HDS 5230: High Performance Computing

Week 12 - Neural Network

Harika Pamulapati

Professor: Adam Doyle

1. Using the data synthesis R script provided by the instructor as part of the week 11 assignment instructions, produce datasets of the following sizes, and fit deep learning models with the configurations shown below. Associated with each model, record the following performance characteristics: training error, validation (i.e., holdout set) error, time of execution. Use an appropriate activation function.

Data size	Configuration	Training error	Validation error	Time of execution
1000	1 hidden layer 4 nodes	0.292500	0.29500	3.40
10000	1 hidden layer 4 nodes	0.132500	0.13000	2.72
100000	1 hidden layer 4 nodes	0.011250	0.01100	7.64
1000	2 hidden layers of 4 nodes each	0.004250	0.00400	7.54
10000	2 hidden layers of 4 nodes each	0.001713	0.00160	62.04
100000	2 hidden layers of 4 nodes each	0.001900	0.00245	65.03

2. Based on the results, which model do you consider as superior, among the deep learning models fit?

The model comprising 2 hidden layers with 4 nodes in each performed best with 10,000 training data points. The model configuration with two hidden layers of 4 nodes each demonstrates the best validation performance (0.00160) and maintains a small difference between training (0.001713) and validation errors which indicates strong generalization capabilities without overfitting. The 100,000 data point model with identical architecture shows similar training error

(0.001900) yet its validation error (0.00245) indicates overfitting occurs. The 10,000 data model runs at 62.04 seconds which shows a reasonable execution time when compared to the minimal time improvement from using 100,000 data points (65.03 seconds). The developed model optimizes accuracy and generalization ability together with computational speed.

3. Next, report the results (for the particular numbers of observations) from applying xgboost (week 11 – provide the relevant results here in a table). Comparing the results from XGBoost and deep learning models fit, which model would you say is superior to others? What is the basis for your judgment?

Method used	**Dataset size**	**Testing-set predictive performance**	**Time taken for the model to be fit**
XGBoost in Python via scikit-learn and 5-fold CV	1000	0.9470	2.81
	10000	0.9750	1.05
	100000	0.9869	4.15

XGBoost delivers superior results than deep learning models at every dataset size according to the model performance metrics. The predictive accuracy of XGBoost reaches 0.9470 for the 1,000 data points while surpassing the best deep learning model's validation error of 0.00400. The XGBoost model achieves 0.9750 predictive performance for 10,000 data points while deep learning reaches only 0.00160 validation error and XGBoost reaches 0.9869 for 100,000 data points but deep learning achieves 0.00245 validation error. The XGBoost model executes data with significantly faster speeds than deep learning models do because it takes 1.05 seconds to process 10,000 data points while deep learning requires 62.04 seconds.

The assessment of XGBoost superiority combines its strong predictive capabilities with its optimized computational performance. The predictive performance of XGBoost approaches 1 in

accuracy while deep learning shows 0 error rates but XGBoost uses less computational time to achieve these results. XGBoost includes 5-fold cross-validation as a built-in feature that produces more reliable performance metrics than the single train-validation split used by deep learning models. The combination of XGBoost trees through ensemble learning uses less computing power to identify hidden data patterns in this particular dataset resulting in better outcomes over all tested dataset scales.