Purpose of this analysis

Use the features in the provided dataset to create a binary classifier that can predict whether applicants will be successful if funded by Alphabet Soup.

Results

- **What variable(s) are the target(s) for your model?**

IS_SUCCESSFUL
1
1
0
1
1

- **What variable(s) are the features for your model?**

APPLICATION_TYPE	AFFILIATION	CLASSIFICATION	USE_CASE	ORGANIZATION	STATUS	INCOME_AMT	SPECIAL_CONSIDERATIONS	ASK_AMT
T10	Independent	C1000	ProductDev	Association	1	0	N	5000
T3	Independent	C2000	Preservation	Co-operative	1	1-9999	N	108590
T5	CompanySponsored	C3000	ProductDev	Association	1	0	N	5000
T3	CompanySponsored	C2000	Preservation	Trust	1	10000-24999	N	6692
T3	Independent	C1000	Heathcare	Trust	1	100000-499999	N	142590

- **What variable(s) should be removed from the input data because they are neither targets nor features?**

EIN	NAME
10520599	BLUE KNIGHTS MOTORCYCLE CLUB
10531628	AMERICAN CHESAPEAKE CLUB CHARITABLE TR
10547893	ST CLOUD PROFESSIONAL FIREFIGHTERS
10553066	SOUTHSIDE ATHLETIC ASSOCIATION
10556103	GENETIC RESEARCH INSTITUTE OF THE DESERT

- **How many neurons, layers, and activation functions did you select for your neural network model, and why?**

80 neurons for the first layer, 30 neurons for the second layer, and 1 neuron for the output layer, a sigmoid activation function is used for this layer since this is a binary classification case. 2 activation functions and 3 layers in total.

```
# Define the model - deep neural net, i.e., the number of input features and hidden nodes for each layer.
nn1 = tf.keras.models.Sequential()


# First hidden layer
nn1.add(tf.keras.layers.Dense(units=80, activation="relu", input_dim=43))

# Second hidden layer
nn1.add(tf.keras.layers.Dense(units=30, activation="relu", input_dim=30))

# Output layer
nn1.add(tf.keras.layers.Dense(units=1, activation="sigmoid"))
```

- **Were you able to achieve the target model's performance?**

No.

- **What steps did you take in your attempts to increase model performance?**

Attempt 1: lower epoch.

Attempt 2: Add more layers and neurons.

Attempt 3: Change activation function.

Summary

The initial model didn't work very well since the accuracy is around 0.726, the optimized model didn't work well either, the accuracy is around 0.727. Even with 3 attempts, the result is not cheerful. Gaining more data and creating more bins are suggested for further analysis.