1. **Overview** of the analysis:

The purpose of this assignment was to create an algorithm to predict whether or not applicants for funding from a non-profit will be successful. We built and tested a neural network to create a binary classifier that is capable of predicting whether applicants will be successful if funded by Alphabet Soup. We were provided with the following fields in the csv:

**EIN** and **NAME**—Identification columns

**APPLICATION\_TYPE**—Alphabet Soup application type

**AFFILIATION**—Affiliated sector of industry

**CLASSIFICATION**—Government organization classification

**USE\_CASE**—Use case for funding

**ORGANIZATION**—Organization type

**STATUS**—Active status

**INCOME\_AMT**—Income classification

**SPECIAL\_CONSIDERATIONS**—Special consideration for application

**ASK\_AMT**—Funding amount requested

**IS\_SUCCESSFUL**—Was the money used effectively

1. **Results**:

* Data Preprocessing
  + *What variable(s) are considered the target(s) for your model?*

The target for this model was IS\_SUCCESSFUL.

* + *What variable(s) are considered to be the features for your model?*

The features of this model are the Application Type, Affiliation, Classification, Use Case, Organization type, Status, Income Amount, Special Considerations, and Ask Amount.

* + *What variable(s) are neither targets nor features, and should be removed from the input data?*

The EIN and Name fields were removed from the dataset as they did not contribute any useful information to the model.

* 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 two layers (excluding the output layer) with ten nodes each and I ran it through 100 epochs. This resulted in a 72.96% accuracy score, a bit lower than the target. The first round of optimization, I decided to double the nodes to 20 and increase the epochs to 125. The accuracy score only increased marginally to 73.25%. The second round of optimization, I increased the nodes and epochs again, this time to 50 and 150, respectively. The accuracy score is still consistent with the previous attempts, coming in at 73.38%.

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

The target of 75% was not achieved, though it did come close.

* + *What steps did you take to try and increase model performance?*

I increased the number of nodes on each iteration, going from 10 nodes, to 20, then 50. I also increased the number of epochs to give the model a chance to increase its accuracy as much as possible.

1. **Summary**: Summarize the overall results of the deep learning model. Include a recommendation for how a different model could solve this classification problem, and explain your recommendation.

Overall, the model did not quite succeed, but it was close. I would likely continue to experiment with adding additional hidden layers as increasing nodes/epochs did not make much of a difference. If I were to choose a different model to use, I think a Random Forest would be a good option. It takes less training than a neural network and reduces data confusion. It would also allow you to tweak and refine your input fields more to see which have the greatest impact on the prediction results.