**DEEP LEARNING CHALLENGE REPORT**

*GT Bootcamp – Data Analytics  
Module 21 – Deep Learning Challenge  
Jason Hanlin  
June 22, 2023*

OVERVIEW

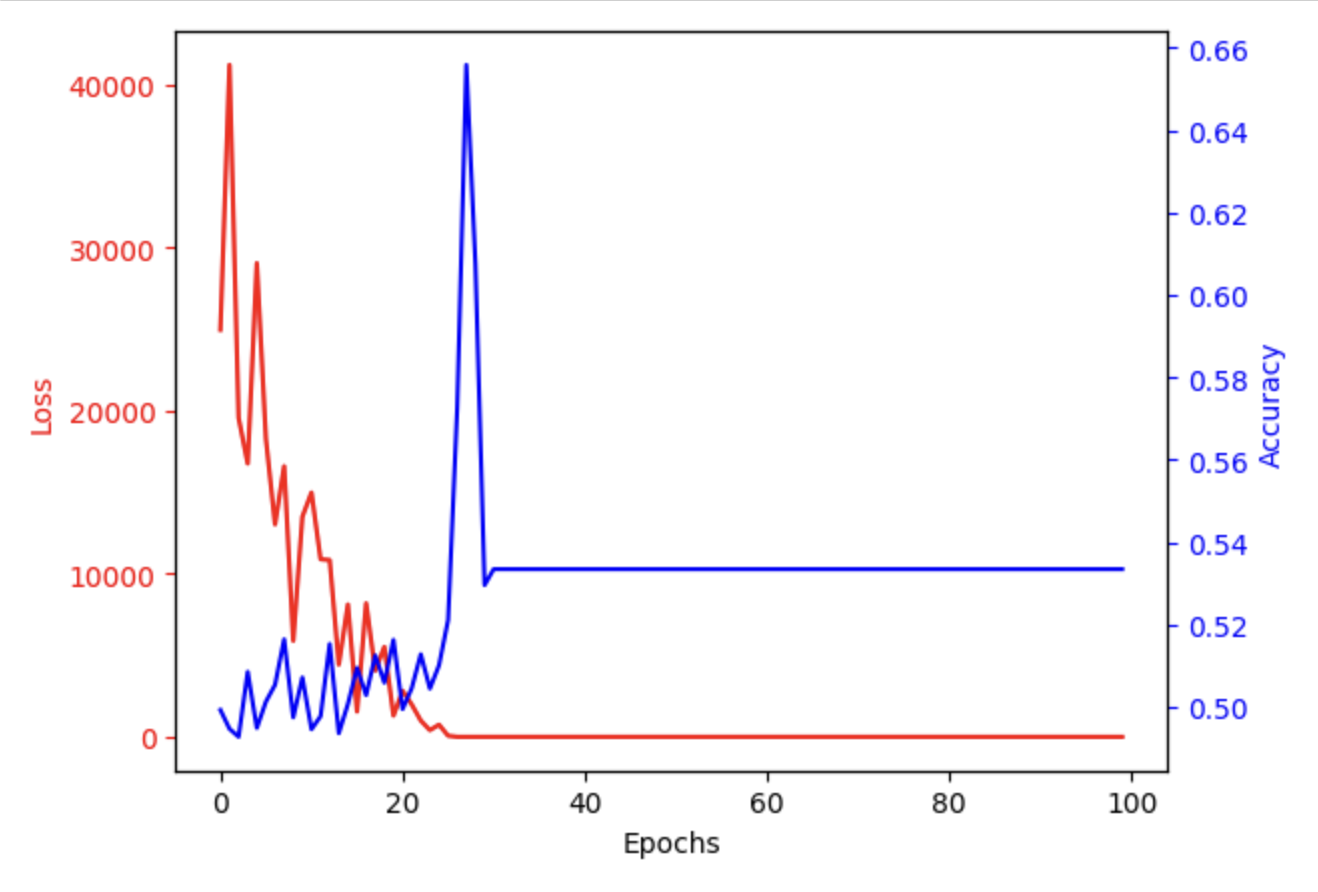
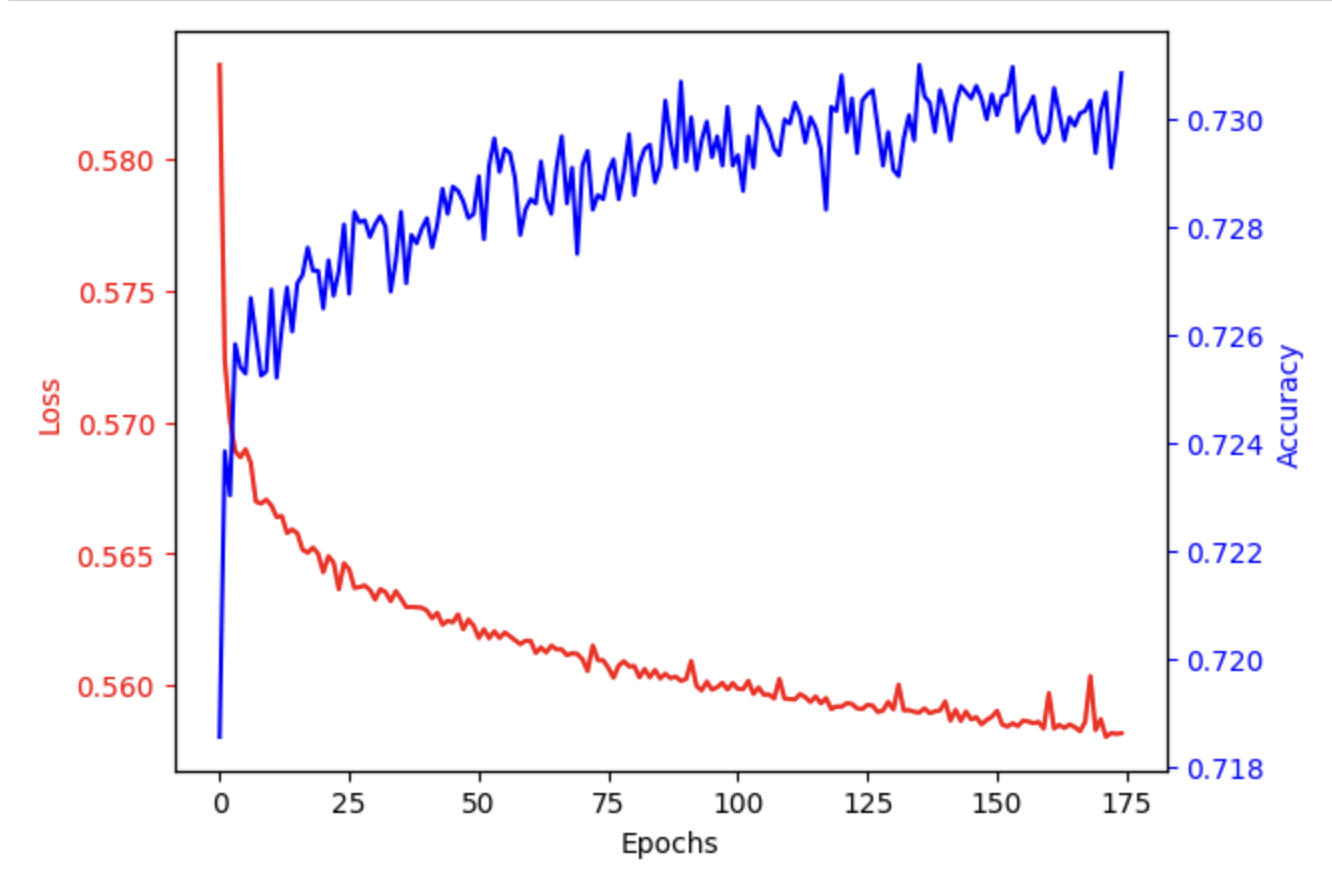
Alphabet Soup is a nonprofit agency that funds other ventures. Applicants must complete an application and then get approved in order to receive a grant. Alphabet Soup is interested in a tool that can predict whether or not an applicant may potentially receive funding based on key factors, or characteristics, about the application and applicant. To that end, they provided over 34,000 applications so that I can build a machine learning model, using a deep learning, neural network methodology, to predict potential success or failure.

