**Overview of the Analysis:**

The purpose of this analysis is to develop a deep learning model for a binary classification problem. A nonprofit foundation Alphabet Soup wanted a tool that could help them select the applicants for funding with the best chance of success in their ventures.

Providing us with a CSV containing more than 34,000 organisations that have received funding from Alphabet Soup we were able to run a number of tests using Google Colab.

The analysis focuses on training neural network models with different architectures, adjusting hyperparameters, and evaluating their performance based on loss and accuracy metrics.

**Results:**

Data Preprocessing:

Target Variable(s): The target variable for the model is 'IS\_SUCCESSFUL'. It represents the binary outcome that the model aims to predict.

Features Variable(s): The features (input data) for the model include the following columns:

'APPLICATION\_TYPE'

'USE\_CASE'

'ORGANIZATION'

'INCOME\_AMT'

'ASK\_AMT'

Variables to be Removed: The non-beneficial ID columns ('NAME' and 'SPECIAL\_CONSIDERATIONS') and the 'CLASSIFICATION' column have been dropped from the input data.

Compiling, Training, and Evaluating the Model:

Model Architecture: The first neural network model has two hidden layers with 80 and 30 neurons, respectively. The activation function used is 'relu' in both hidden layers, and the output layer has 1 neuron with the 'sigmoid' activation function.

Model Performance: The first model achieved the following results:

Loss: 0.7210

Accuracy: 0.6542

Attempts to Improve Model Performance:

The second model changed the architecture, having two hidden layers with 128 and 64 neurons, respectively. The activation function 'relu' is used in both hidden layers, and the output layer has 1 neuron with the 'sigmoid' activation function.

The third model has 140 neurons in the first hidden layer and 20 neurons in the second hidden layer, with 'relu' activation functions. The output layer remains the same with 1 neuron and the 'sigmoid' activation function.

The fourth model used the same architecture as the second model but trained for 20 epochs instead of 10 epochs.

Model Performance:

Model 2: Loss: 0.5599, Accuracy: 0.7323

Model 3: Loss: 0.5598, Accuracy: 0.7325

Model 4: Loss: 0.5583, Accuracy: 0.7343

**Summary:**

The analysis involved training multiple neural network models with different architectures and hyperparameters to solve a binary classification problem. The models achieved moderate accuracy, ranging from approximately 65% to 73%, and the losses decreased across the models, indicating an improvement in model predictions.

Increasing the number of neurons and/or epochs led to slightly better model performance, however the overall improvements were not substantial, and the models still show room for enhancement.

**Conclusion**

In conclusion, while the neural network models provided reasonable accuracy, with more time and more RAM availability; we believe that further experimentation with hyperparameter tuning and exploring other modelling techniques might lead to even better performance for this specific classification problem and obtain that 75% or more accuracy result.