## Explanation of code

In this code, we first converted the ARFF file into a CSV and then split it into the features (X) and the labels (y). Then I had to normalize the features using the MinMaxScaler, which split the data into training and testing sets which also meant it was normalised to a scale of [0, 1], and finally the last task was to build the neural network using the Keras Sequential API.

The neural network consists of two hidden layers with 20 and 10 nodes respectively as per the assignment brief, and an output layer with a SoftMax activation function to classify the data into the two classes. I then compiled the model using the Adam optimizer and sparse\_categorical\_crossentropy loss function, then trained it for 100 epochs using a batch size of 32 and a validation split of 0.1.

After training, you can see that I evaluated the model on the test set and then printed the test accuracy. Then I also plotted the training and validation accuracy over time using Matplotlib. Finally, the program made predictions on the test set, printed the classification report and confusion matrix using scikit-learn's classification\_report and confusion matrix functions, to output them to the console.

## Justification of Activation Function

Based on the dataset and problem description, it looks like we had 17 input features, and we want to classify each eye as having diabetic retinopathy or being normal. Therefore, we need 2 output neurons for the binary classification. Therefore, X=17 and Y=2.

For the hidden layer, we will use 20 neurons as specified in the problem description.

A common choice for the activation function in the hidden layer is ReLU (Rectified Linear Unit), which has been shown to work well in many cases. ReLU is a simple and computationally efficient function that allows the network to learn complex, nonlinear relationships between the inputs and outputs.

For the output layer, since we have a binary classification problem, we can use the sigmoid activation function. Sigmoid squashes the output to a range between 0 and 1, which can be interpreted as the probability of the input belonging to class 1 (diabetic retinopathy).