

Analysis of Neural Network Model

Overview

The purpose of this analysis is to try to optimize a model for the Alphabet Soup Foundation to select applicants with the best chance for success in their ventures using various features of the applicant. The model uses the following variables:

Results

Data Preprocessing

Target variable - "IS_SUCCESSFUL" which indicates that a venture was successful "1" or not successful "0"

Feature variables - feature variables were created from string variables by using the `get_dummies` function. I then dropped variables in an attempt to optimize the model which resulted in the following feature variables:

APPLICATION_TYPE_T19, APPLICATION_TYPE_T3, APPLICATION_TYPE_T4,
APPLICATION_TYPE_T5, APPLICATION_TYPE_T6, AFFILIATION_CompanySponsored,
AFFILIATION_Independent, CLASSIFICATION_C1000, CLASSIFICATION_C1200,
CLASSIFICATION_C2000, CLASSIFICATION_C2100, CLASSIFICATION_C3000,
USE_CASE_CommunityServ, USE_CASE_Healthcare, USE_CASE_Other,
USE_CASE_Preservation, USE_CASE_ProductDev, ORGANIZATION_Association,
ORGANIZATION_Co-operative, ORGANIZATION_Corporation, ORGANIZATION_Trust,
INCOME_AMT_0, INCOME_AMT_1-9999, INCOME_AMT_10000-24999,
INCOME_AMT_100000-499999, INCOME_AMT_10M-50M, INCOME_AMT_1M-5M,
INCOME_AMT_25000-99999, INCOME_AMT_50M+, INCOME_AMT_5M-10M

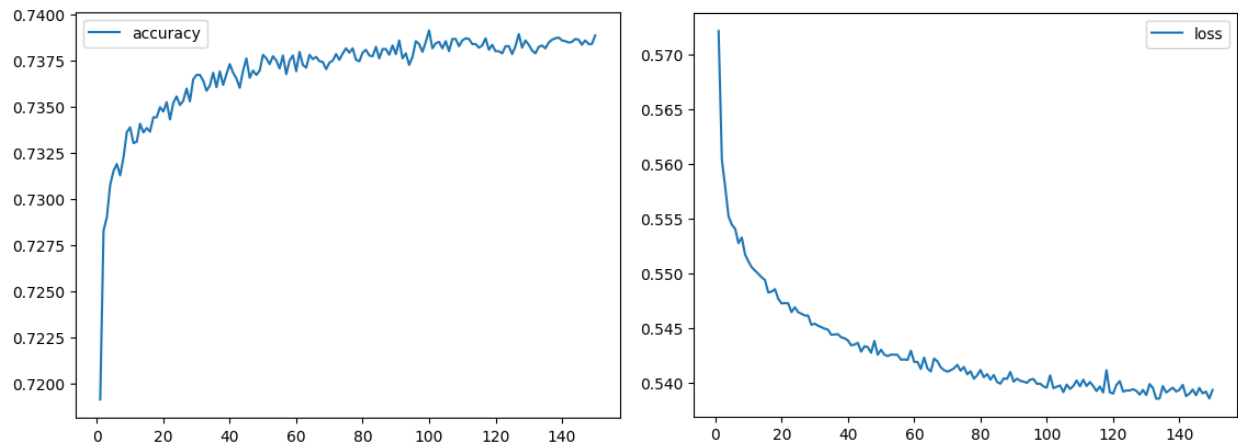
Two variables, EIN and NAME, were removed from the input data because they were neither targets nor features.

Compiling, Training, and Evaluating the Model

I used two hidden layers, both using the "relu" activation function, and an output layer that used the "sigmoid" activation function. In the first hidden layer I used 200 neurons and in the second layer I used 150 neurons. I tried using the `tanh` and `sigmoid` activation functions in the hidden layers, but ultimately `relu` seemed to produce the highest accuracy.

Although I tried probably 50+ combinations of datasets, activation functions, layers, epochs, and number of neurons I still was only able to achieve an accuracy score of .73. It was higher in some other combinations I tried, but it never reached the .75 target performance.

See below for the optimized model accuracy and loss visualizations:



Summary

Overall, the use of this deep learning model did not prove to be a highly accurate way of classifying which ventures would be successful. Potentially some different binning of variables or additional, more predictive variables, could be added to the model to make it more accurate. Additionally, further exploration could be done to determine which variables do not contribute to the model and could be dropped.

A logistic regression supervised learning model could also be used on a dataset such as this, where the outcome variable is binary. This may have simplified the modeling process and prevented the overfitting I was seeing as I experimented with additional layers and neurons.