**Predicting Applicant Success if funded by AlphabetSoup**

The main goal from the outset of this exploratory analysis revolved around the idea of developing a binary classification (YES/NO) model for AlphabetSoup’s business team that will help them to predict whether organizations that they provide funding for will eventually be successful. Initially, a dataset comprised in a CSV format was provided by AlphabetSoup which contained raw data in the form of various characteristics pertaining to different organizations’ applications for funding. These characteristics included: Identification columns, application type, sector of industry of organization, government organization classification, use case for funding, Organization type, present status, income classification, special consideration for application, funding amount requested, and then if the organization was successful in using the funding amount.

As part of the data pre-processing step in the production of a model, it required digging down into the data first to explore what had been provided from AlphabetSoup’s business team. More will be spoken on this shortly. At the crux of building a predictive model is the process of splitting the data between what I want the target to be (the value we want our model to predict for the end user) and what I want the features to be (the determinants resulting in end prediction). As for the target, the purpose of the model is to predict the success of a funding initiative so from the available data the column “IS\_SUCCESSFUL” was isolated as the target. During my first attempt at producing a model, I followed the directions as provided in the assignment and included all columns besides “NAME” and “EID”. This means that the features I was left with were: “application type, sector of industry of organization, government organization classification, use case for funding, Organization type, present status, income classification, special consideration for application and funding amount requested.

In my first attempt at producing a model, I only included one hidden layer and in that hidden layer set the number of units to five. I wanted to see what result I would get for model accuracy by developing a basic neural network model at first so then on my subsequent models I would be able to build from the basic one. From this initial model, I achieved an accuracy score of 0.7223 which is a decent all-around score, but I wanted to record at least a 0.75 score by the end of my model exploration.

Eventually on my third model creation, I did manage to eclipse the target in the activity prompt with an accuracy score of 0.7516 {4dp). In the pre-processing of data step, I decided to explore the distribution of values within each individual column before deciding to drop any specific ones. From this idea, I saw that the ‘NAME’ column had a large amount of variance in values, so therefore to decrease the amount of “noise”, I decided to bin the values and as a result create a cell value labelled as ‘other’. When it came to building the model, I included three hidden layers with the first one starting with 25 units and each following layer had 5 units less than the others. My thought with this was to have the model be created based on a more extensive number of nodes and layers to then generate a better accuracy score due to the data the model being trained and tested on being looked at and filtered through on a deeper level.

From seeing that the accuracy score I achieved in my best model was 0.75 (dp), I feel that this result is decent but should not be relied upon too heavily by AlphabetSoup. It’s always important to zoom out and think about what the accuracy score of a model one has created is in fact predicting. In this case, AlphabetSoup wanted a model that would be able to predict whether organizations that they provide funding for would be successful. Thus, this model is being created to help AlphabetSoup attempt to better justify the decisions they make with their capital regarding funding a given organization. Obviously, there are a variety of other factors (human interaction through interviews, organization’s values/history, etc.) that AlphabetSoup would take into consideration before they provide funding for an organization. For this reason, I don’t believe the model with an accuracy score of 0.75 would be that significant for AlphabetSoup.

Seeing that the question that is being asked of this mode to predict is one of classification (Is Successful vs. Is Not Successful) and we have labelled data in the form of our raw csv file, I believe that creating a Random Forests Classification Model would be interesting to see what accuracy score the model yields. A Random Forests Model combines the results from a massive amount of decision trees that are made when the data is being explored during the training phase in order produce the classification result (conglomerate result from all the decision tree results) that is intended. A deep learning model can be more beneficial for more complex and extensive data, but in this case a Random Forests Model may work well because the data is labeled well and understandable.