3. Section 3: Machine Learning

3.1 Question 1

i Length and width

Firstly, the 15,974 missing values across the 111 features in this data set had to be effectively dealt with. Following Mishra and Khare (2014) columns with more than 50% of data missing were dropped and the remaining 107 were imputed. Iterative imputation was favoured since the data is most likely missing at random (Rubin, 1976). A random forest regressor was then used to iteratively impute values, since it effectively deals with different data types and captures non-linear patterns for the patient attributes under consideration. All values were rounded to the nearest integers with binary variables rounded to 0 or 1. Finally, categorical variables were clipped to their original range.

A simple feedforward neural network was trained on mini batches of size 32 for 20 epochs. Its architecture consists of an input layer and one hidden layer containing 32 neurons with a Sigmoid activation function (preferred for binary classification tasks) and an output layer that leverages BCEwithLogitsLoss.

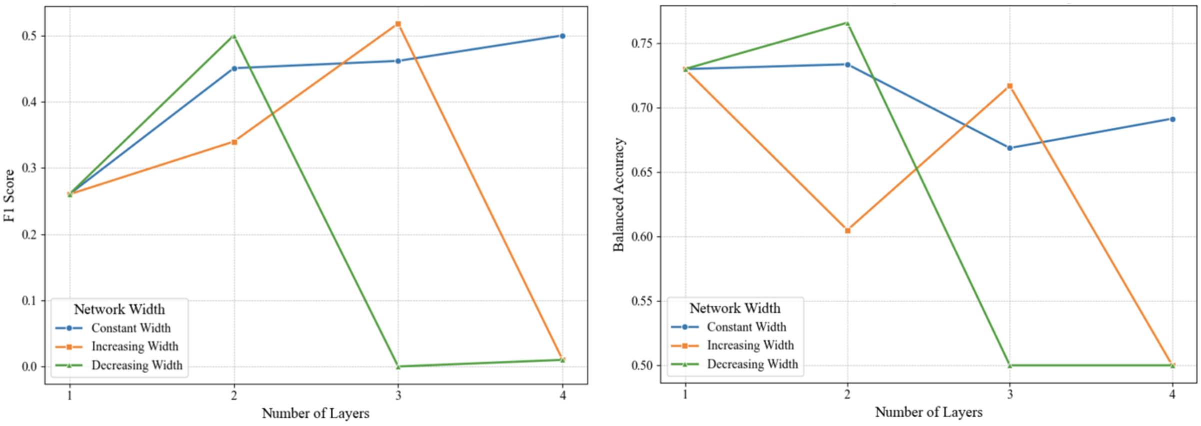

Figure 3.1: F1 scores and balanced accuracy for varying neural network lengths and widths

Figure 3.1 demonstrates that increasing the width (number of neurons per layer) and length (number of layers) of the network does not necessarly lead to improved performance, especially when going beyond two layers, at which point the network predicts all outcomes as 0 due to the unbalanced distribution of y. Hence why balanced accuracy was favoured as a performance metric. These findings also correspond to the literature, which generally hold that randomly initializing the weights of the network before applying gradient descent through backpropagation will result in poor solutions for networks with 3 or more hidden layers since the process tends to get stuck in poor local minima (Sarker, 2021). Leveraging random search can help further improve the network architectue by iterating through additional hyperparameter options. Note that random search was favoured over grid search since it generally performs as well, whilst promoting computational efficiency (Bergstra & Bengio, 2012).

