Deep Learning Challenge: Charity Funding Predictor

Overview

Deep learning and neural networks were used to determine whether or not applicants would be successful in receiving funding from Alphabet Soup, which has previously funded over 34,000 organizations.

Data Preprocessing

Because the dataset removed any irrelevant information, EIN and NAME were removed from the model. The remaining columns were considered model features. Despite the fact that NAME was reinstated in the second test. Due to high fluctuation, CLASSIFICATION and APPLICATION TYPE were replaced with 'Other.' The data was divided into training and testing sets. The model's target variable is "IS SUCCESSFUL," and its value, 1 was considered yes and 0 was considered no. The data from APPLICATION was analyzed, and the value of CLASSIFICATION was used for binning. Each one-of-a-kind value used several data points as a cutoff point to group "rare" categorical variables into a new value, 'Other.' Then I checked to see if the binning was successful. 'pd.get dummies' encoded categorical variables ().

Compiling, Training and Model Evaluation

Each model had three layers of neural networks applied to it. The number of hidden nodes was determined by the number of features.

In [ ]:
```
number_input_features = len( X_train_scaled[0])
hidden_nodes_layer1=7
hidden_nodes_layer2=14
hidden_nodes_layer3=21
nn = tf.keras.models.Sequential()

nn = tf.keras.models.Sequential()

nn.add(tf.keras.layers.Dense(units=hidden_nodes_layer1, input_dim=number_input_featu

nn.add(tf.keras.layers.Dense(units=hidden_nodes_layer2, activation='relu'))

nn.add(tf.keras.layers.Dense(units=1, activation='sigmoid'))

nn.summary()
```

In [ ]:
```
477 parameters were generated by a three-layer training model. The first attempt cam
```

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

Optimization

The second attempt added 'NAME' back into the dataset; this time I got 79%, which was 4% higher than the target. There are a total of 3,298 parameters.

Model: "sequential_1"

Layer (type) Output Shape Param

dense (Dense) (None, 7) 3171

dense_1 (Dense) (None, 14) 112

dense_2 (Dense) (None, 1) 15

===

Total params: 3,298 Trainable params: 3,298 Non-trainable params: 0

Because deep learning is machine-based, it teaches a computer to filter inputs through the layers in order to learn how to predict and classify information.