OVERVIEW

The purpose of this analysis was to develop a binary classification model for Alphabet Soup, a nonprofit organization. The model was designed to predict the probability of success for funding applicants based on various components of the dataset. Using a deep learning approach, a neural network was built, optimized, and evaluated to achieve the best possible accuracy for the classification problem.

RESULTS

Data Preprocessing

- Target Variable:
  - "IS_SUCCESSFUL": binary outcome representing funding success/failure)
- Feature Variables:
  - "ASK_AMT": scaled using StandardScaler
  - "INCOME_AMT": scaled using StandardScaler
- Removed Variables:
  - "EIN": ID column, irrelevant to model
  - "NAME": text-based column with no predictability
  - "STATUS": very low variance (majority status "1")
  - "SPECIAL_CONSIDERATIONS": vet low variance (majority special considerations "NO")

Compiling, Training and Evaluating the Model

- **Alphabet Soup Charity Model:**
  - Dropped columns "EIN" and "Name"
  - Input Layer: Features from the dataset
  - Hidden Layers:
    - 1st layer: 80 neurons, ReLU activation
    - 2nd layer : 30 neurons, ReLU activation
  - Output Layer: 1 neuron, Sigmoid activation
  - Results:
    - Accuracy: 72.83%
    - Loss: 0.5660

- Analysis: The accuracy indicates the model correctly predicted funding success for approximately 73% of test cases. However, the loss suggests there is room for improvement.

- **Optimization Attempt 1**:
  - Dropped additional column “STATUS” and “SPECIAL_CONSIDERATIONS”
  - Added one additional hidden layer (3 total):
    - 1st layer: 100 neurons
    - 2nd layer: 50 neurons
    - 3rd layer: 25 neurons
  - Used the same activation functions ReLU for hidden layers, Sigmoid for output
  - Results:
    - Accuracy: 72.90%
    - Loss: 0.5716
  - Analysis: Removing additional unnecessary columns, adding an additional layer and increasing the neurons improved accuracy but the increase was minimal.

- **Optimization Attempt 2**:
  - Increased neurons in all layers:
    - 1st layer: 200 neurons.
    - 2nd layer: 150 neurons.
    - 3rd layer: 100 neurons.
  - Maintained ReLU activation for hidden layers and Sigmoid activation for output.
  - Results:
    - Accuracy: 73.01%
    - Loss: 0.6055
  - Analysis: Increasing neurons improved accuracy marginally, but it also increased the loss. This suggests potential overfitting.

- **Optimization Attempt 3**:
  - Added a 4th hidden layer to make the model deeper:
    - 1st layer: 200 neurons.
    - 2nd layer: 150 neurons.
    - 3rd layer: 100 neurons.
    - 4th layer: 50 neurons.
  - ReLU activation for hidden layers and Sigmoid for output layer.
  - Results:
    - Accuracy: 73.07%

- Loss: 0.6191

- Analysis: Adding an additional layer improved accuracy slightly, but the loss increased further. The improvement in accuracy is minimal compared to the added complexity.

Summary

Overall Results:

- The best accuracy achieved was 73.07% with the third optimization attempt. In this attempt two additional hidden layers were added, and neurons were progressively increased throughout earlier attempts to reach this outcome.

Challenges:

- Loss metrics suggest limitations in predictability, leading to further complexity.
- Target model performance of 75% was not achieved.

Recommendation:

- Conduct feature engineering to extract additional meaningful features or explore external datasets to enhance predictive power.