BootCampSpot

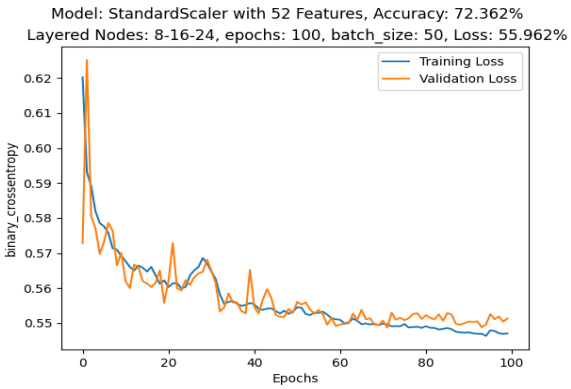
May 24, 2023

AlphabetSoup Charity Funding

Modeling Future Outcomes

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. BootCampSpot's students were challenged with a deep-learning assignment to produce a model for the company to utilize to help make those decisions based on information they have collected in the past regarding types of requested and other classifications and successes.

AlphabetSoup Charity Funding
Modeling Future Outcomes



01 Initial Challenge

Background

From Alphabet Soup's business team, you have received a CSV containing more than 34,000 organizations that have received funding from Alphabet Soup over the years. Within this dataset are a number of columns that capture metadata about each organization, such as: EIN and NAME—Identification columns, APPLICATION_TYPE—Alphabet Soup application type, AFFILIATION—Affiliated sector of industry, CLASSIFICATION—Government organization classification, USE_CASE—Use case for funding, ORGANIZATION—Organization type, STATUS—Active status, INCOME_AMT—Income classification, SPECIAL_CONSIDERATIONS—Special considerations for application, ASK_AMT—Funding amount requested, IS_SUCCESSFUL—Was the money used effectively

Initial findings

Upon our first attempts with reaching this goal, our model produced rather poor results with less than 73% accuracy and up to 61% losses. Utilizing the neural network models by increasing the number of nodes and decreasing the batch sizes, tamed this to a consistent accuracy of 72.3% and a loss of 55.9% based on 4 total layers containing nodes of 8, 16, 24, and 1, batch size of 50 and 100 epochs. We can do better!!

The process

We imported the necessary dependencies, read in the .csv file, analyzed the available data. Looking at the columns in the dataset we dropped the EIN and NAME columns due to lack of value, performed a revised binning of data with the ASK_AMT, the Application Type, and the Classification Code. Split the data into training and test features. Compiled, trained and evaluated the model with 3 hidden layers. Plotted the training and validation losses and saved the model.

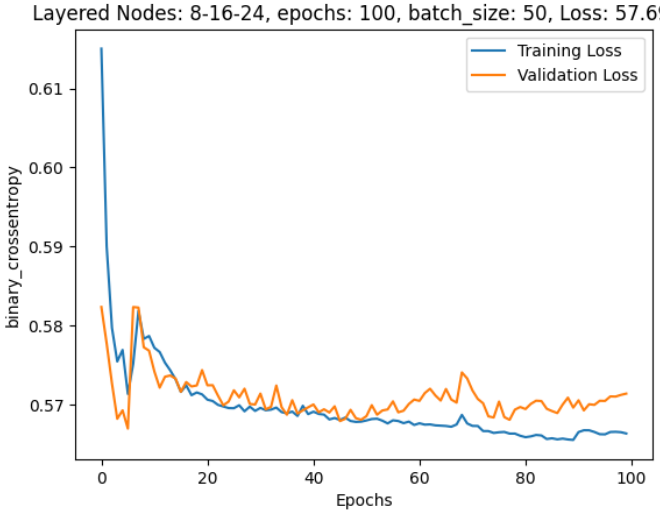
AlphabetSoup Charity Funding
Modeling Future Outcomes

02 Optimizing Challenge

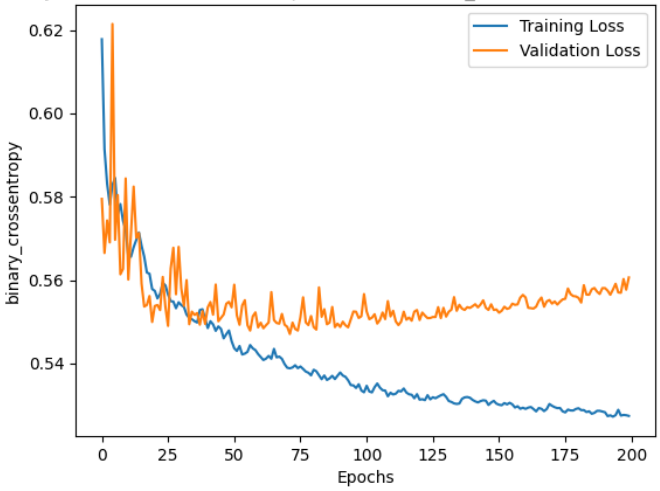
AlphabetSoupCharity_Optimization

We decided to try 3 more optimization types with the cleansed data from the original analyzation process. The graphs can be viewed on this page. The specifications are located in the title of the graphs for the inputs to graph the models. We experienced a higher success rate with three hidden layers using the activation selu.

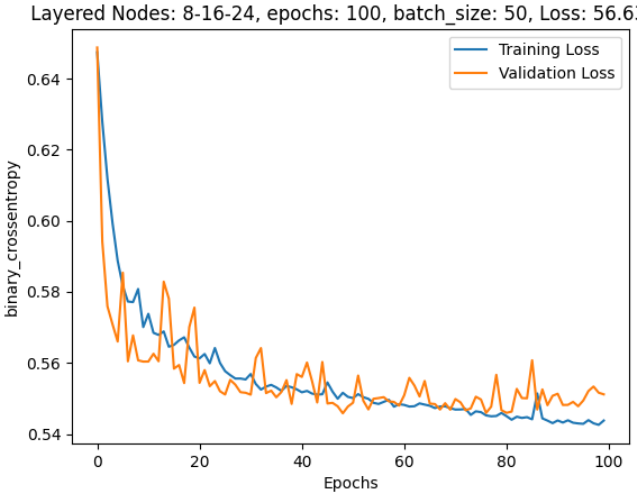
Model OPT 01: StandardScaler with 52 Features, Accuracy: 71.475%
Layered Nodes: 8-16-24, epochs: 100, batch_size: 50, Loss: 57.692%



Model OPT 02: StandardScaler with 52 Features, Accuracy: 72.676%
Layered Nodes: 16-32-48, epochs: 200, batch_size: 100, Loss: 56.030%



Model OPT 03: StandardScaler with 52 Features, Accuracy: 73.259%
Layered Nodes: 8-16-24, epochs: 100, batch_size: 50, Loss: 56.632%



AlphabetSoup Charity Funding
Modeling Future Outcomes

03 Optimizing Further

AlphabetSoupCharity_Extra_Optimizations

Further discussions with colleagues lead our group down the path to analyze multiple nodes, dropout rates, learning rates, and batch sizes. This seems rather promising and has brought us to over a 76% success rate according to one of the graphs listed below. We did decide to add the name of the charity back in as we felt it was an important feature to consider. Smart choice!

Summary Results

We likely would consider a less manual approach by using a KerasTuner HyperModel to make the decisions based on algorithms built. We ran 53 models to make this final choice for the least loss and best calculation for success in charity funding futures.