ii Testing, Logistic regression and XGBoost

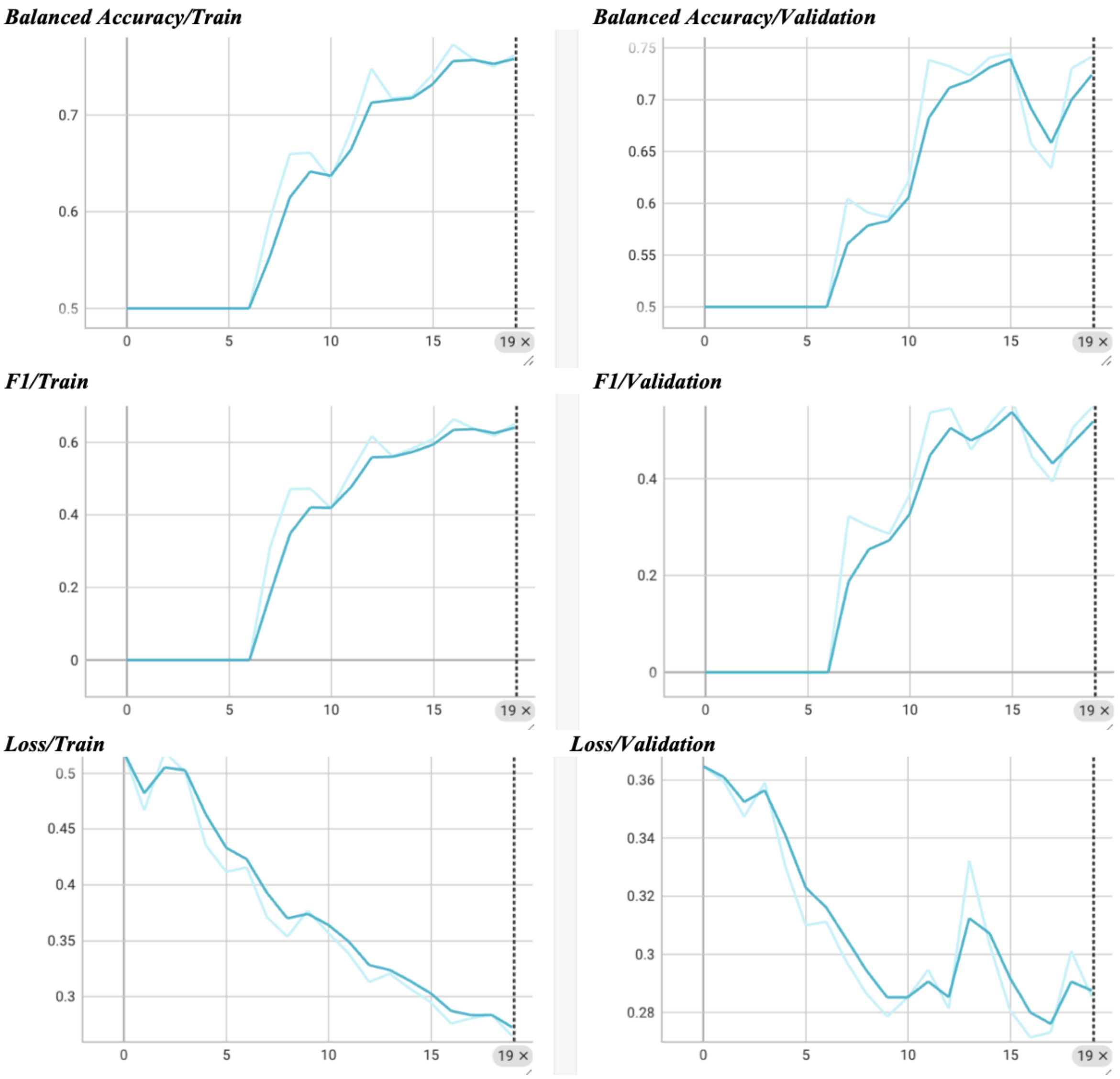

Figure 3.2: Final neural network under the hyperparameters proposed by random search

Table 3.1: Model Performance

Metric	Neural Network	Logistic Regression	XGBoost
F1	0.5735	0.5693	0.4561
Balanced Accuracy	0.7736	0.7157	0.6501

When testing the neural network, it achieved a balanced accuracy of 0.77 and an F1 score of 0.57 which are similar to the values for the training and validation set, suggesting a low risk of overfitting. The logistic regression achieved similar values whilst XGBoost performed slightly worse, perhaps because its hyperparameters had to be moderated to avoid excessive memory use. These findings are consistent with those of the literature, which often finds that for binary classification with tabular data logistic regressions and neural networks often have a similar accuracy (Sarker, 2021). Nevertheless it is important to note that as a result of the weight penalty the neural network correctly predicted 71 deaths while the logisic regression predicted 39 correct deaths. In medical contexts the neural network may therefore be of greater use.

