# Alphabet Soup Analysis

By Carlos Ruiz

## Overview of the Analysis

The purpose of this project is to build and refine a neural network model capable of predicting the success of charity donations.

## Dataset

Charity_data.csv provides various features about organizations and donations, and target variable we are aiming to predict.

Variables in the dataset included:

- AFFILIATION: The type of affiliation the organization had.
- CLASSIFICATION: The classification type of the charity.
- INCOME_AMT: The income range of the charity organization.
- ASK_AMT: The amount requested in donations.
- IS_SUCCESSFUL: The target variable, indicating whether the donation request was successful (1) or not (0).

To ensure the donations remain anonymous, the following variables are removed from the analysis:

- EIN: Employer Identification Number of the company giving the donation.
- NAME: Name of the company giving the donation.

## Insights into the Data:

The label variable, IS_SUCCESSFUL has 18261 successful and 16038 unsuccessful donations. This makes this variable balanced, which will help the predictions.

To improve performance on the model and reduce the complexity of some variables, CLASSIFICATION was reduced to a maximum of 12 different categories by introducing a category called Other.

Notice that AFFILIATION and USE_CASE already had a category of Other in the original dataset; therefore, they didn't need to be changed.

SPECIAL_CONSIDERATIONS had values of "Y" and "N". To help the process of machine learning, these values were changed to "1" and "0" correspondingly.

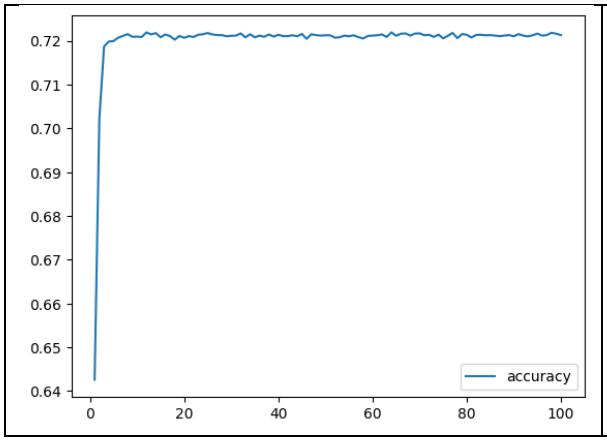
## Stages of Machine Learning Process

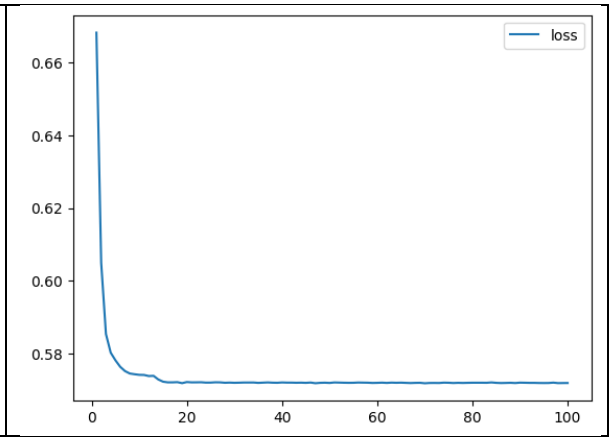
Tensorflow was selected to exercise machine learning. Beginning with a simple model and gradually increasing complexity is a good approached based on the dataset. Following this process, we could improve accuracy and reduce loss.

## Results

Model 1:

- Description: Starting with a simple model, the first model is a perceptron with an activation of relu. Starting with 100 epochs.
- Accuracy: ~72.023%
- Loss: ~57.354%
- Observations: Perceptron is returning good results. Data is not under or over fitting.
- Next step: Add more neurons, keeping a single layer to see the effects.

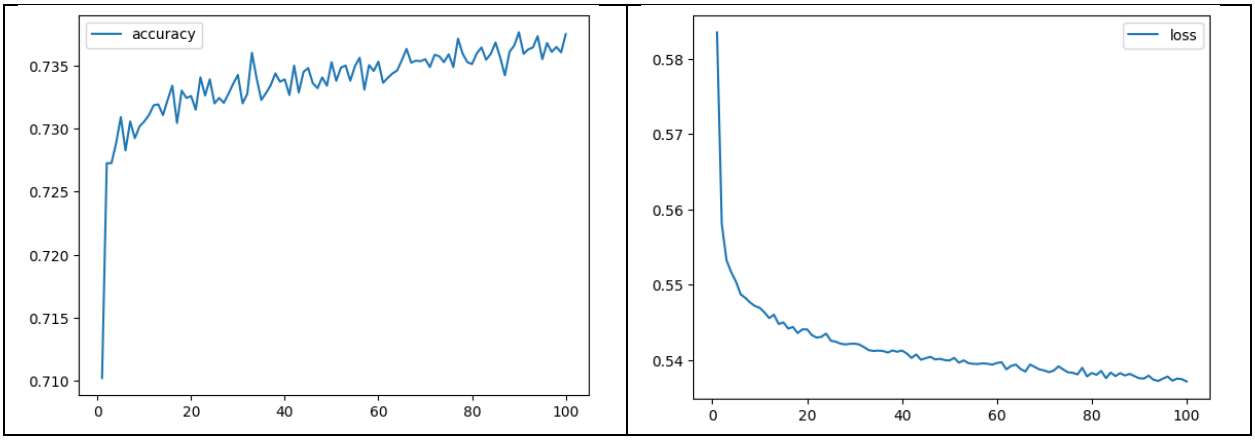



Model 2:

