**Report – Alphabet Soup Charity Analysis**

**Overview of the Analysis**

* Explain the purpose of the analysis.

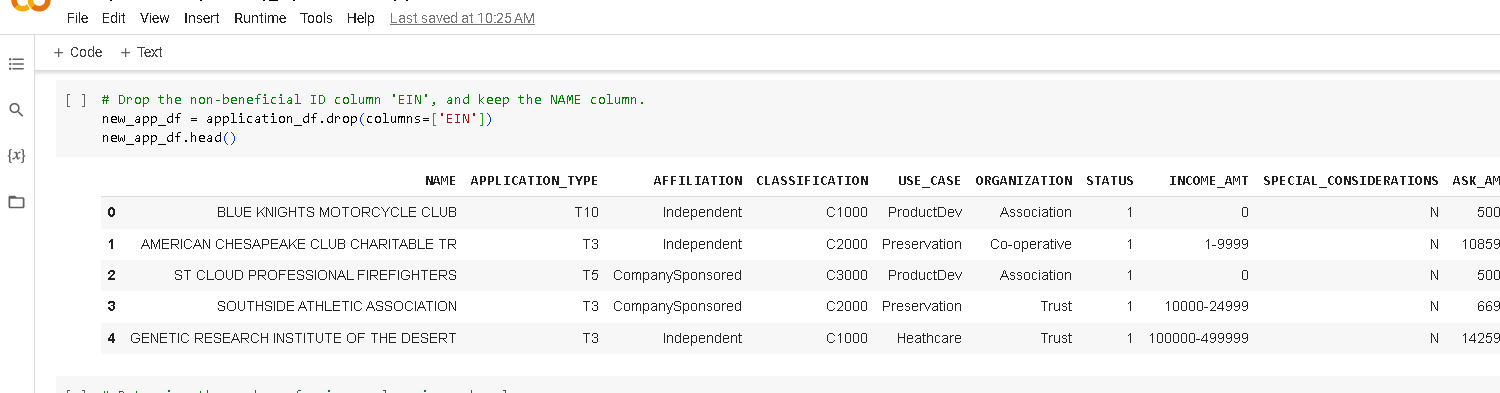
The objective of the analysis is to evaluate the performance of deep learning neural networks in predicting applicants’ success if funded by Alphabet soup.

**Results**

Using bulleted lists and images to support your answers, address the following questions:

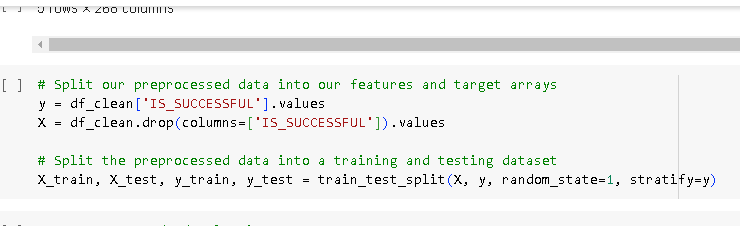
* Data Preprocessing
  + What variable(s) are the target(s) for your model?

The target variable used for the model is IS\_SUCCESSFUL.



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

The features for the model are NAME, APPLICATION\_TYPE, AFFILIATION, CLASSIFICATION, USE\_CASE, ORGANIZATION, STATUS, INCOME\_AMT, SPECIAL\_CONSIDERATIONS, ASK\_AMT.



