Neural Network Model Report for Alphabet Soup

By Ryan Denninger

Introduction

In this analysis, we embark on an exploration to leverage deep learning techniques for predicting the success of funding applications submitted to Alphabet Soup, a philanthropic organization. Through the years, Alphabet Soup has contributed to various projects and initiatives, and with a large dataset of past applications, the goal is to streamline the decision-making process. This report delves into the construction, evaluation, and optimization of a neural network model designed to classify applications as likely to succeed or not, based on a set of features.

Purpose of the Analysis

The primary objective of this analysis is to develop a predictive model capable of assessing the potential success of funding applications. By analyzing historical data, we aim to identify patterns and characteristics that correlate with successful funding outcomes. This insight will not only enhance Alphabet Soup's decision-making process but also enable the organization to allocate resources more efficiently, thereby maximizing the impact of its philanthropic efforts.

Data Preprocessing

Target Variable

- 'IS_SUCCESSFUL': This binary variable, indicating the success (1) or failure (0) of funding applications, served as the target for our model.

Features for the Model

- Application Characteristics: Key features including application type (`APPLICATION_TYPE`), organizational affiliation (`AFFILIATION`), classification (`CLASSIFICATION`), use case (`USE_CASE`), and organizational type (`ORGANIZATION`) were identified as predictive inputs.

- Operational Metrics: Additional inputs such as the application's status (`STATUS`), income amount (`INCOME_AMT`), special considerations (`SPECIAL_CONSIDERATIONS`), and the amount asked for (`ASK_AMT`) were also considered crucial for the model.

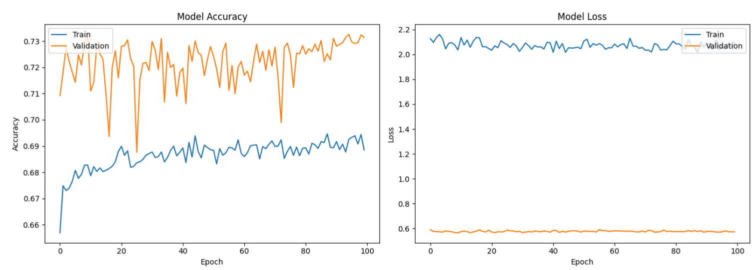
Variables Removed

- Identifiers: The `EIN` (Employer Identification Number) and `NAME` of the organizations were excluded from the dataset as they are unique identifiers and do not contribute predictive value to the model.

Model Architecture and Performance

Initial Model Configuration

- Architecture: The initial neural network consisted of an input layer, two hidden layers with 80 and 30 neurons respectively, utilizing ReLU activation functions, and a sigmoid output layer for binary classification.
- Performance: This configuration achieved a test accuracy of approximately 72%, falling short of the desired benchmark.



Optimization Attempts

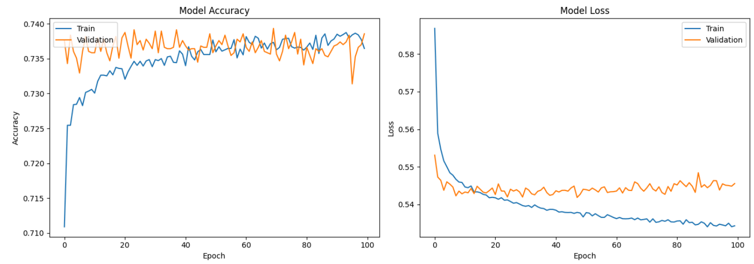
To enhance the model, several strategies were employed:

- Data Adjustment: Binning was refined for `APPLICATION_TYPE` and `CLASSIFICATION`, and `ASK_AMT` was log-transformed to address skewness.
- Model Complexity: Additional neurons and a third hidden layer were introduced to capture more complex patterns in the data.
- Regularization: Dropout layers were added to combat overfitting, a common challenge in deep learning models.
- Training Adjustments: The number of epochs was increased to 100, with a batch size of 200, providing the model more opportunity to learn from the data.

Despite these efforts, the optimized model hovered around a 74% accuracy rate, indicating improvements but still not meeting the 75% target.

Summary of Results

The deep learning model demonstrated potential in predicting funding application success, showcasing the capability of neural networks in extracting insights from complex, multidimensional datasets. While the model did not reach the target accuracy, the optimization efforts underscored the importance of data preprocessing, model complexity, and regularization in enhancing performance.



Alternative Approach

Given the challenges encountered with the neural network model, exploring a ‘Random Forest Classifier’ presents an intriguing alternative. Random Forest is renowned for its versatility, handling both categorical and numerical data effectively. Using the aggregation of predictions from manydecision trees, offers validity to overfitting and provides important metrics, offering valuable insights into which variables most influence funding success.

Why Random Forest?

- Interpretability: Unlike deep learning models, Random Forest offers more straightforward insights into how features influence predictions.
- Handling of Unbalanced Data: Random Forest can be more resilient to imbalanced datasets, a common issue in classification problems.
- Performance: Often, Random Forest requires less tuning compared to deep learning models and can achieve comparable accuracy.

Conclusion

This analysis underscores the potential of machine learning and deep learning in enhancing decision-making processes in philanthropy. While the neural network model offered promising insights, exploring alternative models like Random Forest could provide a balance between predictive performance and interpretability. The journey of refining and optimizing these models is reflective of the complex nature of predictive analytics in real-world scenarios. Through continuous exploration and learning, we move closer to unlocking the full potential of data-driven decision-making in philanthropy.