Alphabet Soup Charitable Funding Predictor Model

# Overview

The non-profit foundation Alphabet Soup wants to create an algorithm to predict whether or not applicants for funding will be successful. The foundation has provided a data set of several thousand funding events containing meta regarding the funding requests. A binary classifier has been created from this data set.

# Results

After some optimization, the classifier model achieved nearly 80% accuracy, as described below.

## Data Preprocessing

The data provided contained the following columns:

* EIN
* Name
* Application Type
* Affiliation
* Classification
* Use Case
* Organization
* Status
* Income Amount
* Special Considerations
* Ask Amount
* Is Successful

**Target Variables**

**The target variable is “Is Successful”, with a value of 1 indicating success and a value of 0 indicating a lack of success.**

**Feature Variables**

**In the initial development of the model, the following columns were presumed to have an influence on success of funding:**

* Application Type
* Affiliation
* Classification
* Use Case
* Organization
* Status
* Income Amount
* Special Considerations
* Ask Amount

**Upon initial review of the data set, it appeared that these variables had some influence on the success of a funding application. After optimization, it was determined that Name was also a feature variable and was included in the model.**

**Unnecessary Variables**

**Initially, the columns EIN and Name were excluded from the model. However, after optimization, Name was included as a feature variable.**

## Compiling, Training and Evaluating the Model

**Neurons, Layers and Activation Function**

**The first iteration of model had two hidden layers after the input layer. The hidden layers each contained 22 and 11 nodes, respectively. Both of the hidden layers used the Rectified Linear Unit (ReLU) activation function. The output layer had a single node and used the Sigmoid activation for its binary output.**

**After optimization, the model had three hidden layers with 100, 30 and 10 nodes each. The first hidden layer continued to use the ReLU activation function, while the second and third hidden layers used the Sigmoid activation function. The output layer continued to have a single node and use the Sigmoid activation function.**

**Model Performance**

**Initial model performance did not achieve the accuracy threshold of 75:**

**Graphical user interface, text, application

Description automatically generated**

**The optimized model achieved an accuracy score of close to 80%:**

**Text

Description automatically generated**

**Optimization Process**

**The first step in optimizing the model was to add back the Name column as a feature variable. The thought being that an organization’s track record has an effect on funding success.**

**Beyond the data itself, the model parameters were change as indicated:**

* **Hidden Layer 1**
  + **Nodes increased from 22 to 100**
* **Hidden Layer 2**
  + **Nodes increase from 11 to 30**
  + **Activation function changed from ReLU to Sigmoid**
* **Hidden Layer 3**
  + **Added additional hidden layer with 10 nodes**
  + **Activation function set to Sigmoid**

# Summary

Although the initial results of the model fell short of the minimum accuracy threshold of 75%, the model was ultimately optimized to have an accuracy of close to 80%. One column that was initially excluded was reintroduced. This is a reminder that initial assumptions about the data may not necessarily be correct.