iii Principal Component Analysis

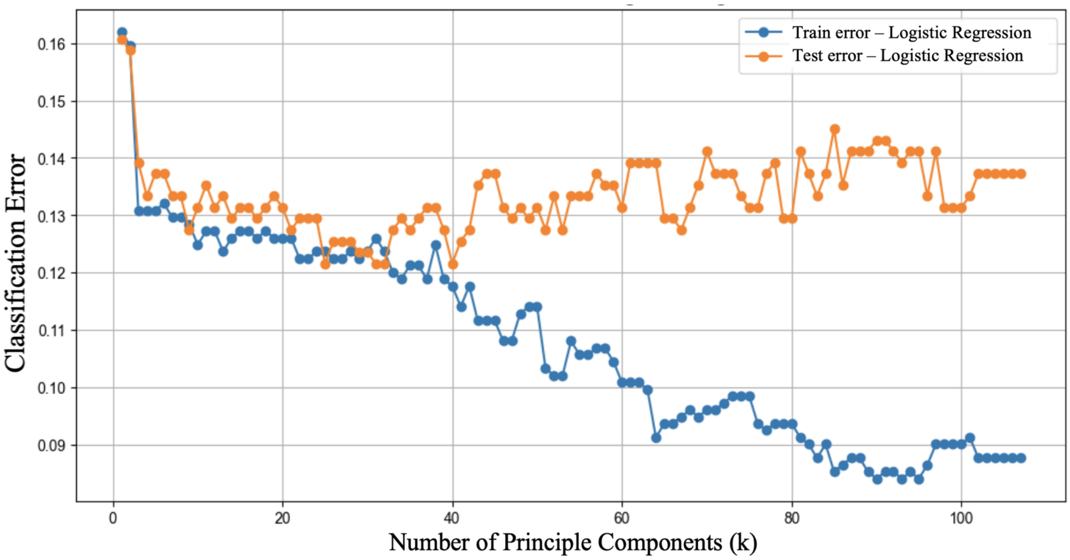

Figure 3.3: Logistic regression for principle components

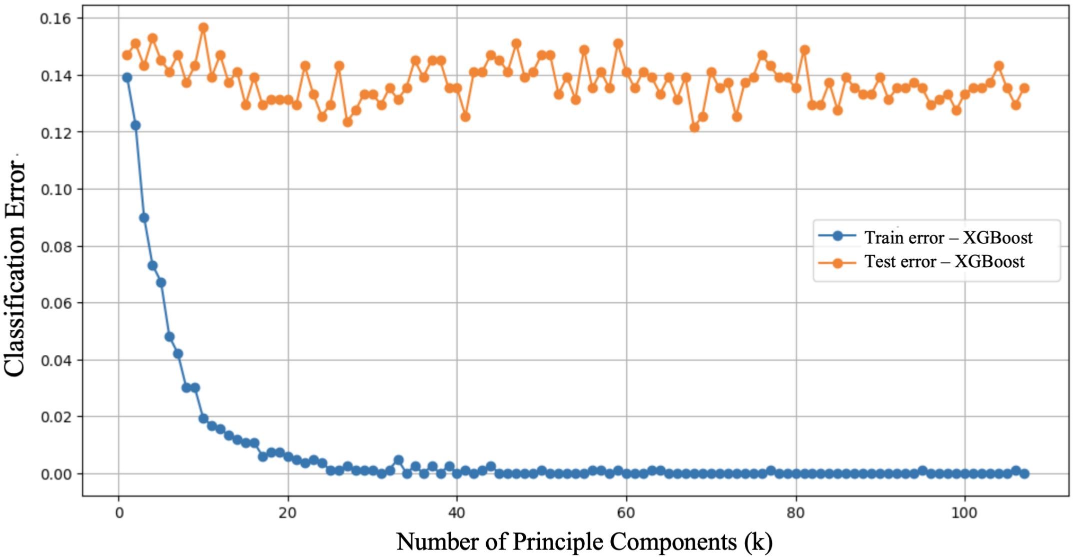

Figure 3.4: XGBoost for principle components

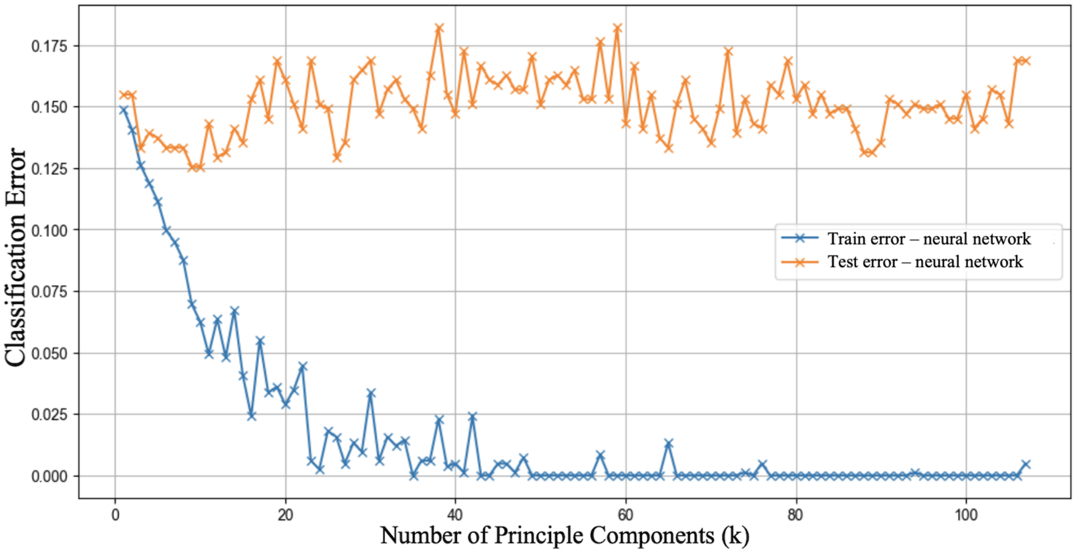

Figure 3.5: Neural network for principal components

As expected, figure 3.3 demonstrates that the logistic regression stabilizes early with consistently strong predictions for the test set. The neural network and XGBoost aim to capture more complex relationships, but given the noisiness of the data, doing so does not reliably improve their performance beyond 20 components, further highlighting the importance of mitigation against overifting. To this end, the visualizations effectively captures that although both the logistic regression and the neural network perform similarly on the test data, the neural network exhibits significantly more variance whilst the logistic regression is more prone to error due to bias. However, it does not capture that the neural network exhibits a higher accuracy for predicting ”deaths” due to the weight penalty introduced. This might be favorable in medical contexts.

3.2 Citations

Sarker, I. H. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science, 2*(1), 420.

Mishra, S., & Khare, D. (2014). On comparative performance of multiple imputation methods for moderate to large proportions of missing data in clinical trials: A simulation study. *Journal of Medical Statistics and Informatics, 2*(1), 9.

Rubin, D. B. (1976). Inference and missing data. *Biometrika, 63*(3), 581–592.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research, 13*, 281–305.