**Report Analysis**

**Overview**

The purpose of this analysis was to use the features within a dataset provided by a funder called “Alphabet Suit” to determine which organisations, over the years, were likely to be successful based on the given features. The dataset was prepared in order to create binary classifiers for each features such as classification type, amount requested, type of organisation and so in order to predict whether applicants would be successful if funded by Alphabet Soup

**Results**:

* Data Preprocessing
  + The key target variable was “IS\_SUCCESSFUL”.
* Nine features were used. The 9 features variables included the Alphabet Soup application type, the affiliated sector of industry, the Government organization classification, the use case for funding, the organization type, theactive status**,** the income classification, the special considerations for application and finally the amount asked for.
  + Two variables were removed from the input as this information was neither a target nor a feature. The variable removed were the organisation number and unique identification number.
* Compiling, Training, and Evaluating the Model
  + The data was scaled and all values were between 0 to 1, the target variable of “success” was binary and there were no negative. Given this and also given this is not a particularly large data set I therefore initially used the sigmoid activation function for the input and hidden layers of the mode. This model took little time to train, returned an accuracy of 72% with a model training loss of 55%. I therefore tested the model again using a combination of “relu” for input layer and “sigmoid” for hidden layers, this results in similiar accuracy remained at 72% but with reduced model training loss of 50%.
  + I could have used the code we were presented in class to automate the process of choosing activation functions but this was not required in the task.
  + I used 80 nodes for the input layers, 30 nodes for the two hidden layers, and 1 node for the output layer. The output layer only required one neuron as the result was a binary “yes and no” as to whether success was achieved. For the same reason, the “sigmoid” activation function was used for the output layer.
  + I was able to achieve the target model performance of 72%.
  + To increase model performance I increased the number of hidden layers from 1 to 2, and the number of nodes and epochs.
  + **Summary**:

Overall the accuracy of my model was 72% with a model training loss of 50%. I chose this model over the “relu” activation function model as this has less training loss despite the accuracy being similar. I chose the former as it had the least amount of training time.

