COMPSCI 589 Final Project

William Cai, Maxwell Tang

9 May 2025

1 MNIST Dataset

2 Rice Dataset

Maxwell's implementations were used for the entirety of this dataset.

For the rice dataset, we chose to use the multilayer perceptron (MLP), k-nearest neighbors (KNN), and random forest (RF) architectures to predict on our dataset. We chose this because decision trees are a subset of random forests and naive bayes is hard to adapt to numerical data.

First, we swept over the secondary hyperparameters in each algorithm. For MLP, we used a training rate of 0.1, a regularization cost of 0.01, and used unbatched gradient descent. As shown in Table 3, you can see that a model shape of [64, 128, 1] achieved the highest accuracy. For RF, we used the set size as a stopping criteria: nodes below the splitting threshold cannot be split. As shown in Table 4, a splitting threshold of 30 achieves the highest accuracy.

As shown in Figure 4, a k value of around 3 or 4 is the best. This indicates that the dataset doesn't have very strongly mixed classes. As shown in Figure 5, all metrics plateau around 100 epochs in. As shown in Figure 6, the metrics rise steeply and then plateau. The exact values for these penomena can be found in the referenced graphs and in the code. The best performing model overall was the MLP with an accuracy of 0.93. This is really close the best performance

Model Shape	Epoch	Accuracy
[64, 128, 10]	989	0.952145
[64, 64, 10]	977	0.933768
[64, 10]	988	0.920987
[64, 16, 10]	991	0.878687
[64, 16, 16, 10]	998	0.803554
[64, 8, 10]	997	0.722818
[64, 8, 8, 10]	999	0.533731

Table 1: The results for the hyperparameter tuning step on the MLP architecture trained on the MNIST dataset. The best epoch is used for each hyperparameter setting.

Minimum Splittable Size	ntree	Mean Accuracy
5	100	0.977182
10	100	0.975515
20	50	0.967722
30	100	0.954926
40	100	0.953256
50	50	0.943811
2	10	0.523063

Table 2: The results for the hyperparameter tuning step on the random forest architecture trained on the MNIST dataset. The best epoch is used for each hyperparameter setting.

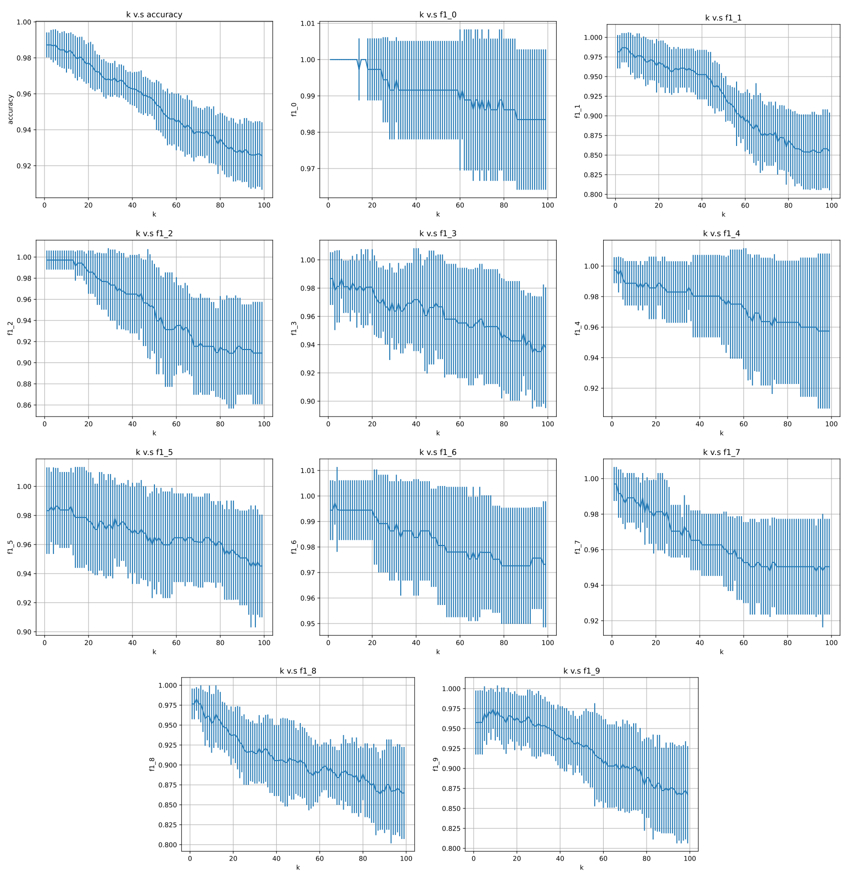

Figure 1: Performance metrics graphed against k for KNN on the MNIST dataset.

