



Task 2

Dry Beans



Prepared by :

Team 15

abdulrhmansayed70@gmail.com

TEAM MEMBERS

NAME	ID	DEPARTMENT
هبة طارق كمال عبدالمطلب	20201700959	CS
عبدالرحمن سيد جابر أحمد	20201701089	CS
نورهان ايمن محمد عبدالرحمن	20201700939	CS
حنين ابراهيم امام عكاشه	20201700230	CS
مريم احمد اسماعيل محمود	20201700800	CS
رقيه محمد ابراهيم مصطفى عبده	20201701253	CS

PRE-PROCESSING

1. Null Handling

Fill nulls with the mean

2. Normalize Train Data

Normalize using Min Max Scaler

3. Normalize Test Data

Normalize using Min Max Scaler

4. Label Encoder

Label the output classes

5. Data Splitting

Split the data set into train & test

BEST ACCURACY

Results

Activation Function	Sigmoid	Tanh
Train Accuracy	100%	100%
Test Accuracy	100%	100%
LR	1.0	0.1
Epochs	1000	100
#Layers	4	2
#HiddenNodes	3,4,2,2	3,4

Training Combinations

Sigmoid Activation Function

Task2

Dry_Bean_Dataset.csv

evaluate_old.py

main.py

multi_layer_perceptron.py

main

Test Accuracy: 0.9833333333333333
data: 148

Train Confusion Matrix:
[[30. 0. 0.]
 [0. 30. 0.]
 [0. 0. 30.]]
Train Accuracy: 1.0
Test Confusion Matrix:
[[20. 0. 0.]
 [0. 20. 0.]
 [0. 0. 20.]]
Test Accuracy: 1.0

Number of hidden layers

3

Number of neurons

3, 4, 6

Learning rate (eta)

1

Number of epochs (m)

100

Activation Function

Sigmoid

Hyperbolic Tangent

Area

Perimeter

Major Length

Minor Length

Roundnes

Bias

Train

Classify

Abdelrhman_multi_Layer_percep

Dry_Bean_Dataset.csv

evaluate.py

evaluate_old.py

main.py

multi_layer_perceptron.py

multi_layer_perceptron_nonVect

multi_layer_perceptron_vectroiz

preprocessing.py

main.py

README.md

main

Train Confusion Matrix:
[[30. 0. 0.]
 [0. 29. 1.]
 [0. 0. 30.]]
Train Accuracy: 0.9888888888888889
Test Confusion Matrix:
[[20. 0. 0.]
 [0. 19. 1.]
 [0. 0. 20.]]
Test Accuracy: 0.9833333333333333

Number of hidden layers

2

Number of neurons

2, 4

Learning rate (eta)

0.5

Number of epochs (m)

40

Activation Function

Sigmoid

Hyperbolic Tangent

Area

Perimeter

Major Length

Minor Length

Roundnes

Bias

Train

Classify

Training Combinations

Hyperbolic Tangent Activation Function

```
perceptron.py
preprocessing.py
Task2
  Dry_Bean_Dataset.csv
  evaluate_old.py
  main.py
  multi_layer_perceptron.py
main
```

```
Test Accuracy: 1.0
data: 148
*****
Train Confusion Matrix:
[[30.  0.  0.]
 [ 0. 30.  0.]
 [ 0.  0. 30.]]
Train Accuracy: 1.0
Test Confusion Matrix:
[[20.  0.  0.]
 [ 0. 20.  0.]
 [ 0.  0. 20.]]
Test Accuracy: 1.0
```

Number of hidden layers 2

Number of neurons 3, 4

Learning rate (η) 0.1

Number of epochs (m) 100

Activation Function

Sigmoid **Hyperbolic Tangent**

Area Perimeter Major Length Minor Length Roundnes

Bias

Train **Classify**

```
perceptron.py
preprocessing.py
Task2
  Dry_Bean_Dataset.csv
  evaluate_old.py
  main.py
  multi_layer_perceptron.py
main
```

```
Test Accuracy: 0.9666666666666667
data: 148
*****
Train Confusion Matrix:
[[30.  0.  0.]
 [ 0. 30.  0.]
 [ 0.  0. 30.]]
Train Accuracy: 1.0
Test Confusion Matrix:
[[20.  0.  0.]
 [ 0. 20.  0.]
 [ 1.  0. 19.]]
Test Accuracy: 0.9833333333333333
```

Number of hidden layers 3

Number of neurons 3, 4, 6

Learning rate (η) 0.1

Number of epochs (m) 100

Activation Function

Sigmoid **Hyperbolic Tangent**

Area Perimeter Major Length Minor Length Roundnes

Not Bias

Train **Classify**

User Input Testing

```
perception.py
preprocessing.py
Task2
  Dry_Bean_Dataset.csv
  evaluate_old.py
  main.py
  multi_layer_perceptron.py
  preprocessing.py
main.py
README.md
External Libraries
Scratches and Consoles

main x
:
Test Confusion Matrix:
[[20.  0.  0.]
 [ 0. 19.  1.]
 [ 0.  1. 19.]]
Test Accuracy: 0.9666666666666667
Sample Prediction: [[1 0 0]]
predicted class for ur Inputs is: BOMBAY
```

Number of hidden layers

Number of neurons

Learning rate (η)

Number of epochs (m)

Activation Function

Sigmoid **Hyperbolic Tangent**

Area **Perimeter** **Major Length** **Minor Length** **Roundness**

Bias

Train **Classify**

**THANK
YOU**