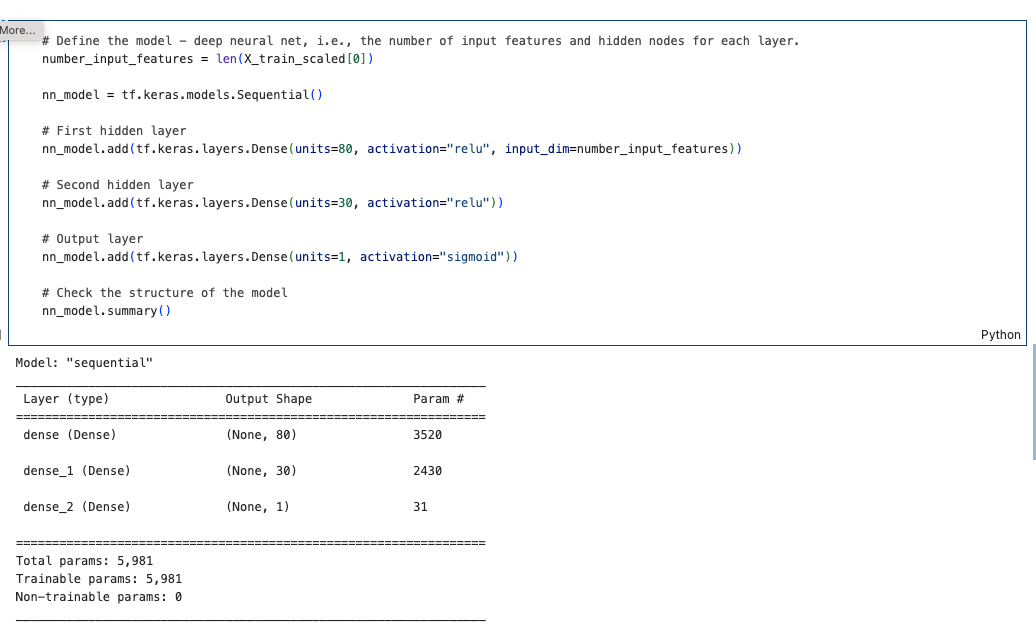
**Overview**

The task is to help Alphabet Soup prepare a predictive analysis tool to “Select the applicants for funding that will have the best chance of success in their venture”. Using machine learning and neural networks the task is to preprocess the available dataset provided. We will compile the data into a neural network model using TensorFlow. We are going to train the model to make predictions about the dataset by selecting data features, setting a binned structure for the data, and re-running the data with an algorithm to best learn how to understand the data. Lastly we will evaluate the output to determine how much data was lost and how accurate the model is at its predictions.

**Results**:

Figure 1: Original model showing 2 hidden layers with 80, 30 neurons, respectively. The Relu activation function was used. Accuracy = 72.7% Loss = 0.566



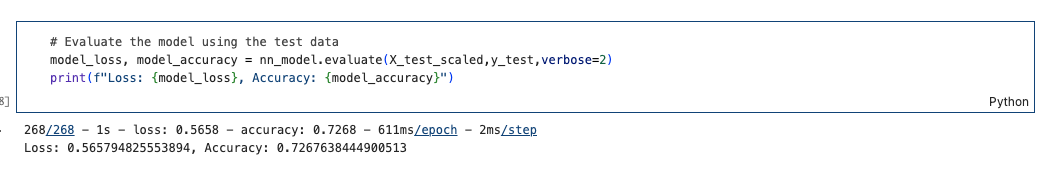
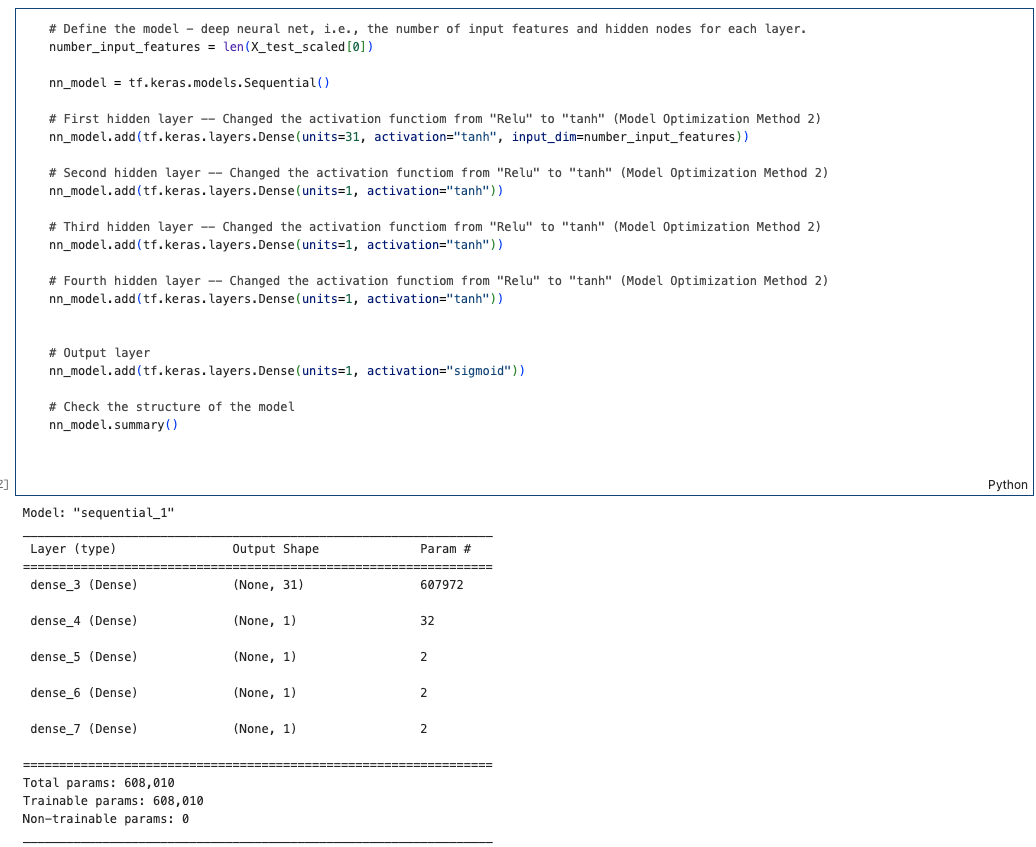
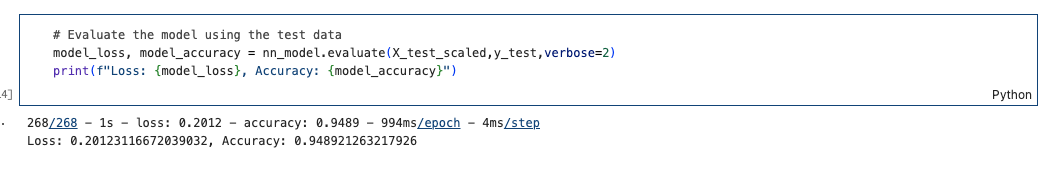


