**Deep Learning and Neural Networks**

**Overview of the Analysis**

*Purpose*

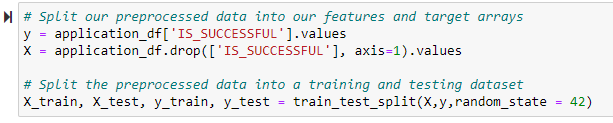
The purpose of this project was to create a Deep Learning Model that would help analyze whether or not applicants for funding from the non-profit organization Alphabet Soup would be success. For this model, 9 features from the charity data set were used. Tensorflow Keras Sequential model with multiple Dense hidden layers and a binary classification output layer was used for optimization by changing the parameters including epochs, activation functions, batch size, learning rate, and number of input features.

**Results**

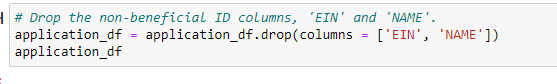
*Data Processing*

The data from the charity\_data.csv was preprocessed to enhance the quality of the data. The data was cleaned, scaled, compiled, and trained before going through a neural network model.

* The IS\_SUCCESSFUL column was considered the target for the model
* The APPLICATION\_TYPE, AFFILIATION, CLSSIFICATION, USE\_CASE, ORGANIZATION, INCOME\_AMT, and ASK\_AMT columns were the features of the model

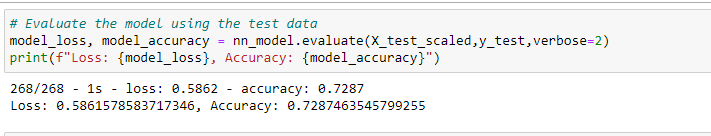


* The EIN and NAME columns were neither targets nor features, and were removed from the input data

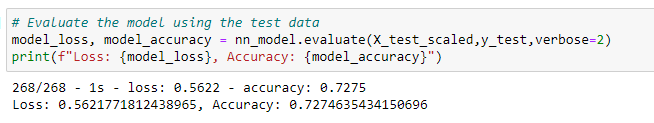


*Compiling, Training, and Evaluating the Model*

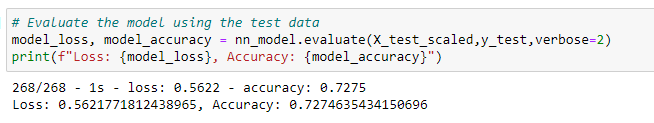
* I initially chose 80 neurons with a Relu function for my first layer, 30 neurons with a Relu function for the second, and 1 neuron with a sigmoid function for the outer layer. After training and compiling the model, I was only able to achieve an accuracy of approximately 73% and was not able to achieve the target model performance of 75%.
* To try to increase the model’s accuracy I initially increased the epoch from 100 to 200, but that still gave an accuracy of 73%.



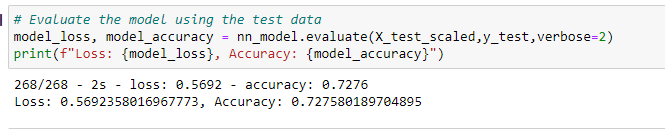
* The next attempt was to change the activation and made all the layers sigmoid with an epoch of 200, which still kept it at approximately 73%.



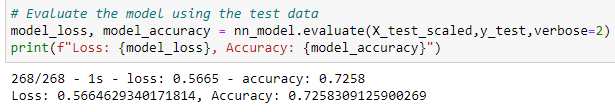
The third attempt was adding a 3rd layer with a neuron of 7 and relu and epoch of 100, but the accuracy was still at 73%, so no change from previous test.



Next attempt I increased the epoch to 200 and the accuracy did not change



The final test I ran was changing all the activations to sigmoid and keeping the epoch at 200, which resulted in the accuracy remaining at approximately 73%.



**Summary**

After multiple tests including changing the epochs, activations, and adding another hidden layer, there was no change to the success rate. The accuracy of the model I optimized for predicting whether a donation is successful ended up being approximately 0.74 and its loss metric had a small decrease from 0.58 to 0.56, but otherwise no significant change after attempting to change the model to optimize results. In the end, the model did not reach the target model performance.

**Recommendation**

To better optimize the model, I could’ve done other tests including removing additional features during the preprocessing step, and possibly increasing the neurons when defining the model. Alternatively, the best model for this most likely would’ve been using a Random Forest Classification Model. Using this model would’ve been the optimal solution because it would’ve chosen the important features and would be able to run a large dataset. This model also typically works well for binary classification. This model is also random when it builds trees and therefore uses its predictiveness to make accurate decisions.