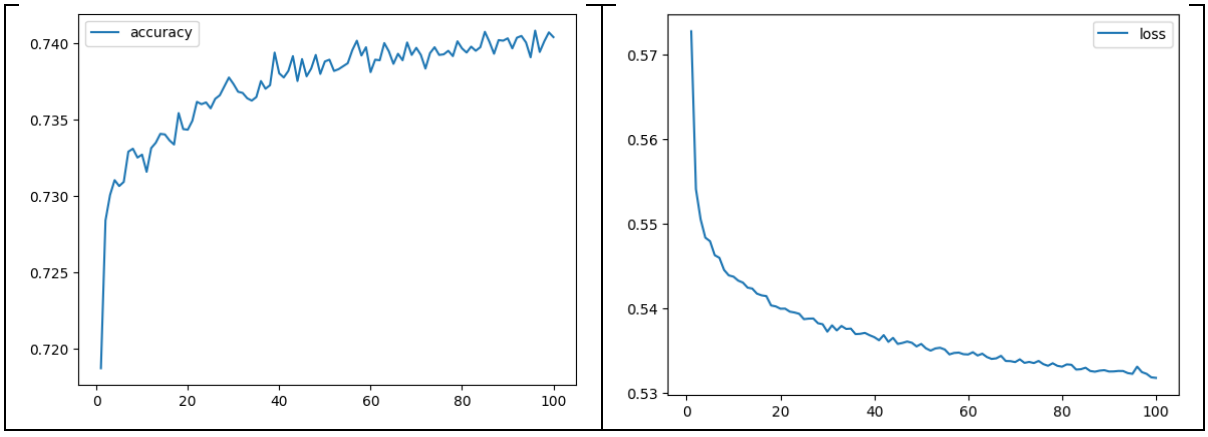
- Description: Single layer with 50 neurons. Keeping activation relu and 100 epochs
- Accuracy: ~73.120%
- Loss: ~55.363%

- Observations: Model 2 showed improvements in accuracy and loss.
- Next step: Add a second layer with 25 neurons.



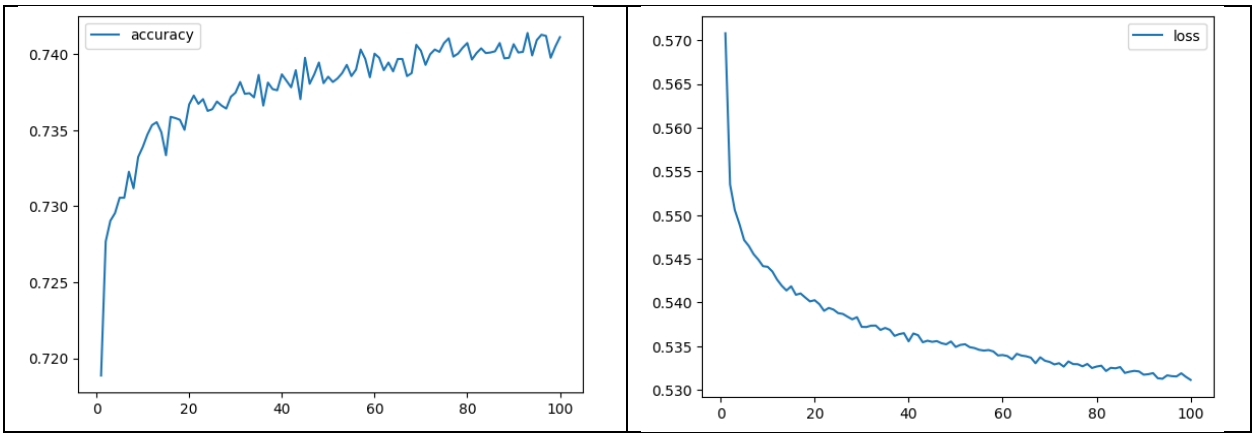

Model 3:

- Description: Two-layer model (50 and 25 neurons), with activation relu and 100 epochs.
- Accuracy: ~73.085%
- Loss: ~55.367%
- Observations: Accuracy showed marginal improvements, but Loss decrease slightly. Model 2 is better than Model 3.
- Next step: To validate the lack of improvement, try two layers of 50 each, with an activation of relu and 100 epochs.



