For this assignment we decided to create and train a Neural Network to predict heart disease. The dataset consists of 918 samples each with 11 features, some notable features being the patient’s age, sex, chest pain type, resting blood pressure, cholesterol level, blood sugar levels while fasting, resting electrocardiogram results, and maximum heart rate. We used to predict whether the patient has heart disease or not. We utilized the Pandas library to read the .csv file and to sample the data to create unique training, validation, and testing data. Since the data had a mixture of categorical and numerical data, we utilized Pandas’ get_dummies method to encode the categorical data so it could be used alongside the numerical data.

We used Sequential networks for both the Keras and Pytorch models. During testing for the Keras model, we varied the batch size, optimizers, number of layers and number of outputs from each layer. As we removed layers or increased the number of batches, the testing accuracy dropped. We landed on a 32x16x8x128x5x1 architecture for our Keras model. For the Pytorch model, we settled on a much simpler architecture for our network, that being 32x8x16x64x1. This is due to the Pytorch model outperforming the Keras model in accuracy.

The Pytorch model was more challenging to create due to the difficulty of understanding how the library worked. Once we understood the library and corrected a minor error causing the model to always predict true, our accuracy reached a more desirable range than the Keras model did.

When seeded (3520), the Keras model had an accuracy of 92.391%, a precision of 97.67%, and a recall of 87.5%. When using the same seed, the Pytorch model achieved 100% accuracy on the testing sample. Below are the loss graphs and confusion matrices for both models. Given the background of the problem that we are attempting to predict with this model we feel that it is more desireable to achieve a higher accuaracy on our recall than our precision. This is due to the fact that this would result in a situation where our model would have better accuarcy in correctly classifying individuals who did have heart disease rather than doing so for those individuals that are healthy and then misclassifying those who had the disease.

Unfortunately we believe that, while our accuaracy is at a reasonable level, our Keras model does fall short in the regard that our recall is in fact lower than the precision. Overall, we still feel confident in the performance of our models, although we do see areas of slight shortcomings in results, and through our implementation of these models we discovered that we have a personal preferance towards working with the Pytorch model rather than Keras as we find Keras to be more difficult to generally implement and harder for us to achieve our desired reults.