RESULTS

Data Preprocessing

* The target for the model is the variable that indicates whether or not an application is successful identified in the column “IS\_SUCCESSFUL”.
* The features of the model are all of other variables that could potentially affect the outcome of the application, or target value. Namely Application Type, Affiliation, Classification, Use Case, Organization, Status, Income Amount, Special Considerations, and Ask Amount.
* I originally removed the EIN and NAME from the feature set since, theoretically, they shouldn’t have an impact on the application outcome. During optimization I removed Classification to try to simplify the model since there were many unique discrete variable sets.

Compiling, Training, and Evaluating the Model

* To set up the initial neural network model I used 2 hidden layers and 1 output layer. The first hidden layers used 80 neurons, layers and the ReLu activation functions, the second layer used 30 neurons and the ReLu activation function, and the output layer used 1 neuron and the Sigmoid activation function. These were selected to match the starter code suggestion. Multiple iterations of different activation functions, # of neurons, and using a single hidden layer were used during optimization and the biggest improvement was from using the Sigmoid activation function for all three layers. The Tensorflow Playground was used to try to understand the impact on the model by changing the model parameters. However, a more experienced working understand of the process would help in optimizing these tuning functions.
* The original target model accuracy results for the model tuning was 53.3% and the testing accuracy was 52.9%   
    
  I was able to significantly improve the results of the model and come close to achieving the target model accuracy for the model tuning at 73% and a testing accuracy of 71.7%  
    
    
  Loss was also significantly improved to below 0.72 from around 10,000.
* The model performance improvements were accomplished with a few changes. The iterations in model tuning parameters produced minimal improvements. So I began making changes to the features. Classification was removed since it had many unique discrete variable sets and I wanted to simplify the model. Second, I wanted to clean and scale the Ask Amount numeric parameter since it was important but out of scale from the rest of the parameters. I used to the standardscaler function to do this on its own forgetting that I had used standardscaler on all of the features later in the code. But this actually significantly improved the model so I kept it in. I also deleted the Status rows since there were only 5 of them out of over 34,000 and I assumed they weren’t critical. I further removed the Status as a feature to simplify the model. All of these changes helped improve the model, in sum.

SUMMARY

The deep learning neural network model was fairly successful at around 73% accuracy and resulting in 0.72 of loss with the predictions. With more time, it would help to do three things to improve this model:

1. Investigate if there are any outliers or clear mistakes with each of the features, especially the numeric ask\_amount feature. Then clean those outliers or delete the rows.
2. Use techniques to understand which features most affect the results of the labeled value, or approval. This could be used to further simplify the model. Also, a better understand of the fundamental rules for how funding applications are evaluated for approval could also provide insight for the features affect the model. This apriori knowledge could also simplify the model.
3. Better understand how the tuning parameters work or iterate the model tuning among them.

In summary, there is plenty of data in this set and the parameters are appropriate and logical for using deep learning to predict the funding outcome. The application for machine learning to the data set should result in good performance of the model. With more time and experience with the dataset and neural network tools, I think this model could achieve over 90% accuracy.