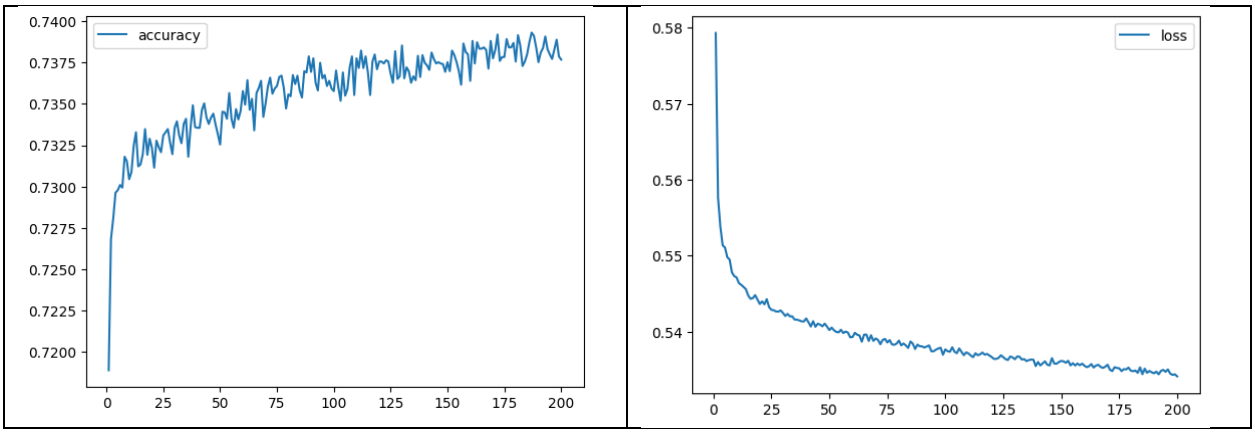
Model 4:

- Description: Two-layer model 50 neurons each, with activation relu and 100 epochs.
- Accuracy: ~72.968%
- Loss: ~55.806%
- Observations: No improvements at all. Model 2 remains as the best model so far.
- Next step: Since improvements by adding layers or neurons are marginal, go back to a single layer with 50 neurons with an activation of relu. However, add more training to see if performance is improved, or the model will overfit.



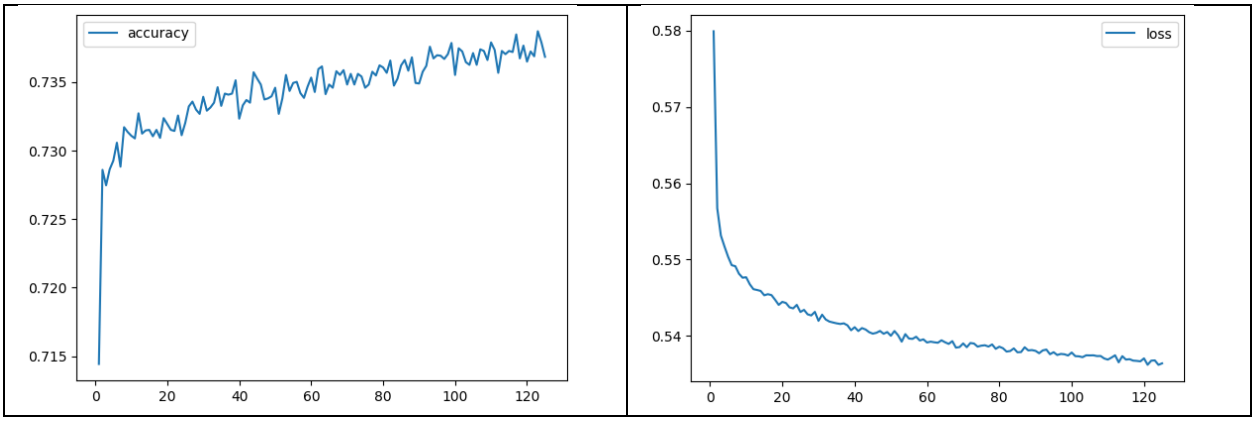

Model 5:

- Description: Two-layer model 50 neurons each, with activation relu and 200 epochs.
- Accuracy: ~72.653%
- Loss: ~55.673%
- Observations: No improvements at all. Model 2 remains as the best model so far. However, the model more or less plateaued at about 100 epochs, and Loss showed slight improvements with the extra epochs. Increasing epochs can be beneficial; however, the improvements are very marginal, not enough to justify the extra processing power.
- Next step: The final recommendation is to run a single layer with 50 neurons with activation of relu and 125 epochs.




Model 6

- Final model
- Accuracy: ~73.166%
- Loss: ~55.619%




## Summary

Based on the results, Model 06 performed the best without overfitting, achieving an accuracy of ~73%, with a loss score of 56%. This project indicates that increasing the number of neurons above 50 and/or increasing layers above 1, only affects the model marginally.

## Recommendation:

Further analysis has been undertaken on Starter_Code_extra_analysis to challenge the theory explained on this document. The extra analysis runs the same dataset through a variety of models, including more complex models, as well as using tanh activation. This extra analysis validates the summary of this document.