# Week 12 Assignment – Deep Learning vs. XGBoost Comparison

## A performance assessment of deep learning and XGBoost models with various configurations takes place through PIMA Diabetes dataset-based classification tasks. The deep learning models contained 1 or 2 hidden layers with 4 nodes in each layer and XGBoost results were obtained from Week 11 analysis. The evaluation measures performance through different dataset sizes by analyzing training error and validation error and execution time.

## Deep Learning Results Summary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset Size | Model Type | Train Error | Val Error | Time (s) |
| 1000 | 1 hidden layer | 0.0125 | 0.01 | 3.72 |
| 1000 | 2 hidden layers | 0.0000 | 0.00 | 4.13 |
| 10000 | 1 hidden layer | 0.0000 | 0.00 | 7.70 |
| 10000 | 2 hidden layers | 0.0000 | 0.00 | 11.37 |
| 100000 | 1 hidden layer | 0.0000 | 0.00 | 75.84 |
| 100000 | 2 hidden layers | 0.0000 | 0.00 | 78.56 |

## Summary and Recommendation

Deep learning models which adopted two hidden layers demonstrated superior validation error compared to those with one hidden layer especially when working with smaller datasets. The performance gap between these architectures decreases when working with larger datasets because they both demonstrate near-perfect accuracy on the biggest datasets.

The execution time of models with two hidden layers proved to be longer than the time needed for one-layer models during training sessions. The time needed for training extends as the dataset size increases.

The PIMA structured tabular data shows that Python XGBoost achieves better results coupled with significantly accelerated training duration than the XGBoost model from Week 11. XGBoost stands as the better selection for structured classification problems even though deep learning maintains value when used with complex feature types or unstructured data.

The recommended approach for structured data tasks requiring speed and efficiency is to use Python XGBoost. Deep learning algorithms should be limited to analyzing unstructured data sets (for example images or texts) as well as situations that require feature interactions which tree-based approaches cannot easily model.