Deep Learning Challenge – Alphabet Soup Charity

Overview

The goal of this project is to assist the nonprofit organization, Alphabet Soup, in identifying the funding applicants with the best chance of success in their businesses by using Deep Learning and Neural Networks technique.

Results:

- Data Preprocessing
  - Target array (y) is the column "IS_SUCCESSFUL"
  - Features arrays (X) are the remaining columns after converting categorical variables into numeric data
  - Identification columns "EIN" and "NAME" are removed from the data

- Compiling, Training, and Evaluating the Model

  I tried 3 attempts to optimize the model; however, the model still does not achieve 75% accuracy.

  - Attempt 1:
    - 2 hidden layers with 'relu' activation function
    - Layer 1: 80 neurons (relu) / Layer 2: 30 neurons (relu)
    - Epochs = 100
    - Results: 73% accuracy & 56% loss value

  - Attempt 2:

    By changing relu to sigmoid in the second layer, the model can learn more are not captured by the linear activation. However, the result is almost the same as the results of attempt 1.

    - 2 hidden layers with 'relu' and 'sigmoid' activation function
    - Layer 1: 80 neurons (relu) / Layer 2: 30 neurons (sigmoid)
    - Epochs = 100
    - Results: 72.8% accuracy & 56% loss value

  - Attempt 3:

    By adding the third layer, the model can potentially capture more complex features and improve the model's ability to capture non-linear relationships. To prevent overfitting

risk, I reduced the number of neurons in each layer. However, the result is almost the same as the results of the previous 2 attempts.

- 3 hidden layers with 'relu' and 'sigmoid' activation function
- Layer 1: 30 neurons (relu) / Layer 2: 15 neurons (sigmoid) / Layer 3: 20 neurons (sigmoid)
- Epochs = 100
- Results: 72.7% accuracy & 55% loss value

Summary:

My 3 attempts generated similar results, making it difficult to determine which model achieves the best classification results. It is unclear whether adding hidden layers or changing activation function would optimize the model. To reach the targeted 75% accuracy, I need to remove outliers from the data through more thorough data cleaning and add more hidden layers with different activation functions.