**Neural Network Model Report**

**Overview**

The purpose of the analysis was to create a predictive model that would determine which applicants for non-profit funding ventures were the most likely to succeed. The data centered around data points which included: Application Type, Organization Type, Income Amount, and Funding required.

**Results**

The target for the analysis was a binary value, IS\_SUCCESSFUL. The features initially consisted of 36 fields centering around the application types, organizations, as well as income and funding ranges. The name and ID numbers for the organizations applying were removed as they were not salient data points for any analysis conducted.

A total of 6 Neural Network analyses were conducted utilizing various combinations of layers, neurons, and activation functions. The first five were conducted in the hopes of differentiating which values for these fields might produce better results from which a higher performing model could be built upon.

The first five models were conducted using all 36 fields in the data set and showed limited success. The accuracy ranged from as low as .534 to .620, with loss ranges quite steady from .683 to .691. Both of those values fell short of target values for a predictive model.

A decision was made to employ a supervised learning model (Random Forest). This model was chosen primarily to identify the specific features which correlate more highly with the target value the model is trying to predict. The Random Forest model itself was much more accurate with an accuracy value that came in at .72, easily besting the initial Neural Network models.

After determining the importances of the features, the top 10 features were selected for another Neural Network model. A two layer model with a total of 15 neurons was implemented and the higher accuracy did carry over with an accuracy of .726 and an improved loss of .566 as well.

Finally, a keras-tuner was run to see if an optimal model setup could be found to improve the accuracy and loss values could be found. The best set of hyper parameters came in an .726 accuracy which was a decided improvement over the initial models and represents a pretty solid accuracy rate overall for the model given the size and structure of the data provided.

**Summary**

The results clearly demonstrate that the data as initially provided was not sufficient for a Neural Network model that would provide the requisite level of accuracy. Only after the features were reduced to the top 10 most highly correlated did the accuracy level even approach the level sought.

It should be noted that even the top 10 most highly correlated features reduced the coefficient to .15, so even the top features did not represent particularly strong correlations. The overarching recommendation from this analysis would be to continue to expand and analyze the data itself as the models were prone to potential underfitting in the first several incarnations of the model.

It was only when many of the features that were not at all correlated to the target were removed, thus removing the irrelevant noise, that the model even began to approach the accuracy levels set out as the goal for the model. One would suspect that continued tweaking of these data fields might continue to improve accuracy and decrease loss.

As of yet, no changes have been made to the bins for budget and income which might yield results in fine tuning the performance of the model. With the utilization of the keras-tuner, one might reasonably believe that sufficient attention was paid to variations in Neural Network protocols. I don’t believe the answer lies in a Neural Network setup that has not been tried.

A model with an accuracy level of better than 75 percent would appear to be possible, but continued efforts surrounding the data, specifically the quantity and quality of the features implemented will need to be investigated further.