Figure 2: Optimized model showing 4 hidden layers with 31, 1, 1, 1 neurons, respectively. The Relu activation function was switched to tanh. Accuracy = 94.9% Loss = 0.201





The optimization of the model led to 22.2% increase in accuracy and decrease of 0.365 in data loss.

Using bulleted lists and images to support your answers, address the following questions:

* Data Preprocessing
  + What variable(s) are the target(s) for your model?
    - IS\_SUCCESSFUL
  + What variable(s) are the features for your model?
    - The features of the model include:
      * APPLICATION\_TYPE
      * AFFILIATION
      * CLASSIFICATION
      * USE\_CASE
      * ORGANIZATION
      * STATUS
      * INCOME\_AMT
      * SPECIAL\_CONSIDERATIONS
      * ASK\_AMT
  + What variable(s) should be removed from the input data because they are neither targets nor features?
    - Both EIN and NAME can be considered features with little importance for modeling. However, it will be explained later, that including NAME back into the model had a major influence on the accuracy output.
* Compiling, Training, and Evaluating the Model
  + How many neurons, layers, and activation functions did you select for your neural network model, and why?
    - Based on the results given by the keras tuner, a model using four hidden layers, with 31, 1, 1, 1 neurons respectively, and utilizing the tanh activation function provided a model with an accuracy of
  + Were you able to achieve the target model performance?
    - Yes I was able to increase the accuracy from 73% to 95% (22% increase)
  + What steps did you take in your attempts to increase model performance?
    - There were several actions taken to achieve this goal. The first attempt at optimizing the data involved adjusting the level set for the Classification and Application Type bins. However, further tests concluded that these adjustments never gained any significant change in accuracy. Tuning the model to find the best hyperparameters led to a better optimization by changing the activation function from Relu to tanh, doubling (from 2 to 4) the hidden layers, and adjusting the neurons based on the tuner’s output. The evaluation was performed with a reduction of Epochs from 100, originally, down to 50. This is in effort to optimize run time. Lastly it was discovered that the models surpassed the accuracy goal of 75% by including NAME as a feature.

**Summary**

It is believed strongly that without adding the categorical feature, NAME, for analysis in the neural network, the algorithm would never be able to gain an accuracy above 75%. Other adjustments were explored with mediocre results. It appears despite the feature to be highly unique, categorical and prone to a disruption if new data were incorporated, the neural network performed better than the original model. By dichotomizing the Names of applicants, the neural network was able to provide a better prediction because weight was attributed to the applicant itself, as with the other features. I would recommend exploring more models that put weight on Affiliation, Income Amount, and Status. All of these features seemed the most relevant towards making a decision on funding an organization. However, previous tests with the optimized data never got beyond 73% accuracy. Depending on Alphabet Soup’s tolerance for risk, these features could be accepted as better values for prediction, but that is dependent more on a human process than a machine.