Kayla Lopilato

Module 21 Challenge

October 2024

Deep Learning Challenge

The goal of this project is to create a neural network that functions as a binary classifier for the non-profit foundation, Alphabet Soup, to predict the success of applicants. This challenge simulates a real-world scenario where organizations attempt to determine which applicants are likely to succeed, helping optimize the selection process.

Data Source:

The dataset, provided by Alphabet Soup, is a CSV containing the metadata of 34,000+ organizations that have received funding. Using this data, the model aims to predict applicant success.

Data Preprocessing:

Target Variable:

- IS_SUCCESSFUL: Indicates whether the applicant achieved the intended outcome.

Feature Variable:

- Remaining columns:
    - APPLICATION_TYPE, AFFILIATION, CLASSIFICATION, USE_CASE, ORGANIZATIONS, STATUS, INCOME_AMT, SPECIAL_CONSIDERATIONS, ASK_AMT

Removed Variables:

- **EIN** and **NAME** were excluded as they are identifiers that don't contribute to the prediction task.

Compiling, Training, and Evaluating the Model:

Three iterations to obtain the highest level of accuracy:

First Attempt:

- 2 hidden layers + 1 output layer
  - Layer 1: 10 neurons, tanh activation
  - Layer 2: 20 neurons, sigmoid activation
  - Output Layer: 1 unit, sigmoid activation, Adam optimizer
- Accuracy: 72.87%

Second Attempt:

- 3 hidden layers + 1 output layer
  - Layer 1: 15 neurons, tanh activation
  - Layer 2: 25 neurons, sigmoid activation
  - Layer 3: 25 neurons, ReLU activation
  - Output Layer: 1 unit, sigmoid activation, Adam optimizer
- Accuracy: 72.84%

Third Attempt:

- 3 hidden layers + 1 output layer
  - Layer 1: 30 neurons, tanh activation
  - Layer 2: 25 neurons, sigmoid activation
  - Layer 3: 20 neurons, sigmoid activation
  - Output Layer: 1 unit, sigmoid activation, Adam optimizer

- Accuracy: 72.29%

Model Performance:

The level of accuracy for the first attempt was 72.87% which is below the target performance rate of 75%. The second and third attempts were 72.84% and 72.29% accuracy respectively.

Performance Optimization Steps:

By removing non-informative variables, several techniques were applied to optimize the model:

- Added an extra hidden layer and varied neurons across layers to capture complex patterns.
- Dropped less-important columns, binned variables, and reduced the "Other" bin size to minimize noise.
- Experimented with different activation functions and optimizers to find the best configuration.

Summary and Recommendations:

Despite multiple iterations, the model's highest accuracy reached **72.87%,** falling short of the 75% target. Further fine-tuning of features and hyperparameters—such as adjusting the number of layers, neurons, and epochs—could improve performance toward meeting the target. Additionally, exploring other machine learning models like Random Forest or XGBoost may yield more effective results.