

Deep Learning Model

Overview

The goal of this challenge was to implement knowledge from machine learning and neural networks to create a machine learning model. The machine learning model would need to be able to predict whether or not an applicant would be successful if funded by the non-profit foundation, Alphabet Soup. The data set provided contained over 34,000 companies that received funding from Alphabet Soup.

Results

- Data Processing
 - The targeted variable for this model is the “Is_Successful” marker because we want our model to predict whether or not a company will be successful.
 - The “Is_Successful” marker is also the key feature.
 - The “EIN” and “Name” columns were dropped because they are neither targets nor features.
- Compiling, Training, and Evaluating the model

```

# Define the model - deep neural net, i.e., the number of input features and hidden nodes for each layer.
n_features = len(X_train_scaled[0])
print(n_features)

nn = tf.keras.models.Sequential()

# First hidden layer
nn.add(tf.keras.layers.Dense(units=40, activation="relu", input_dim=n_features))

# Second hidden layer
nn.add(tf.keras.layers.Dense(units=20, activation="relu"))

# Third hidden layer
nn.add(tf.keras.layers.Dense(units=10, activation="relu"))

# Output layer
nn.add(tf.keras.layers.Dense(units=1, activation="sigmoid"))

# Check the structure of the model
nn.summary()

```

-
- Initially 2 hidden layers were used, but produced only a 68% accuracy. Adding a 3rd layer increased the accuracy to 74%.
- I wasn't able to achieve the target performance of over 75%, but I got pretty close to it with 74%

```

# 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}")

```

7]

```

268/268 - 1s - loss: 0.6842 - accuracy: 0.7425 - 591ms/epoch - 2ms/step
Loss: 0.6842029690742493, Accuracy: 0.74250727891922

```

-
- As mentioned before, adding another hidden layer increased the model's performance.
- Summary
 - My first model achieved an accuracy of 68%, and by increasing the number of hidden layers, it was able to achieve 74% prediction accuracy. I would recommend a change in hidden layer amount or units for better predictability.