```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler


# Import the dataset
df = pd.read_csv("/content/SDN_DDoS_.csv")


# Splitting dataset into features and label
x=df.drop('Label', axis =1)
y = df['Label']

# Splitting the dataset into the training set and the test set
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0)


# Feature scaling (or standardization)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)


print(X_train.shape)
print(y_train.shape)
print(X_test.shape)
print(y_test.shape)
```

```
(60820, 66)
(60820,)
(15206, 66)
(15206,)
```

```
from tensorflow.keras.optimizers import SGD, Adam
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Dropout
classifier = Sequential()
classifier.add(Dense(66, input_dim=66, activation='relu'))
classifier.add(Dropout(0.2))
classifier.add(Dense(30,  activation='relu'))
classifier.add(Dropout(0.2))
classifier.add(Dense(1, activation='sigmoid'))
sgd = SGD(lr=0.01, momentum=0.8)

classifier.compile(loss='binary_crossentropy', optimizer=sgd, metrics=['accuracy'])
classifier.fit(X_train, y_train, validation_data=(X_test, y_test), batch_size = 100, epochs =
```

```
Epoch 73/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 74/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0019 - accuracy: 0.
Epoch 75/100
```

```
Epoch 75/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 76/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0014 - accuracy: 0.
Epoch 77/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 78/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 79/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 80/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 81/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0013 - accuracy: 0.
Epoch 82/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 83/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 84/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 85/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 86/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 87/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 88/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 89/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 90/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 91/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0010 - accuracy: 0.
Epoch 92/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0010 - accuracy: 0.
Epoch 93/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0010 - accuracy: 0.
Epoch 94/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 95/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 96/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 97/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0011 - accuracy: 0.
Epoch 98/100
609/609 [==============================] - 2s 3ms/step - loss: 9.9259e-04 - accuracy
Epoch 99/100
609/609 [==============================] - 2s 3ms/step - loss: 0.0012 - accuracy: 0.
Epoch 100/100
609/609 [==============================] - 2s 3ms/step - loss: 9.9131e-04 - accuracy
<keras.callbacks.History at 0x7ff990509dd0>
```

```
testloss,testaccuracy= classifier.evaluate(X_test, y_test)
print('test loss : ', testloss)
print('test accuracy :', testaccuracy)
```

```
476/476 [==============================] - 1s 2ms/step - loss: 7.5512e-04 - accuracy: 0
test loss :  0.0007551221642643213
test accuracy : 0.9994081258773804
```

```
import tensorflow
model=tensorflow.keras.models.Sequential()
#Hidden layer definitions
model.add(tensorflow.keras.layers.Dense(units=66, activation='relu', input_shape = X_train.sh
model.add(tensorflow.keras.layers.Dense(units=50, activation='relu'))
model.add(tensorflow.keras.layers.Dropout(0.2))
model.add(tensorflow.keras.layers.Dense(units=50, activation='relu' ))
model.add(tensorflow.keras.layers.Dropout(0.2))
model.add(tensorflow.keras.layers.Dense(units=50, activation='relu' ))
#OP layer
model.add(tensorflow.keras.layers.Dense(units=2, activation='sigmoid' ))
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_3 (Dense)             (None, 66)                4422

 dense_4 (Dense)             (None, 50)                3350

 dropout_2 (Dropout)         (None, 50)                0

 dense_5 (Dense)             (None, 50)                2550

 dropout_3 (Dropout)         (None, 50)                0

 dense_6 (Dense)             (None, 50)                2550

 dense_7 (Dense)             (None, 2)                 102

=================================================================
Total params: 12,974
Trainable params: 12,974
Non-trainable params: 0
_________________________________________________________________
```

```
#compile
model.compile(loss='sparse_categorical_crossentropy' , optimizer='adam' , metrics=['accuracy'

#fit
history=model.fit(x=X_train, y=y_train, validation_split=0.1, epochs=100, batch_size=16)
```

```
Epoch 73/100
```

```
Epoch 73/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.8296e-07 - accura
Epoch 74/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.4438e-08 - accura
Epoch 75/100
3422/3422 [==============================] - 9s 3ms/step - loss: 8.3272e-08 - accura
Epoch 76/100
3422/3422 [==============================] - 9s 3ms/step - loss: 2.3738e-09 - accura
Epoch 77/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.8947e-10 - accura
Epoch 78/100
3422/3422 [==============================] - 9s 3ms/step - loss: 7.3392e-10 - accura
Epoch 79/100
3422/3422 [==============================] - 9s 3ms/step - loss: 5.5469e-04 - accura
Epoch 80/100
3422/3422 [==============================] - 9s 3ms/step - loss: 8.4669e-04 - accura
Epoch 81/100
3422/3422 [==============================] - 9s 3ms/step - loss: 9.4900e-08 - accura
Epoch 82/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.0620e-07 - accura
Epoch 83/100
3422/3422 [==============================] - 9s 3ms/step - loss: 2.4663e-08 - accura
Epoch 84/100
3422/3422 [==============================] - 9s 3ms/step - loss: 5.8298e-04 - accura
Epoch 85/100
3422/3422 [==============================] - 9s 3ms/step - loss: 2.0251e-07 - accura
Epoch 86/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.3761e-07 - accura
Epoch 87/100
3422/3422 [==============================] - 9s 3ms/step - loss: 4.2902e-08 - accura
Epoch 88/100
3422/3422 [==============================] - 9s 3ms/step - loss: 4.1160e-09 - accura
Epoch 89/100
3422/3422 [==============================] - 9s 3ms/step - loss: 8.9775e-04 - accura
Epoch 90/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.1787e-06 - accura
Epoch 91/100
3422/3422 [==============================] - 9s 3ms/step - loss: 4.3436e-08 - accura
Epoch 92/100
3422/3422 [==============================] - 9s 3ms/step - loss: 0.0010 - accuracy:
Epoch 93/100
3422/3422 [==============================] - 9s 3ms/step - loss: 0.0013 - accuracy:
Epoch 94/100
3422/3422 [==============================] - 9s 3ms/step - loss: 3.5546e-04 - accura
Epoch 95/100
3422/3422 [==============================] - 9s 3ms/step - loss: 2.8979e-04 - accura
Epoch 96/100
3422/3422 [==============================] - 9s 3ms/step - loss: 1.5340e-05 - accura
Epoch 97/100
3422/3422 [==============================] - 9s 3ms/step - loss: 3.3942e-06 - accura
Epoch 98/100
3422/3422 [==============================] - 9s 3ms/step - loss: 8.7425e-04 - accura
Epoch 99/100
3422/3422 [==============================] - 9s 3ms/step - loss: 0.0015 - accuracy:
Epoch 100/100
3422/3422 [==============================] - 9s 3ms/step - loss: 0.0056 - accuracy:
```

```
testloss,testaccuracy= model.evaluate(X_test, y_test)
print('test loss : ', testloss)
print('test accuracy :', testaccuracy)
```

```
476/476 [==============================] - 1s 2ms/step - loss: 2.4859e-08 - accuracy: 1
test loss :  2.485874084356965e-08
test accuracy : 1.0
```

✓ 0s completed at 7:35 PM