CO4

# Question: 19 feed forward network using iris data set

Q: Programs on feedforward network to classify any standard dataset available in the public domain.

Procedure

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.datasets import load_iris
from tensorflow.keras import models, layers


# Load the Iris dataset
iris = load_iris()
X = iris.data
y = iris.target


# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Standardize the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)


# One-hot encode the target variable
label_encoder = LabelEncoder()
y_train = label_encoder.fit_transform(y_train)
y_test = label_encoder.transform(y_test)


# Create a simple feedforward neural network model
model = models.Sequential()
model.add(layers.Dense(64, activation='relu', input_shape=(4,)))
model.add(layers.Dense(32, activation='relu'))
```

```
model.add(layers.Dense(3, activation='softmax'))  # Output layer with 3 classes


# Compile the model
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',  # Use 'categorical_crossentropy' if y is
one-hot encoded
              metrics=['accuracy'])


# Train the model
model.fit(X_train, y_train, epochs=50, batch_size=8, validation_split=0.1)


# Evaluate the model on the test set
test_loss, test_acc = model.evaluate(X_test, y_test)
print(f'Test accuracy: {test_acc}')
```

Output