'C', 'kernel', 'degree', 'gamma', and 'decision_function', each of which plays a crucial role in determining the behavior and flexibility of the SVM model.

2 Results

The optimized hyper parameters of the featurized datasets are shown in Table 1. The results show that the covariance matrix and its eigenvalues and eigenvectors, are not good features for the classification themselves. The centroid method proved effective in transforming raw data into meaningful representations, thereby contributing to improved classification performance for the training set, with a training set accuracy reaching 99.95%. However, its validation accuracy 78.85% was not as high, likely due to overfitting. The best test set accuracy till now is 97.871%. This accuracy ranks 132 on the Public Leaderboard and 284 on the Private Leaderboard.

Table 1: Best test accuracies of each feature engineering

		Test Accuracy		
Tuning Time (hrs)	Featurizing	Private Score (%)	Public Score (%)	Validation Accuracy (%)
89.5315	Flattened	97.871	97.266	97.91
84.1106	Thresholded and Flattened	94.871	94.533	92.67
19.2589	Covariance Matrix	43.928	44.666	48.61
73.3889	Eigenvalues and vectors of Cov Matrix	42.228	42.533	42.56

3 Future Directions

Future efforts will focus on exploring additional feature extraction techniques, including edge detection, corner detection, Principal Component Analysis (PCA), scale-invariant feature transform (SIFT), speeded-up robust feature (SURF), and histogram of oriented gradients (HOG). Additionally, training SVM models on different combinations of features will be explored to assess their impact on classification accuracy.

4 Conclusion

In conclusion, this project demonstrates the effectiveness of hyper parameter tuning and feature engineering in enhancing machine learning model performance. Through rigorous experimentation and optimization, significant improvements in accuracy metrics were achieved, paving the way for further exploration and optimization in future research endeavors. It's noteworthy that this progress till now will continue with the exploration of different methods to further increase the test set accuracy.