**[deep-learning-challenge](https://github.com/randallwvr90/deep-learning-challenge/blob/main/README.md" \l "deep-learning-challenge)**

**Deep learning Challenge for the Data Science Bootcamp!**

The purpose of this analysis was to create a deep learning model for Alphabet Soup, a charitable foundation, to predict whether applicants are likely to be successful if they are funded. The goal is to efficiently allocate resources by identifying applicants most likely to succeed, thereby maximizing the impact of the foundation's grants.

Results:

Data Preprocessing:

Target Variable: The target variable for our model is the "IS\_SUCCESSFUL" column, which indicates whether an applicant's funding request was successful (1) or not (0).

Features: Features used in the model include various data points related to applicants, such as "APPLICATION\_TYPE," "AFFILIATION," "CLASSIFICATION," "USE\_CASE," and many others.

Variables to Remove in step\_1: Variables like "EIN" and "NAME" were removed as they do not provide meaningful information for the prediction task.

Variables to Remove in step\_2: Variables like "EIN", "NAME", “STATUS” and “SPECIAL\_CONSIDERATIONS.”

were removed as they do not provide meaningful information for the prediction task.

Compiling, Training, and Evaluating the Model:

Neurons, Layers, and Activation Functions: We selected a deep learning model with multiple layers and varying numbers of neurons. The specific architecture is determined through experimentation and evaluation of model performance. Activation functions like ReLU and sigmoid were chosen for hidden and output layers based on their effectiveness in capturing non-linear relationships and performing binary classification tasks.

Step\_1:

* I had 2 layers, layer\_1 had 80 neurons , layer\_2 had 30 neurons.
* used two different activation functions - rectified linear units (relu) and sigmoid.

See the image below for details of the model construction.

A screenshot of a computer

Description automatically generated

The model\_1 exceeded the target performance of 72%.

See the image below:

A screenshot of a computer code

Description automatically generated

Step\_2:

* I had 5 layers, layer\_1 had 300 neurons , layer\_2 had 200 neurons, layer\_3 had 100 neurons, layer\_4 had 50 neurons and layer\_5 had 10 neurons.
* Used two different activation functions - rectified linear units (relu) and sigmoid.

See the image below for details of the model construction.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generatedUnfortunately, The Optimization of Model model\_1 exceeded the target performance of 73%.

See the image below:

Target Model

Performance: The target model performance is typically defined by a combination of factors, including accuracy, precision, recall, and 72%. We aimed for a high 75%, as it balances precision and recall and is suitable for imbalanced datasets.

Steps to Increase Model Performance: To increase model performance, we adopted various strategies, such as:

Feature scaling and normalization to ensure all features have similar scales.

Adding more neurons or layers to the neural network.

Summary:

The deep learning model developed for Alphabet Soup demonstrates promising results in predicting the success of funding applicants. However, the performance metrics may vary based on the specific dataset and hyperparameter settings. The model can be further improved by fine-tuning hyperparameters and exploring more complex architectures.

Recommendation for a Different Model:

A different approach to solving this classification problem could involve using an ensemble model, such as a Random Forest or Gradient Boosting, in combination with a deep learning model. Ensemble models can provide robust predictions by combining the strengths of multiple models. For instance, a Random Forest model can capture complex non-linear relationships and interactions in the data, while the deep learning model can handle feature engineering and extraction. By combining both approaches, we may achieve a more accurate and reliable predictive model for Alphabet Soup's funding applicants.