1.

Data size	Configuration	Training error	Validation error	Time of execution
1000	1 hidden layer 4 nodes	0.238	0.2429	0.071
10000	1 hidden layer 4 nodes	0.0128	0.0142	0.495
100000	1 hidden layer 4 nodes	0.0006	0.0006	6.255
1000	2 hidden layers of 4 nodes each	0.246	0.2383	0.065
10000	2 hidden layers of 4 nodes each	0.0012	0.0021	1.149
100000	2 hidden layers of 4 nodes each	0.0007	0.0007	6.365

2.

The single hidden layer model with 4 nodes trained using 100,000 data points demonstrates the best performance according to the results. The model with one hidden layer and 4 nodes demonstrates the best generalization ability by achieving the smallest validation error rate of 0.0006. The single-layer model demonstrates equivalent training error (0.0006) compared to the two-layer model when trained on the same dataset but avoids overfitting because its training and validation errors remain consistent. The single-layer network trains faster than the two-layer network because it requires 6.255 seconds to complete training while the deeper model takes 6.365 seconds. The single-layer model with 100,000 data points proves superior results alongside

better computational efficiency when trained with sufficient data even though the two-layer model with 10,000 data points performs well.

3.

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	0.97	0.65
	10000	0.9820	2.03
	100000	0.9866	4.06

XGBoost outperforms both deep learning models and all other metrics in its evaluation results. XGBoost reaches a testing-set predictive performance of 0.9866 (which seems to represent accuracy or R^2 score) in 4.06 seconds using the 100,000 dataset while the best deep learning model (single hidden layer with 4 nodes) requires 6.255 seconds to reach a validation error of 0.0006. XGBoost delivers outstanding performance at 0.97 with 1,000 samples and 0.982 with 10,000 samples while training at a much faster rate. XGBoost demonstrates superiority through four key factors: (1) it maintains high predictive accuracy across all dataset sizes, (2) it trains faster than other models, (3) it performs well with limited data by reaching 0.97 accuracy with 1,000 samples while the deep learning model achieves only 0.2429 validation error and (4) its inherent interpretability surpasses neural networks. XGBoost shows greater efficiency and better effectiveness in resolving this specific problem compared to deep learning methods.