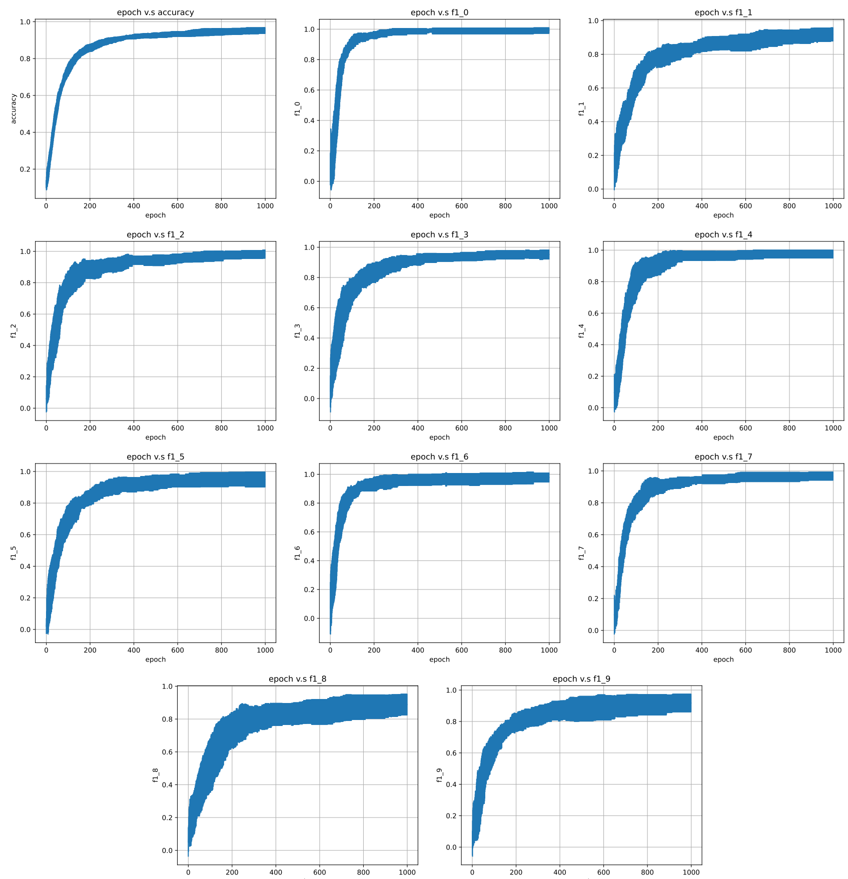

Figure 2: Training graphs for a MLP trained on the MNIST dataset.

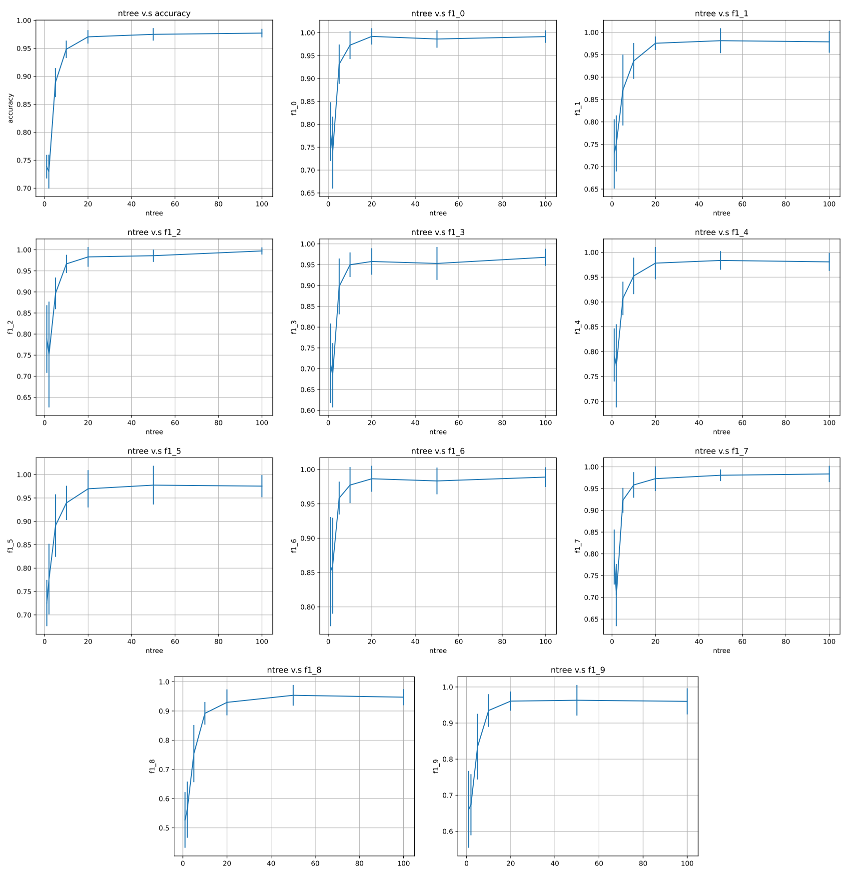

Figure 3: Performance metrics graphed against ntree for random forests trained on the MNIST dataset