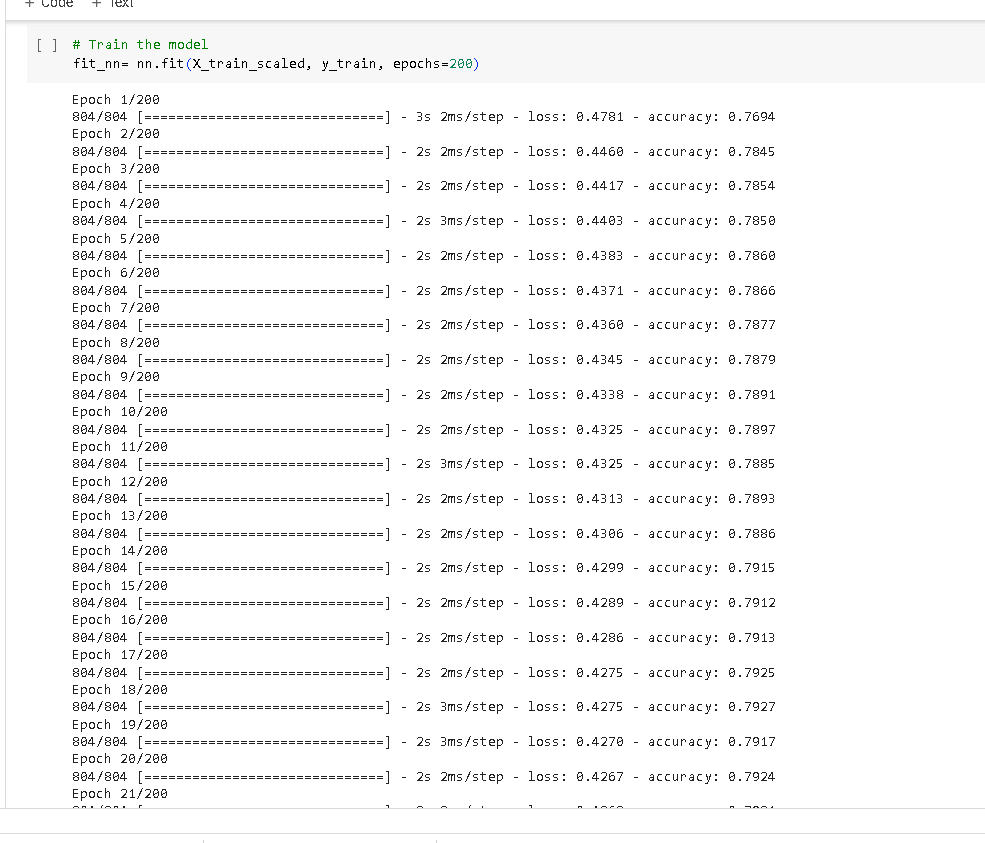
* + What variable(s) should be removed from the input data because they are neither targets nor features?

According to the multiple attempts in achieving the target accuracy, the STATUS, and SPECIAL\_CONSIDERATION columns are neither targets nor features because the difference type count is huge and had no impact on accuracy. But I left them in place since removing them makes no difference.

* Compiling, Training, and Evaluating the model.
  + How many neurons, layers, and activation functions did you select for your neural network, and why?

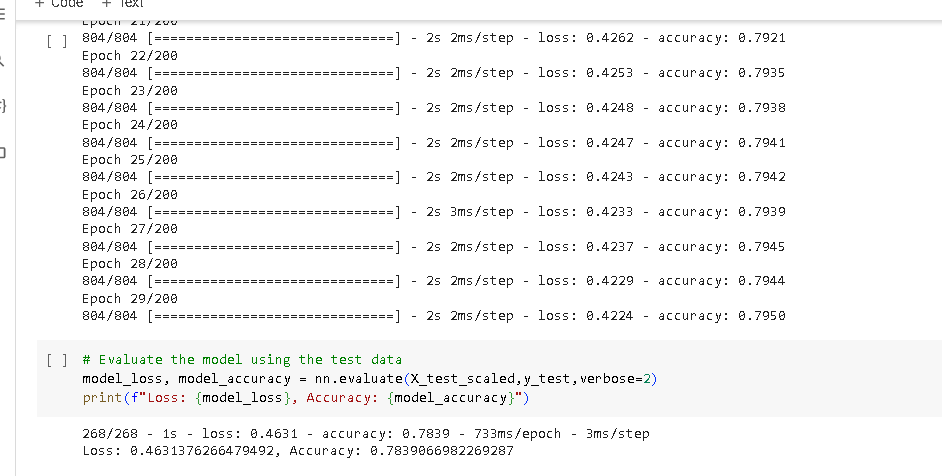
For the optimization of model, I selected three hidden layers with a total of 180 neurons. The first layer had 100 neurons with relu as the activation function, the second layer had 50 neurons with tanh as activation function, and the third layer had 30 neurons with sigmoid as activation function. The output layer remains one neuron as usual with sigmoid as activation function. The multiple layers increase the capacity, and number of neurons helps control the complexity of the model, both actions prevent under fitting and thereby gives less errors. The purpose of using multiple activation functions is for each layer to build up on the result of preceding non- linear layer and which then leads to a complex non-linear function that’s able to approximate every possible function with the right weight and depth, and provides a better outcome.





* + Were you able to achieve the target model performance?

Yes, I achieved an accuracy of 0.78.



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

1. I added back the NAME column (Drop fewer column).
2. Added more neurons to the hidden layers (1 &2).
3. Added extra hidden layer (Third hidden layer).
4. Used different activation functions for each hidden layer (relu, tanh, sigmoid).
5. Increased the number of epochs to the training regimen (From 100 to 200).

**Summary**

Summarize the overall results of the deep learning model. Include a recommendation for how a different model could solve this classification problem, and then explain your recommendation.

Overall, the accuracy of the deep learning model prior to optimization is poor compared to with optimization which means the NAME column is super relevant and the use of more hidden layers, neurons, and different activation function for each layer improves prediction accuracy.

I would recommend the use keras-tuner to solve the classification problem because it eliminates the guessing game and the model decides on the best activation function, number of neurons and layers to use to achieve the best accuracy.

.