Written Analysis

# Overview of the Analysis

The purpose of this analysis is to create a binary classification model using deep learning to predict the success of organizations funded by Alphabet Soup based on various features provided in the dataset. The primary goal is to achieve a target predictive accuracy higher than 75%. The analysis involves data preprocessing, model compilation, training, evaluation, optimization, and a detailed report on the performance of the deep learning model.

# Results

## Data Preprocessing

Target and Feature Variables

* Target Variable(s): IS\_SUCCESSFUL
* Feature Variable(s): All other columns except EIN and NAME.

Removed Columns

* EIN and NAME columns were dropped as they do not contribute to the prediction.

Unique Values

* The number of unique values for each column was determined.

Handling Categorical Variables

* For columns with more than 10 unique values, the number of data points for each unique value was examined.
* A new value called "Other" was created to bin rare categorical variables together.

Feature and Target Arrays

* Feature array, X, and target array, y, were created using the preprocessed data.

Train-Test Split

* The preprocessed data was split into training and testing datasets.

Scaling

* The data was scaled using a **StandardScaler** fitted to the training data.

## Original Model

The original model had two layers and one output layer. The number of nodes chosen was 8 and 5. Smaller numbers of nodes, such as 8 and 5, may prevent the model from becoming overly complex and overfitting to the training data. Similar logic was used when selecting the number of epochs for the data. The following performance:

* Loss: 0.5549
* Accuracy: 0.7235

A screenshot of a computer

Description automatically generated

## Optimization Method 1 - Increased Epochs to 150

By increasing the number of epochs from 100 to 150, the model's performance slightly improved. The rationale behind this adjustment is to allow the model more opportunities to learn from the training data. The hope is that with additional epochs, the model captures complex patterns in the data, potentially leading to better generalization on the test set.

* Loss: 0.5587
* Accuracy: 0.7238

A screenshot of a computer program

Description automatically generated

## Optimization Method 2 - Increased Nodes in Hidden Layers

Increasing the number of nodes in the hidden layers to 90 and 70 was an attempt to make the model more expressive and capture intricate relationships within the data. A higher number of nodes allows the model to learn more complex representations. However, it's crucial to strike a balance, as too many nodes can lead to overfitting. The chosen values were based on experimentation to find a suitable trade-off.

* Loss: 0.5670
* Accuracy: 0.7259

A screenshot of a computer code

Description automatically generated

## Optimization Method 3 - Added a Third Hidden Layer

Adding a third hidden layer with 3 nodes was another attempt to enhance the model's capacity for learning hierarchical features in the data. The decision to use 3 nodes in the third layer was based on the principle of gradually decreasing the number of nodes in deeper layers, allowing the network to extract increasingly abstract features.

* Loss: 0.5547
* Accuracy: 0.7254

A screenshot of a computer

Description automatically generated

In addition to this, I attempted to alter the columns dropped by keeping the ‘NAME’ column. Unfortunately, my workbook kept crashing due to lack of available RAM so I could not see if this effected overall accuracy.

A close-up of a computer code

Description automatically generated

# Summary

The deep learning model achieved moderate success in predicting the success of Alphabet Soup-funded organizations. While the target accuracy of 75% was not reached, some optimizations showed marginal improvements.

## Model Configuration

* The original model had two hidden layers with 8 and 5 nodes, followed by an output layer.
* Optimization methods involved tweaking epochs, nodes in hidden layers, and adding an extra hidden layer.

## Model Performance

* The accuracy ranged from 72.35% to 72.59% across different optimization attempts.
* The loss remained in the range of 0.5547 to 0.5670.

## Recommendations

* Further exploration and experimentation are encouraged to identify the most effective combination of hyperparameters.
* Consider additional feature engineering or exploring different architectures to enhance predictive capabilities.

In summary, the deep learning model shows promise, but further refinement is needed to consistently achieve the desired predictive accuracy. Experimenting with different architectures, activation functions, or alternative models may lead to improved results. As mentioned previously, the columns dropped may change accuracy.