**Report on Neural Network**

### **Overview of the Analysis**

The purpose of this analysis is to assist the nonprofit foundation Alphabet Soup in identifying the applicants for funding who are most likely to succeed in their ventures. By utilizing machine learning and deep learning techniques, a binary classifier was developed to predict whether applicants will be successful if funded. This analysis was conducted using a dataset of over 34,000 organizations funded by Alphabet Soup, which includes metadata such as application type, classification, use case, and funding amount.

### **Results**

#### **Data Preprocessing**

What variable(s) are the target(s) for your model?

* **Target Variable:**
  + ‘IS\_SUCCESSFUL’: This binary variable indicates whether the funding was used effectively (1 for success, 0 for failure).

What variable(s) are the features for your model?

* **Feature Variables:**
  + ‘APPLICATION\_TYPE’: The type of application submitted.
  + ‘AFFILIATION’: The affiliated sector of the applicant.
  + ‘CLASSIFICATION’: Government classification of the organization.
  + ‘USE\_CASE’: The use case for the funding.
  + ‘ORGANIZATION’: The type of organization.
  + ‘STATUS’: Active status of the organization.
  + ‘INCOME\_AMT’: Income classification of the organization.
  + ‘SPECIAL\_CONSIDERATIONS’: Whether the application had special considerations.
  + ‘ASK\_AMT’: The amount of funding requested.

What variable(s) should be removed from the input data because they are neither targets nor features?

* **Removed Variables:**
  + ‘EIN’ and ‘NAME’: These columns are unique identifiers that do not contribute to the predictive power of the model.

#### **Compiling, Training, and Evaluating the Model**

How many neurons, layers, and activation functions did you select for your neural network model, and why?

* **Neurons, Layers, and Activation Functions:**
  + The model includes:
    - Input Layer: Matches the number of features after preprocessing.
    - First Hidden Layer: 80 neurons with ReLU activation.
    - Second Hidden Layer: 30 neurons with ReLU activation.
    - Output Layer: 1 neuron with sigmoid activation for binary classification.
  + These choices were made to balance model complexity and performance, ensuring sufficient capacity to learn from the data without overfitting.

Were you able to achieve the target model performance?

* **Model Performance:**
  + The model achieved an accuracy of approximately 73% on the test data.

What steps did you take in your attempts to increase model performance?

* **Steps to Improve Performance:**
  + Grouped rare categories in ‘APPLICATION\_TYPE’ and ‘CLASSIFICATION’ to reduce noise.
  + Used ‘pd.get\_dummies’ to encode categorical variables.
  + Applied feature scaling using ‘StandardScaler’ to normalize the data.
  + Experimented with different numbers of neurons and layers.
  + Adjusted the learning rate and batch size during training.

### **Summary**

The deep learning model achieved a moderate accuracy of 73%, which is reasonable for a binary classification problem. However, further improvements could be made by:

* Using a Random Forest or Gradient Boosting model, which may better capture non-linear relationships in the data.
* Applying feature selection techniques to identify and retain only the most important features.
* Using cross-validation to tune hyperparameters more effectively.

These approaches could provide better insights and improve the predictive accuracy of the model.