## Overview of the analysis

I completed this analysis on data from the Alphabet Soup nonprofit foundation. They receive and assess applications for funding, and have collected data on previous applications and the organisations that submitted them, as well as whether or not the funded ventures were successful or not.

I did this analysis to design a tool that would help Alphabet Soup to predict which future applications are most likely to succeed and are therefore the best prospects to provide funding to.

The aim was for this analysis to develop a model with 75% accuracy.

## Results

**Data Preprocessing**

**What variable the target for your model?**

The target variable for this model was the ‘IS\_SUCCESSFUL’ column, which indicates which applications for funding resulted in a successful proposition. This column has binary inputs of 0 for propositions that were not successful and 1 for successful propositions.

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

The feature variables for this model were the remaining columns:

* APPLICATION\_TYPE
* AFFILIATION
* CLASSIFICATION
* USE\_CASE
* ORGANIZATION
* STATUS
* INCOME\_AMT
* SPECIAL\_CONSIDERATIONS
* ASK\_AMT

**What variables should be removed from the input data because they are neither targets nor features?**

The EIN and NAME variables were removed because they are neither targets nor features and are therefore not useful for the modelling.

**Compiling, Training, and Evaluating the Model**

**How many neurons, layers, and activation functions did you select for your neural network model, and why?**

I started with 1 output layer and 2 hidden layers with 10 on the 1st and 20 neurons on the 2nd. I used the ‘sigmoid’ activation function for the output layer – because this is a model based on a binary variable of 1 or 0, successful or not successful. And I used the ‘relu’ activation function for the hidden layers.

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

Yes, I got to 75% accuracy by the 3rd attempt.

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

I focused on the neurons and hidden layers used in the model. For the second version of the model, I switched the number of neurons on the hidden layers to 20 on the 1st and 10 on the 2nd. Then, for the third version I added a hidden layer and used 30 neurons on the 1st, 10 on the 2nd and 10 on the 3rd.

Summary

The final accuracy score for the model was %, after optimisation. The key differences were…

A good way to further improve the accuracy would be to use a model based on keras automated model optimisation using keras-tuner. This would tap into that keras feature to identify the best possible combination of layers, neurons, and activation functions.