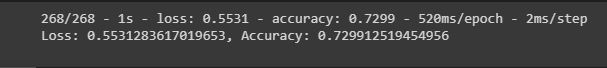
Alphabet Soup Analysis

Overview:  
The nonprofit foundation Alphabet Soup wants to create a tool that can help it select the applicants for funding with the best chance of success in their ventures. It’ll require specific features to create a binary classifier that can predict whether applicants will be successful if funded by Alphabet Soup.

Results:

* Data Preprocessing
  + What variable(s) are the target(s) for your model?  
    The “IS\_SUCCESSFUL” column was used as the target for our model.
  + What variable(s) are the features for your model?  
    The features for our model would include: Application\_type, Affiliation, Classification, Use\_Case, Organization, Status, Income\_AMT, Special\_Considerations and Ask\_AMT.
  + What variable(s) should be removed from the input data because they are neither targets nor features?  
    The Name and EIN columns should be removed.
* Compiling, Training, and Evaluating the Model
  + How many neurons, layers, and activation functions did you select for your neural network model, and why?  
    I went with the random approach, trying a variety of neurons, layers and activation functions.   
    Test 1: 2 layers, 90/120 Neurons and LeakyReLU activation.  
    Test 2: 3 layers, 7/14/21 Neurons and relu activation.  
    Test 3: 4 layers, 10/20/30/40 Neurons and relu activation.
  + Were you able to achieve the target model performance?  
    Unfortunately no, the highest I was able to achieve was 72.9%  
    
  + What steps did you take in your attempts to increase model performance?  
    My steps involved adding more layers while keeping the neurons relatively close.

Summary:

Overall, there was not a big change when adding in additional Layers and Neurons. The three models that I ran all came back at around 73%. With further testing and adding additional features, I believe we would be able to reach a higher accuracy within our models. Allowing us to pass the 75% accuracy and being a valuable tool for Alphabet Soup to use.