### **INDEX**

| No. | Practical | Date | Sign |
| --- | --- | --- | --- |
| 1 | Implement a basic feedforward neural network from scratch |  |  |
| 2 | Implement different activation functions and optimization techniques |  |  |
| 3 | Build a CNN for image classification using a suitable dataset |  |  |
| 4 | Apply transfer learning using VGG |  |  |
| 5 | Implement a simple RNN for time series prediction |  |  |
| 6 | Develop a sequence to sequence model for machine learning |  |  |
| 7 | Implement GAN to generate synthetic image |  |  |
| 8 | Apply transformer model to text generation tasks |  |  |
| 9 | Deploy a deep learning model as an API |  |  |
| 10 | Research and implement a cutting-edge technique in s small-scale project |  |  |

## 

## 1 - Implement a basic feedforward neural network from scratch

import pandas as pd

from sklearn.datasets import load\_iris

data = load\_iris()

df = pd.DataFrame(data.data, columns=data.feature\_names)

df['target'] = data.target

x = df.drop('target', axis=1)

y = df['target']

weights = [0.8, 0.3, 0.1, 0.5]

threshold = 0.5

weighted\_sum = 0

epoch = 10

learning\_rate = 0.1

for epoch in range(epoch):

print(f"Epoch {epoch + 1}")

for i in range(len(x)):

weighted\_sum = sum([weights[j] \* x.iloc[i][j] for j in range(len(weights))])

predicted = 1 if weighted\_sum >= threshold else 0

error = y[i] - predicted

if error != 0:

for j in range(len(weights)):

weights[j] += learning\_rate \* error \* x.iloc[i][j]

print(f"Input: {x.iloc[i]}, Predicted: {predicted}, True: {y[i]}, Weights: {weights}")

**Output:**

Epoch 1

Input: sepal length (cm) 5.1, sepal width (cm) 3.5, petal length (cm) 1.4, petal width (cm) 0.2

Name: 0, dtype: float64, Predicted: 1, True: 0, Weights: [0.290000, -0.050000, -0.039999, 0.48]

Input: sepal length (cm) 4.9, sepal width (cm) 3.0, petal length (cm) 1.4, petal width (cm) 0.2

Name: 1, dtype: float64, Predicted: 1, True: 0, Weights: [-0.2, -0.350000, -0.179999, 0.459999]

**…**

Epoch 10

Input: sepal length (cm) 5.1, sepal width (cm) 3.5, petal length (cm) 1.4, petal width (cm) 0.2

Name: 0, dtype: float64, Predicted: 1, True: 0, Weights: [96.210000, -3.669999, 191.510000, 81.909999]

Input: sepal length (cm) 4.9, sepal width (cm) 3.0, petal length (cm) 1.4, petal width (cm) 0.2

Name: 1, dtype: float64, Predicted: 1, True: 0, Weights: [95.720000, -3.969999, 191.370000, 81.889999]

****

****

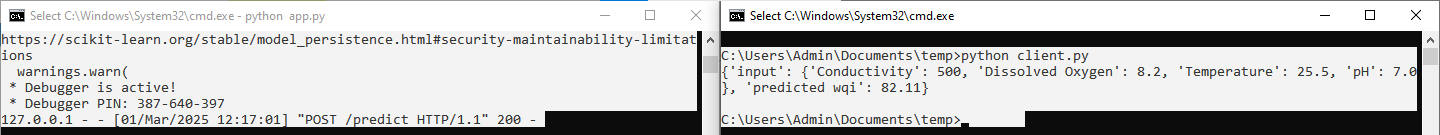
****

****

****

**Output:**

**run python app.py:**



**run python client.py:**

