**REPORT**

**1. Overview of the Analysis**

**Purpose of the Analysis:** The purpose of this analysis is to build a binary classifier using a neural network to predict the success of applicants funded by Alphabet Soup. The goal is to develop a model that can help Alphabet Soup select applicants with the best chance of success.

**2. Results**

**Data Preprocessing:**

* **Target Variable(s):**
  + IS\_SUCCESSFUL: Indicates whether the funding was used effectively.
* **Feature Variable(s):**
  + APPLICATION\_TYPE
  + AFFILIATION
  + CLASSIFICATION
  + USE\_CASE
  + ORGANIZATION
  + STATUS
  + INCOME\_AMT
  + SPECIAL\_CONSIDERATIONS
  + ASK\_AMT
* **Removed Variable(s):**
  + EIN: Identification column.
  + NAME: Identification column.

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

* **Model Architecture:**
  + **Number of Layers:**
    - Input Layer
    - 3 Hidden Layers
    - Output Layer
  + **Neurons per Layer:**
    - 1st Hidden Layer: 128 neurons
    - 2nd Hidden Layer: 64 neurons
    - 3rd Hidden Layer: 32 neurons
  + **Activation Functions:**
    - Hidden Layers: ReLU (Rectified Linear Unit)
    - Output Layer: Sigmoid
* **Model Performance:**
  + Target performance: 75% accuracy
  + Achieved performance:
    - **Loss:** 0.5653
    - **Accuracy:** 73.16%
* **Optimization Steps Taken:**
  + Increased the number of neurons in each hidden layer.
  + Added more hidden layers to improve model capacity.
  + Experimented with different activation functions like tanh and leaky\_relu.
  + Adjusted the learning rate for the optimizer.
  + Increased the number of training epochs to allow the model more time to learn.
  + Implemented dropout layers to prevent overfitting.

**Data Preprocessing:**

1. **Loading the Data:**
   * The data was loaded from a CSV file.
2. **Dropping Irrelevant Columns:**
   * EIN and NAME columns were removed as they do not contribute to the prediction.
3. **Handling Categorical Variables:**
   * Combined rare categories in APPLICATION\_TYPE.
   * Converted categorical variables to numerical using one-hot encoding.
4. **Splitting the Data:**
   * The data was split into training and testing sets.
5. **Scaling the Data:**
   * Features were scaled using StandardScaler for normalization.

**Model Definition:**

1. **Model Architecture:**
   * Defined a neural network with 3 hidden layers.
   * Used ReLU activation for hidden layers and sigmoid for the output layer.
2. **Compiling the Model:**
   * Used Adam optimizer.
   * Binary crossentropy as the loss function.
   * Accuracy as the evaluation metric.

**Training the Model:**

1. **Training:**
   * Trained the model for 100 epochs.
   * Used ModelCheckpoint to save the model during training.
2. **Evaluation:**
   * Evaluated the model on the test set.
   * Achieved a loss of 0.5653 and accuracy of 73.16%.

**Summary:**

The neural network model was able to achieve an accuracy of 73.16%. Although this is slightly below the target of 75%, several optimization techniques were applied, including increasing the number of neurons, adding more layers, using different activation functions, and implementing dropout for regularization.