Sabrina Alves
Neural Network Homework

The purpose of this report is to evaluate and predict whether or not funding for loan applicants based on application details is successful using a neural network algorithm. The following is the data provided to use in the algorithm:

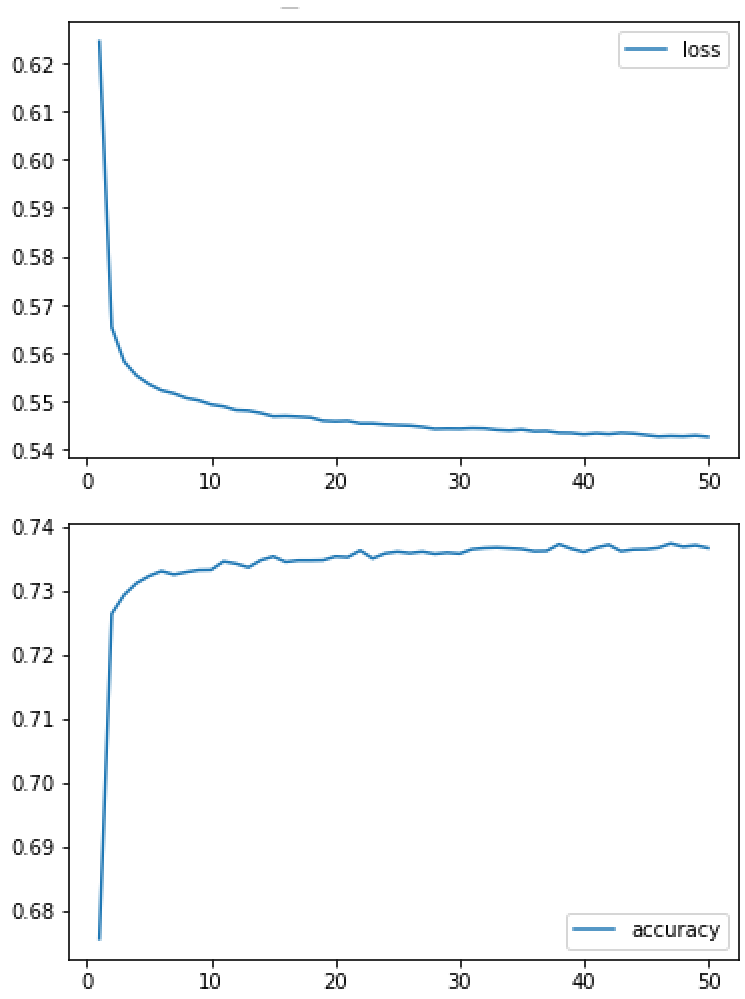
EIN and **NAME**—Identification columns
APPLICATION_TYPE—Alphabet Soup application type
AFFILIATION—Affiliated sector of industry
CLASSIFICATION—Government organization classification
USE_CASE—Use case for funding
ORGANIZATION—Organization type
STATUS—Active status
INCOME_AMT—Income classification
SPECIAL_CONSIDERATIONS—Special consideration for application
ASK_AMT—Funding amount requested
IS_SUCCESSFUL—Was the money used effectively

Data Preprocessing:

- The target of the model is the column IS_SUCCESSFUL
- The features of the model are the columns (with their data preprocessing procedure in parentheses):
    - APPLICATION_TYPE (one-hot encoded)
    - AFFILIATION (one-hot encoded)
    - CLASSIFICATION (one-hot encoded)
    - USE_CASE (one-hot encoded)
    - ORGANIZATION (one-hot encoded)
    - STATUS (one-hot encoded)
    - INCOME_AMT
    - SPECIAL_CONSIDERATIONS
    - ASK_AMOUNT (scaled)
- EIN and NAME are not targets or features and should be removed

Compiling, Training, and Evaluating the Model
Model 1:

- In order to avoid overfitting, I started with 16 total nodes across three layers (9, relu-6, relu-1, sigmoid), which generated an accuracy of 0.7252, which can be seen in the accuracy plot below.



- Relu and sigmoid were chosen as activation functions because of how they handle features. Sigmoid because it handles binary classification datasets and normalizes data to a probability between 0 and 1, which is the result we are looking for as a result of the algorithm. Relu because it is ideal for modeling positive, nonlinear data sets, which defines the data set well.

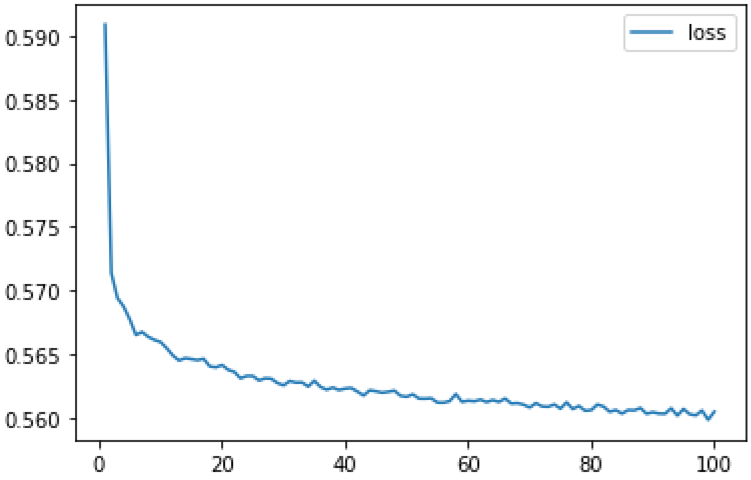
Model 2:

