**ALY6020 Predictive Analytics**

Dr. Thomas Goulding

Module 5

Handwriting Classification Analysis

Jeff Hackmeister

6/22/2025

**INTRODUCTION**

The goal of this project is to assist a school in developing a model to identify students who may need additional help in the development of their motor skills. The data provided by the school is handwriting analysis from students who were asked to draw digits from 0 to 9. The observations in the dataset contain 45 different pixel readings from each drawn figure. The task is to build a classification model that can accurately identify which digit (0-9) a student intended to write, with the ultimate goal of identifying students who may benefit from additional motor skill development support.

**DATA EXPORATION**

To begin the project, all necessary packages will be loaded into Jupyter notebook for analysis. A screen shot of a computer program

AI-generated content may be incorrect.

After reading in the csv file provided for the project, a histogram is produced to see the distribution of our handwriting samples.

A graph of blue bars

AI-generated content may be incorrect.

This confirms that there is ample representation for all 10 digits in the dataset.

In order to prepare the data for modeling, the label variable is separated from all pixel data and then an 80/20 split is used to create training and test datasets.

**A screenshot of a computer code

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**MODEL #1 – NEAREST NEIGHBORS**

The first modeling technique used will be nearest neighbors. This is a relatively simple modeling technique but should provide a useful baseline for analysis of additional models to come. A crucial step for this technique is scaling, or normalizing, the data.

**A computer code with black text

AI-generated content may be incorrect.**

Once that is done, a model can be fit using the KNeighborsClassifier from sklearn. Once the model is fit to the training data, the model can be used to make predictions on the test data. K was set for 5 in this initial model

**A black and white text

AI-generated content may be incorrect.**

****

And diagnostic statistics can be generated from the predictions.

**A screenshot of a computer

AI-generated content may be incorrect.**

61.36% accuracy from the model is not indicative of a highly predictive model. The variation between the precision and recall of each digit does provide some useful insights for the study.

To improve the performance of the nearest neighbors model, the data was rescaled, this time using the MinMaxScaler from sklearn, this methodology does not allow for negative values, which should not be included in this study and should help improve the predictive power of the model. Additionally, a series of models was fit, testing K values of 1-15 (only the odd values to avoid dealing with ties in even K values). From this testing, the optimum K value was determined to be k=11.

A screen shot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

This new versions of the nearest neighbors model was fit and again used to make predictions on the test dataset with the following results/

A screenshot of a computer

AI-generated content may be incorrect.

A close up of a white board

AI-generated content may be incorrect.The improved scaling method and optimized K value did lead to a higher accuracy score in this model (66% vs 62%). Of particular interest are the high precision scores of 0, 1 and 4. This may align with those digits being visually distinct from other digits. The low recall for 4 may be a result of different variations in the acceptable drawing of the digit (see example on the right, found on Reddit), while 1 and 0 would likely have fewer acceptable variations. These variations are classic indicators for the need to implement more sophisticated modeling techniques.

**MODEL #2 – NEURAL NETWORKS**

Based on the results from the nearest neighbors model, the decision was made to move towards neural networks methodology. These models are able to process multiple inputs and assign predictive values to each and then calculate a predictive value based on the input from each input. These models are able to process vast amounts of data and produce highly accurate predictive models. While they are sometime referred to as “black box” models since they have fewer illustrative properties (compared to a nearest neighbors or decision tree model), there are several aspects of the model that be finetuned to improve performance.

The first neural network model iteration used the Stochastic Gradient Descent (sgd) solver with 10,000 iterations.

A screenshot of a computer code

AI-generated content may be incorrect.

The resulting model’s accuracy was less that both nearest neighbors models. To improve the results, the value for hidden layers will be increased from 3 to 100.

A screenshot of a computer code

AI-generated content may be incorrect.

This made an immediate difference in the accuracy of the model by increasing the ability of the model to identify and seperate the key features from the data. To attempt to further improve the model, the max iteration is increased to 20,000

A screenshot of a computer program

AI-generated content may be incorrect.

This did lead to an increased accuracy score, but only a subtle improvement. For the next iteration, the adam or Adaptive Movement Estimation solver will be used. Adam tends to handle noisier data better as it uses moving averages, all other inputs will remain the same.

A screenshot of a computer code

AI-generated content may be incorrect.

This model took a step back in accuracy from the previous version. The next step towards model optimization is to introduce a pyramidal structure across the hidden layers .

A blue and black text

AI-generated content may be incorrect.

A number of numbers and a percentage

AI-generated content may be incorrect.

This model comes closer to the sgd, 20,000 iterations model. The next step is to implement Principal Component Analysis or PCA to reduce the dimensionality of the data.

A screenshot of a computer code

AI-generated content may be incorrect.

Through this method, 45 features were compressed into the most influential 30 while only 97.4% of the original variance (only 2.6% information loss). With this newly optimized dataset, a neural network model is once again fit and will be applied to the test data as well.

A screenshot of a computer program

AI-generated content may be incorrect.

This achieved the best result on training data, with 72.69% accuracy and performed very well on the test data as well (69.43%).

A graph of a graph showing a number of blue rectangular objects

AI-generated content may be incorrect.**MODEL COMPARISION**

In looking at the performance of all the models fit to the data, we will focus on the optimized nearest neighbors, the initial neural network and the final, optimized neural network model. As seen in the chart, the optimized nearest neighbor model (KNN) outperformed the original neural network model, but through a series of optimization steps, the final neural network model achieves the highest accuracy.

To further compare the two optimized models, confusion matrices were produced.

A close-up of a diagram

AI-generated content may be incorrect.

From these results, it is notable that both models struggle with digits with similar physical structures such as 7 and 9, and 4 and 9 but the neural network outperformed the KNN model for recognizing these more difficult digits. Overall, the neural network model made 262 more correct predictions, an improvement of 3.1%.

**CONCLUSIONS AND RECOMMENDATIONS**

After several optimization steps for both modeling techniques, neither model achieved greater than 75% accuracy. This indicates that the nuances in handwriting for students is difficult to model and any attempts to use the neural network model for student intervention should be done carefully. Creating a new dataset of students based on their performance in this model and whether or not they receive interventions for motor skill development, followed by another sampling of their handwriting after a period of time (1 school year) would be a worthwhile and interesting continuation of this work.

**REFERENCES**

[1] *Configure MLPClassifier “solver” Parameter | SKLearner*. (2024). Sklearner.com. https://sklearner.com/sklearn-mlpclassifier-solver-parameter/

[2] GeeksforGeeks. (2023, October 11). *Classification Using Sklearn Multilayer Perceptron*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/classification-using-sklearn-multi-layer-perceptron/

[3] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning : Data Mining, Inference, and Prediction, Second Edition*. Springer New York.

[4] OpenAI. (2024). ChatGPT (Claude 4) [Large language model]. <https://claude.ai>

[5] scikit-learn. (2019). *sklearn.preprocessing.MinMaxScaler — scikit-learn 0.22.1 documentation*. Scikit-Learn.org. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

‌[6] Starmer, J. (n.d.). *The Essential Main Ideas of Neural Networks*. YouTube. https://www.youtube.com/watch?v=CqOfi41LfDw

[7] Steele, B., Chandler, J., & Reddy, S. (2016). *Algorithms for Data Science*. Springer.