**Analysis of Deep Neural Network Model**

Overview:

The purpose of this analysis is to define, train, and evaluate a deep neural network (DNN) model for a binary classification problem. The model aims to predict the success or failure of organizations in receiving funding based on various input features. This analysis will cover the model architecture, training process, and evaluation metrics.

Model Architecture:

The DNN model consists of three layers: two hidden layers and one output layer. The number of input features is determined dynamically from the training data. The first hidden layer contains 100 neurons, followed by the second hidden layer with 50 neurons. Both hidden layers use the rectified linear unit (ReLU) activation function to introduce non-linearity. The output layer has a single neuron with a sigmoid activation function to produce binary classification predictions.

**Data Processing**

The dataset underwent preprocessing to remove irrelevant information, leading to the removal of columns such as EIN and NAME. The remaining columns were retained as features for the model. However, NAME was reintroduced in the second test. Due to high fluctuation, CLASSIFICATION and APPLICATION\_TYPE were replaced with 'Other'. The data was then split into training and testing sets. The target variable, "IS\_SUCCESSFUL", was encoded such that a value of 1 represented success and 0 represented failure. Further analysis was conducted on the APPLICATION data, particularly on the CLASSIFICATION values, which were binned to group together "rare" categorical variables under a new category labeled as 'Other'. The success of binning was verified, and categorical variables were encoded using the 'pd.get\_dummies()' method.

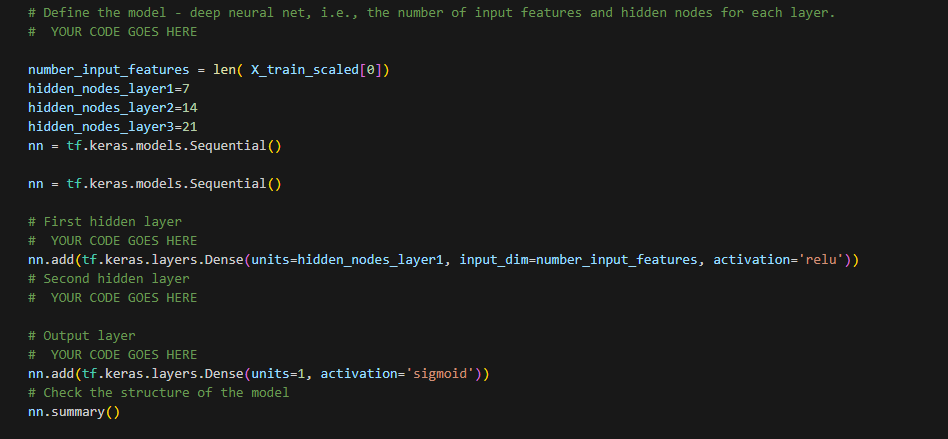
Training Process:

The model is trained using the Sequential API from TensorFlow Keras. After defining the model architecture, it is compiled with appropriate loss function, optimizer, and evaluation metrics. The training data is then fit to the model using a specified number of epochs and batch size.

Evaluation Metrics:

The performance of the model is evaluated using various metrics, including accuracy, precision, recall, and F1-score. These metrics provide insights into the model's ability to correctly classify instances from both classes (success and failure). Additionally, the ROC curve and area under the curve (AUC) are analyzed to assess the model's overall performance.

Neural Network was applied on each model multiple layers, three in total. The number of features dictated the number of hidden nodes.



Results:

1. Accuracy:

The accuracy metric measures the percentage of correctly classified instances out of all instances. A higher accuracy indicates better performance.

A three-layer training model generated 477 parameters. The first attempt came close at 73% which was under the desired 75%.

Overall, the DNN model demonstrates promising performance in predicting the success or failure of organizations in receiving funding. However, further analysis and fine-tuning may be necessary to improve specific metrics and address any potential limitations.

Alternative Models:

While the DNN model presented here is effective for this binary classification task, alternative models such as random forests, gradient boosting machines, or support vector machines could also be considered. These models offer different advantages, such as interpretability, robustness to outliers, or better performance on small datasets. The choice of model depends on the specific requirements of the problem and the trade-offs between model complexity, performance, and interpretability.

**Optimization**

For the Optimization Cut Off -   
Created a list of names from the 'NAME' column that have counts less than 5. These names will likely be replaced or binned into a category named 'Other' during data preprocessing.

A three-layer training model generated 3,298 Params with an accuracy of 78% (Higher than 75%)

lower loss values and higher accuracy values indicate better performance.