**Machine Learning**

**Ex: 08 Multi-Layer Peceptron**

**Name:**Athithraja. R

**Reg.no:** 2022503702

1)

**Code:**

import pandas as pd

from sklearn.preprocessing import StandardScaler

from sklearn.model\_selection import train\_test\_split

from sklearn.neural\_network import MLPClassifier

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import classification\_report

import matplotlib.pyplot as plt

df=pd.read\_csv("Forest\_Fire\_Dataset.csv")

#print(df)

#le=LabelEncoder()

df['Fire'], unique\_values = pd.factorize(df['Classes'])

#df['Classes']=le.fit\_transform(df['Classes'])

#df['Classes']=df['Classes'].replace('fire','1')

inplist=df.columns[:-3]

scale=StandardScaler()

df[inplist]=scale.fit\_transform(df[inplist])

#print(df.tail(10))

"""

y=df.values[:,-1]

for i in range(len(y)):

print(str(i+2)+" "+str(y[i]))

"""

x=df.iloc[:,:-3].values

y=df['Fire'].values

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.3, random\_state=11)

clf=MLPClassifier(hidden\_layer\_sizes=(3,),activation='logistic',max\_iter=150,solver='adam',learning\_rate='constant',learning\_rate\_init=0.19)

clf.fit(x\_train,y\_train)

ypred = clf.predict(x\_test)

cm = confusion\_matrix(y\_test, ypred)

print("Confusion Matrix:")

print(cm)

print("--------------------------------------------------------------------------")

print('Classification Report:\n',classification\_report(y\_test,ypred))

print("--------------------------------------------------------------------------")

print('Coefficient: ',clf.coefs\_)

print("--------------------------------------------------------------------------")

print("Intercepts: ",clf.intercepts\_)

print("--------------------------------------------------------------------------")

loss\_values=clf.loss\_curve\_

plt.plot(loss\_values)

plt.xlabel('Epochs')

plt.ylabel('Loss')

plt.show()

**Output:**