Model Shape	Epoch	Mean Accuracy
[7, 8, 8, 1]	765	0.930971
[7, 2, 1]	979	0.929921
[7, 2, 4, 1]	996	0.929659
[7, 1]	670	0.929396
[7, 16, 16, 1]	553	0.928871
[7, 4, 16, 1]	735	0.928871
[7, 2, 16, 1]	656	0.928871
[7, 16, 1]	584	0.928871
[7, 16, 8, 1]	757	0.928084
[7, 8, 2, 1]	969	0.928084
[7, 8, 1]	535	0.928084
[7, 8, 4, 1]	638	0.928084
[7, 16, 2, 1]	760	0.927822
[7, 2, 8, 1]	960	0.927822
[7, 8, 16, 1]	748	0.927822
[7, 16, 4, 1]	732	0.927822
[7, 4, 8, 1]	835	0.927559
[7, 4, 4, 1]	993	0.927297
[7, 4, 1]	781	0.927034
[7, 4, 2, 1]	770	0.925722
[7, 2, 2, 1]	999	0.887139

Table 3: The results for the hyperparameter tuning step on the MLP architecture trained on the rice dataset. The best epoch is used for each hyperparameter setting.

Minimum Splittable Size	ntree	Mean Accuracy
30	100	0.928215
20	100	0.927979
40	100	0.927428
50	100	0.927008
10	50	0.926772
2	50	0.926772
5	50	0.926772

Table 4: The results for the hyperparameter tuning step on the Random Forest architecture trained on the rice dataset. The best ntree value is used for each hyperparameter setting.

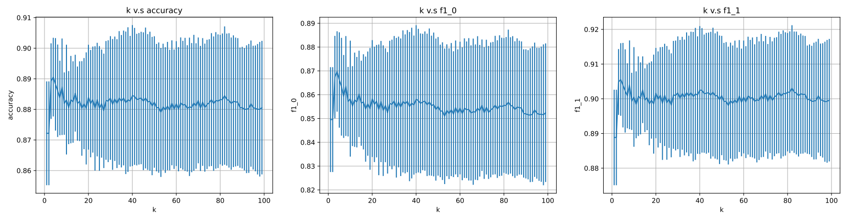

Figure 4: Performance metrics graphed against k for KNN on the rice dataset.

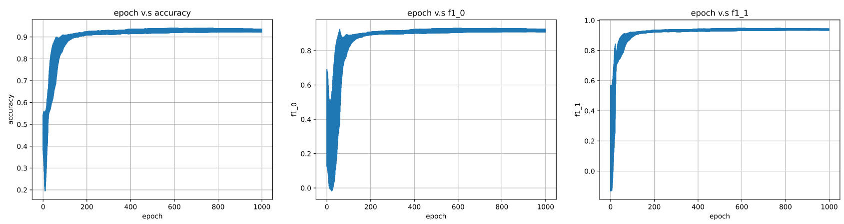

Figure 5: Training graphs for a MLP trained on the rice dataset.

of random forests, which was 0.928, but this was significantly higher than the best performace of the KNN architecture, which was 0.89. This might indicate that the dataset has differently behaved regions, where some areas have more outliers than others, making KNN models dificult to fit to the dataset.

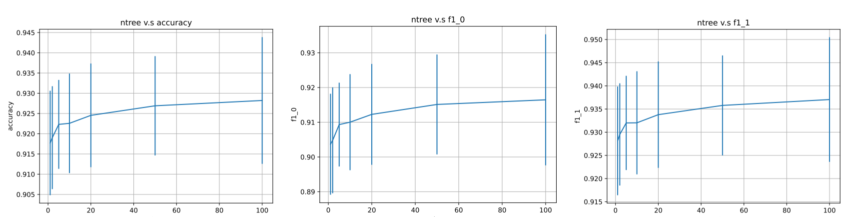

Figure 6: Performance metrics graphed against ntree for random forests trained on the rice dataset