PROGRAM

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.utils import to_categorical

url = " url of data set"
columns = ['sepal_length', 'sepal_width', 'petal_length', 'class']
iris_data['class'] = iris_data['class'].map({'Iris-setosa': 0,
'Iris-versicolor': 1, 'Iris-virginica': 2})
X = iris_data.iloc[:, :-1].values
y = iris_data['class'].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
y_train = to_categorical(y_train)
y_test = to_categorical(y_test)
model = Sequential()
model.add(Dense(10, input_dim=4, activation='relu'))
model.add(Dense(10, activation='relu'))
model.add(Dense(3, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
model.fit(X_train, y_train, epochs=100, batch_size=5, verbose=1)
loss, accuracy = model.evaluate(X_test, y_test)
print("Test Accuracy: {accuracy:.2f}")
```

28/10/24

EXPNO: ~~[illegible]~~ 5

AIM: Implement artificial neural networks for an application using python.

ABOUT:

- contains artificial neurons.
- The neurons are connected to each other.
- They are arranged in layeres to constitute a neural network.
- The data passes through them multiple layes and gets processed
- The output layer provides output for the network.

ALGORITHM

1. Start by importing necessary libraries
2. Load the iris dataset
3. Split data set into training and testing.
4. Create simple forward neural network
5. Fit model to training data
6. Check models performance on data

RESULT

Thus the program has been implemented.