

# Overview:

The nonprofit foundation Alphabet Soup wants a tool that can help it select the applicants for funding with the best chance of success in their ventures. Using knowledge of machine learning and neural networks, Alphabet Soup has tasked one with using the data provided to create a binary classifier capable of predicting successful applicants. The model must have a 75% accuracy in order to be put into production.

## Results:

### Data Preprocessing

- All columns besides “EIN” and “NAME” were considered features for the first model. While in the second model’s “NAME” was left as a feature
- The target variable for both models was the “IS\_SUCCESSFUL” column which was made into our y variable.

### Compiling, Training, and Evaluating the Model

- There were three layers and three activation functions for the neural network.
- Target model performance was not achieved in the first round. This model only achieved a 60.1% accuracy.

```
# Evaluate the model using the test data
model_loss, model_accuracy = nn.evaluate(X_test_scaled,y_test,verbose=2)
print(f"Loss: {model_loss}, Accuracy: {model_accuracy}")

268/268 - 0s - loss: 1.1949 - accuracy: 0.6014 - 339ms/epoch - 1ms/step
Loss: 1.1949056386947632, Accuracy: 0.6013994216918945
```

- Target model performance was achieved in the second attempt by adding the name column as a feature. This rose the accuracy to 77.3%.

```
# Evaluate the model using the test data
model_loss, model_accuracy = nn.evaluate(X_test_scaled,y_test,verbose=2)
print(f"Loss: {model_loss}, Accuracy: {model_accuracy}")

268/268 - 0s - loss: 0.6022 - accuracy: 0.7734 - 356ms/epoch - 1ms/step
Loss: 0.6022396087646484, Accuracy: 0.7734110951423645
```

## Summary:

Ultimately, all columns except for “EIN” must be taken into consideration as features and “IS\_SUCCESSFUL” must be set as the target. In order to train a model that would satisfy Alphabet Soup 75% accuracy guideline.