



```
[1] ✓ 5m
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Input

# Input data (XOR)
X = np.array([
    [0, 0],
    [0, 1],
    [1, 0],
    [1, 1]
], dtype='float32')

Y = np.array([
    [0],
    [1],
    [1],
    [0]
], dtype='float32')

# Create Feed Forward Neural Network
model = Sequential()
model.add(Input(shape=(2,)))
model.add(Dense(2, activation='tanh'))
model.add(Dense(1, activation='sigmoid'))

# Compile model
model.compile(
    loss='binary_crossentropy',
    optimizer='adam',
    metrics=['accuracy']
)

# Train model
model.fit(X, Y, epochs=10000, verbose=0)

# Predictions
print("Predictions for XOR inputs:")
predictions = model.predict(X)
print(predictions)

rounded_predictions = np.round(predictions)
print("Rounded Prediction:")
print(rounded_predictions)

# Evaluate accuracy
loss, accuracy = model.evaluate(X, Y, verbose=0)
print(f"\nAccuracy: {accuracy * 100:.0f}%")
```



Q Commands + Code + Text ▶ Run all ▼



[1]

✓ 5m



```
print('Predictions for XOR inputs: ',
predictions = model.predict(X)
print(predictions)

rounded_predictions = np.round(predictions)
print("Rounded Prediction:")
print(rounded_predictions)

# Evaluate accuracy
loss, accuracy = model.evaluate(X, Y, verbose=0)
print(f"\nAccuracy: {accuracy * 100:.0f}%")
```



```
... Predictions for XOR inputs:
1/1 _____ 0s 58ms/step
[[0.00246465]
 [0.996208 ]
 [0.9947739 ]
 [0.00200161]]
Rounded Prediction:
[[0.]
 [1.]
 [1.]
 [0.]]

Accuracy: 100%
```