Figure 1 – Keras Loss Graph

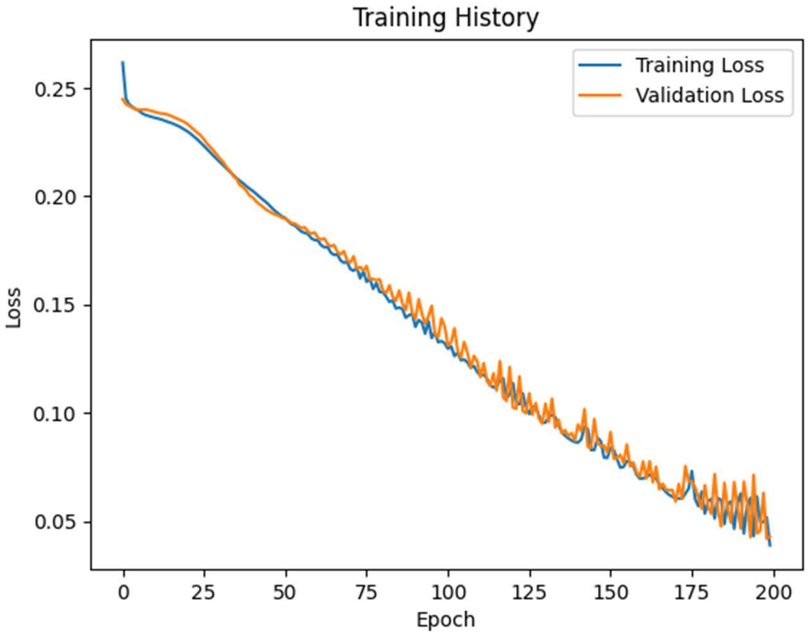


Figure 2 – Keras Confusion Matrix

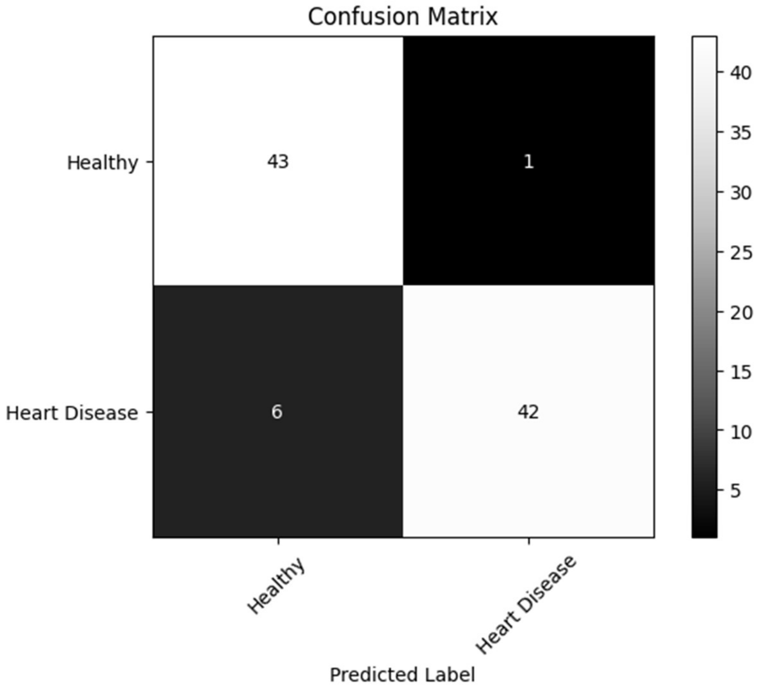


Figure 3 – Pytorch Loss Graph

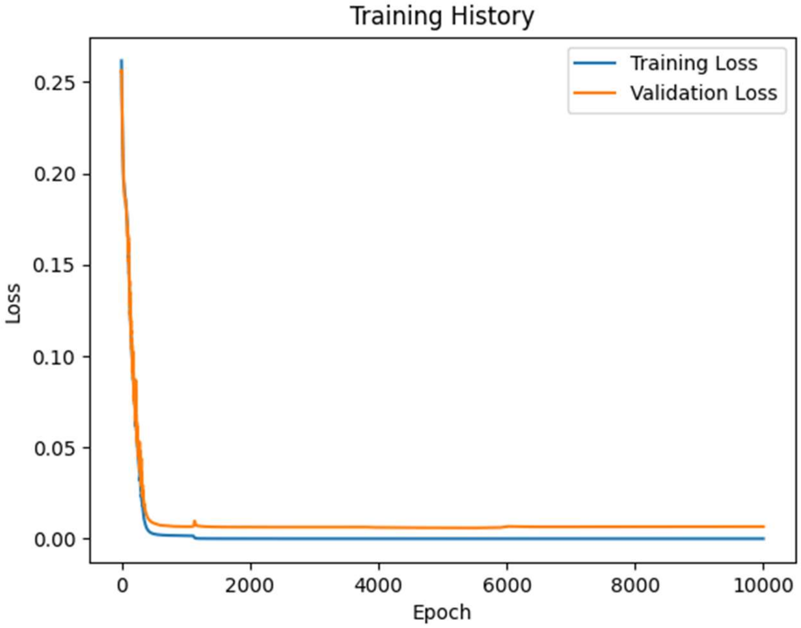
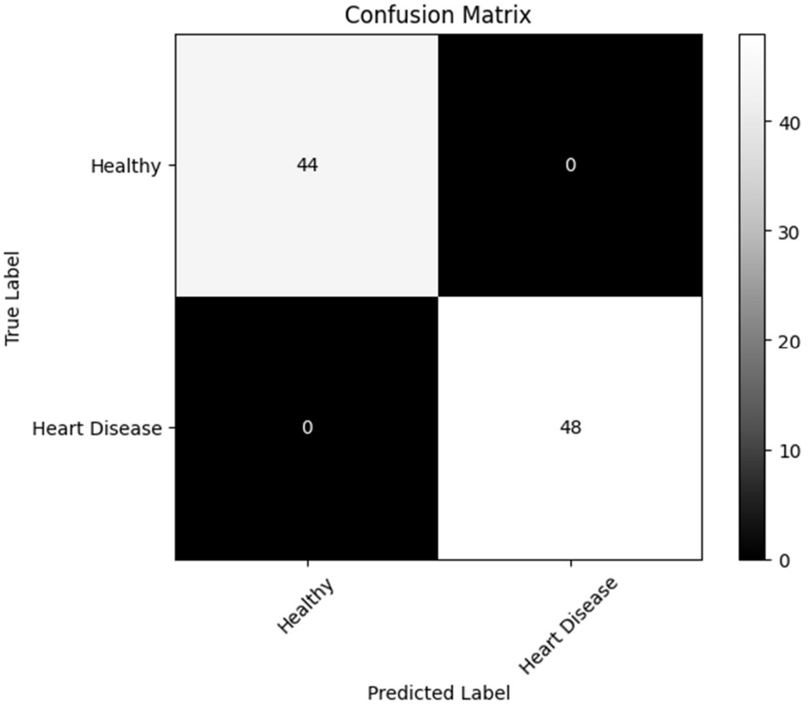


Figure 4 – Pytorch Confusion Matrix


Confusion Matrix
True Label
Healthy
Heart Disease
Healthy
Heart Disease
Predicted Label