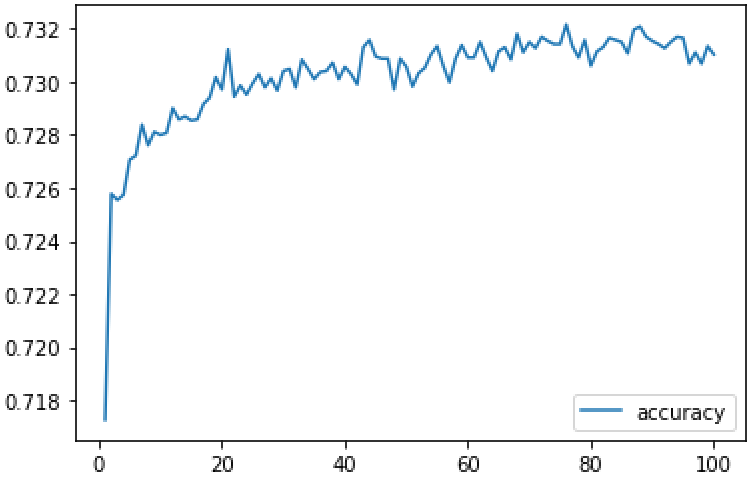
- I increased the number of nodes used to 26 (15- relu, 10-rely, 1-sigmoid).
- The number of epochs was increased to 75 to increase the fit.
- These changes did not increase the fit of the model, with overall accuracy of 0.7276, which is a small increase in accuracy from model 1.

Model 3:

- Since the general rule of thumb is to use 2-3 times the number of input features, I increased the number of nodes used to 66 (35-relu, 20-relu, 10-relu, 1-sigmoid) for model 3, as well as added a layer which should be overfitting (and can therefore be dialed back).

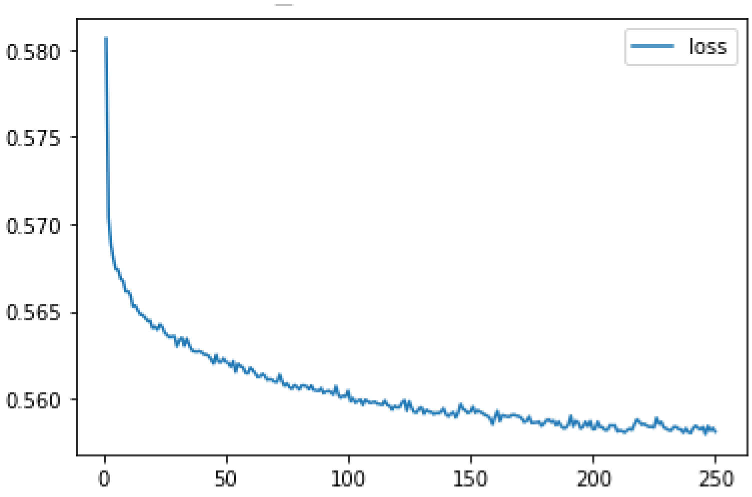
- The number of epochs was increased to 100 to increase the fit.
- The model, however, ultimately did not increase in accuracy to over 75%, staying at 0.7206 (a decrease from model 2), even though the accuracy started high.

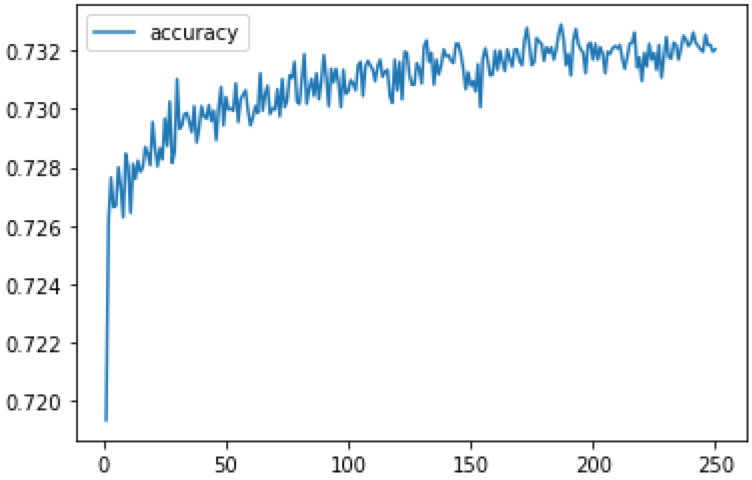




Model 4:

- Due to these changes, I concluded that the features need to be refined and dropped the classification one-hot encoded columns. This information was dropped due to its variability (there were about 16 columns for classification).
- I reran model 3 with an increased number of epochs (250) and with a tanh activation in the second layer, and this did not increase my model's accuracy substantially either (0.7221, reaching this range quickly but not learning past it).





- Tanh was added to attempt increasing the fit of the model, as relu and sigmoid did not capture the data set.

Target performance was not achieved with these steps taken.

Summary:

In summary, the neural network model does not produce an accurate learning/predicting engine with this data set and algorithm settings. For future work, different configurations of the features should be tested against the algorithm to see if there is improvement in the neural networks' efficiency.

Because we want the end product to be the probability of success from an application, a Supervised Learning model, such as a Logistic Regression and Decision Trees, should be used as the learning model for this data set.