Deep Learning Challenge—Alphabet Soup

Overview:

> "The nonprofit foundation Alphabet Soup wants a tool that can help it select the applicants for funding with the best chance of success in their ventures. With your knowledge of machine learning and neural networks, you'll use the features in the provided dataset to create a binary classifier that can predict whether applicants will be successful if funded by Alphabet Soup."

Results:

Data Preprocessing

What variable(s) are the target(s) for your model?

- The target variable was "IS_SUCCESSFUL" and has a value of 1 for yes and 0 for no

What variable(s) are the features for your model?

- I analyzed "APPLICATION_TYPE" and "CLASSIFICATION" was used for binning

What variable(s) should be removed from the input data because they are neither targets nor features?

- I removed "EIN" and "NAME" from the dataset

Compiling, Training and Evaluating the Model

How many neurons, layers, and activation functions did you select for your neural network model, and why?

- Initially used 2 hidden layers
  - One of the re-optimization techniques was to add a third hidden layer
  - Layer 1 had 80 neurons and layer 2 had 30 neurons
  - Selected "relu" for both of these layers because it activates the neurons one by one which makes the network calculated properly

Were you able to achieve the target model performance?

- No, my accuracy landed at 73% vs. the goal of 75%

What steps did you take in your attempts to increase model performance?

- Optimization Number 1
  - Increased the neurons within the hidden layers from 80 and 30 to 100 and 50
- Optimization Number 2
  - Added the 20 neurons to the hidden layers to make 100 and 50 nodes
  - Added an additional hidden layer of 10 nodes
  - Used sigmoid for the hidden layer activations instead of relu
- Optimization Number 3
  - Added the 20 neurons to the hidden layers to make 100 and 50 nodes
  - Used sigmoid for all layers again instead of relu
  - Increased the epoch from 100 to 200

Summary

The original model had an accuracy score of 72.9%. The first optimization resulted in an optimization score of 72.9%, the second optimization attempt resulted in 72.6% accuracy and the final optimization attempt resulted in 72.9% optimization. None of my attempts are optimization resulted in an accuracy score of 75% or above. I could've tried different activation layers than the "relu" or "sigmoid" activations. It seems like taking "NAME" may have not been helpful and could've reduced the amount of neurons.