**Alphabet Soup Neural Network Model Analysis Report - Module 21 – Deep Learning Challenge**

**Analysis Overview:**

Alphabet Soup, a nonprofit foundation, aims to identify funding applicants with the highest chances of success. To achieve this, we developed a deep learning neural network model for a classification problem. The objective was to accurately classify instances based on the provided features.

Results: Data Preprocessing

**Results:**

**Data Preprocessing**

* **Target Variable:**
  + The target variable for the model is "IS\_SUCCESSFUL"
* **Features:**
  + After deleting “EIN” and “NAME” features, the remaining features include:
    - APPLICATION\_TYPE
    - AFFILIATION
    - CLASSIFICATION
    - USE\_CASE
    - ORGANIZATION
    - STATUS
    - INCOME\_AMT
    - SPECIAL\_CONSIDERATIONS
    - ASK\_AMT
* **Dropped Features:**
  + During optimization:
    - In v1: Dropped “STATUS,” “AFFILIATION,” and “SPECIAL\_CONSIDERATIONS.” Surprisingly, this had -ve impact on accuracy.
    - In v3: Dropped “STATUS.”

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

* **Initial version (StaterCode.jpynb)**
  + Accuracy: 0.7270
  + Key Decisions:
    - Deleted “EIN” and “NAME,” retaining other features.
    - Application type value counts < 500.
    - Classification value counts cutoff < 1000.
    - 2 hidden node layers.
    - Hidden node values: Layer 1 (8 nodes), Layer 2 (4 nodes).
    - Activation function: ReLU.
    - Optimizer: Adam.
    - Epochs: 100.
* **Optimization 1 (Starter CodeOptimization1.jpynb)**
  + Accuracy: 0.6599
  + Key Improvements:
    - Dropped “EIN,” “NAME,” “AFFILIATION,” “SPECIAL\_CONSIDERATIONS,” and “STATUS.”
    - Increased application type value counts cutoff to < 1000.
    - Increased classification value counts cutoff to < 1500.
    - 4 hidden node layers.
    - Hidden node values: Layer 1 (16 nodes), Layer 2 (12 nodes), Layer 3 (8 nodes), Layer 4 (4 nodes).
    - Activation function: Mish (combines benefits of ReLU and Tanh).
    - Optimizer: Nadam (Adam with Nesterov momentum).
    - Epochs: 250.
* **Optimization 2 (Starter CodeOptimization2.jpynb)**
  + Accuracy: 0.7238
  + Key Adjustments:
    - Retained all features (no drops).
    - Experimented with application type value counts cutoff (500 to 1000 and back).
    - Experimented with classification value counts cutoff (1000 to 1500 and back to 1000).
    - 4 hidden node layers.
    - Hidden node values: Layer 1 (24 nodes), Layer 2 (12 nodes), Layer 3 (6 nodes), Layer 4 (3 nodes).
    - Activation function: Mish.
    - Optimizer: Nadam.
    - Lowered epochs to 50.
* **Optimization 3 (Starter CodeOptimization3.jpynb)**
  + Accuracy: 0.7254
  + Key Changes:
    - Deleted “EIN” and “NAME.”
    - Dropped the “STATUS” feature.
    - Application type value counts cutoff < 500.
    - Classification value counts cutoff < 1000.
    - 4 hidden node layers.
    - Hidden node values: Layer 1 (30 nodes), Layer 2 (20 nodes), Layer 3 (10 nodes), Layer 4 (4 nodes).
    - Activation function: Mish.
    - Optimizer: Nadam.
    - Epochs: 40.

**Summary and Recommendations**

After multiple model training attempts, the maximum accuracy did not reach the .75 threshold. Further research suggests that increasing the dataset size could improve model accuracy.