Report on Neural Network Model for Alphabet Soup Data

Alexandra Pfleegor

Overview of the analysis: Explain the purpose of this analysis.

In order for Alphabet Soup to choose who to approve, they want to know which ventures are most likely to succeed using the money provided. Thus, a model that can predict whether the venture will succeed can help them in this choice. The analysis of the model is key because it will help determine whether Alphabet Soup should use this to predict success or not.

Results: Using bulleted lists and images to support your answers, address the following questions.

Data Preprocessing

- What variable(s) are the target(s) for your model?
  - The target variable is the boolean is_successful determining whether the venture was successful or not.
- What variable(s) are the features for your model?
  - The features for the model are application type, affiliation, classification, use case, organization, income amount and ask amount.
- What variable(s) should be removed from the input data because they are neither targets nor features?
  - I removed EIN, name, status and special considerations. EIN and name were identification variables, which we want to avoid with neural networks. The status and special considerations columns had so few data points of 0 (not active) and Y (there is a special consideration) that the neural network would not be able to use these features to accurately predict success.

Compiling, Training, and Evaluating the Model

- How many neurons, layers, and activation functions did you select for your neural network model, and why?
  - For my final neural network model, I used 2 hidden layers with 81 nodes and 11 nodes respectively. I used the tanh activation function for all of the layers except for the output layer that used the sigmoid function for its activation function. I used these specific numbers after using a keras tuner to find the best model.
- Were you able to achieve the target model performance?
  - I was able to achieve the target model performance. However, the methods that I used in order to get there means that the model might not be able to generalize to data points who request more than $10,000. Shown on the next page is a screenshot of the model summary and the results. I also decreased the loss from my original model by 0.04.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_5 (Dense)             (None, 81)                2511

 dense_6 (Dense)             (None, 11)                902

 dense_7 (Dense)             (None, 1)                 12

=================================================================
Total params: 3,425
Trainable params: 3,425
Non-trainable params: 0
```

```
203/203 - 0s - loss: 0.5256 - accuracy: 0.7507 - 313ms/epoch - 2ms/step
Loss: 0.5255534052848816, Accuracy: 0.7507322430610657
```

- What steps did you take in your attempts to increase model performance?
  - I dropped the status and special considerations columns.
  - I created a bin for the organization column labels that had a value count of less than 500.
  - I also created a bin for the use case column labels that had a value count of less than 500.
  - Additionally, I created a bin for the affiliation column labels that had a value count of less than 500.
  - I replaced the four labels classifying income over a million dollars to just one label called “1M+”.
  - I used the interquartile range of the original data for the ASK_AMT column to identify outliers. However, this assumed a normal distribution, so was probably not the best method statistically.
  - I used a keras tuner 5 times throughout the process to look for the best hyperparameters for the neural network model.

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

Overall, the deep learning model uses 2 hidden layers and the tanh function in order to get 75.07% accuracy with a loss of 0.5256. However, this model will only be accurate for funding requests of under $10,000. I would recommend splitting the data by the amount of money requested in a future model. The difference between people who receive $10,000 and $1,000,000 could be potentially different enough that it warrants 2 or 3 different classification models. There is also enough data for this segmentation. Thus, a different deep learning model could be created to deal with the larger amounts of funding requested.