Lab Number: 6

Title
Illustrate Multilayer Perceptron

Objective
To implement and illustrate a Multilayer Perceptron (MLP) using Weka's built-in neural network classifier and analyze its performance.

IDE/Tools Used
Weka 3.8.6

Theory
Multilayer Perceptron: Multilayer perceptron or MLP is a fully connected feed-forward artificial neural network with:

- One input layer
- One or more hidden layers (user-defined)
- One output layer

Key features of MLP include:

- Uses backpropagation for training
- Employs sigmoid or tanh activation functions
- Can model complex non-linear relationships
- Universal function approximator

Important Parameters in Weka include:

- -L for Learning Rate
- -M for Momentum
- -N for Training Epochs
- -H for Hidden Layers
- Auto-build which automatically sets optimal hidden nodes

Implementation

The following steps were followed to execute MLP on dataset.

1. Load Dataset

Load the default or built-in dataset named “diabetes.arff”. To do so open Weka Explorer and in the preprocess tab click on open file and select the file and click on open. Also press on the edit button to visualize the data.

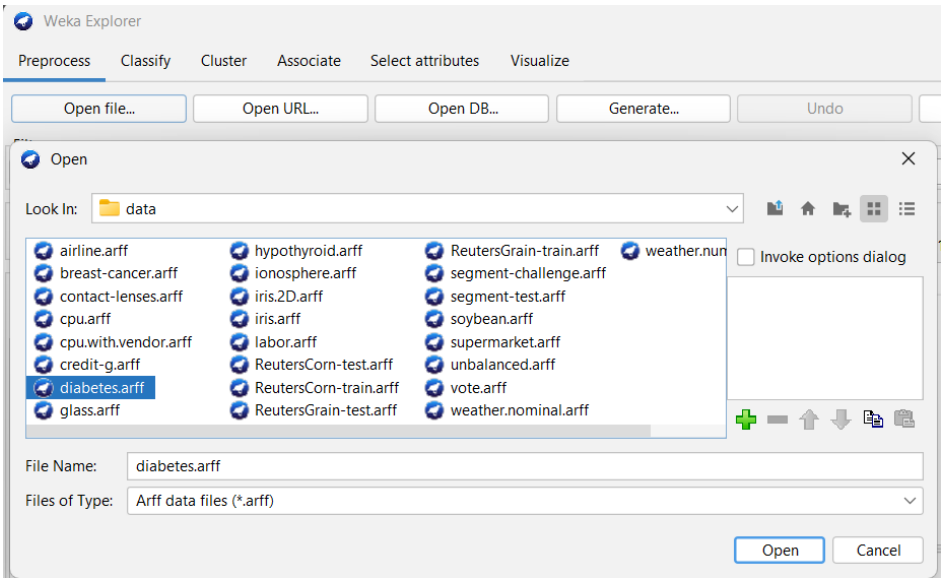
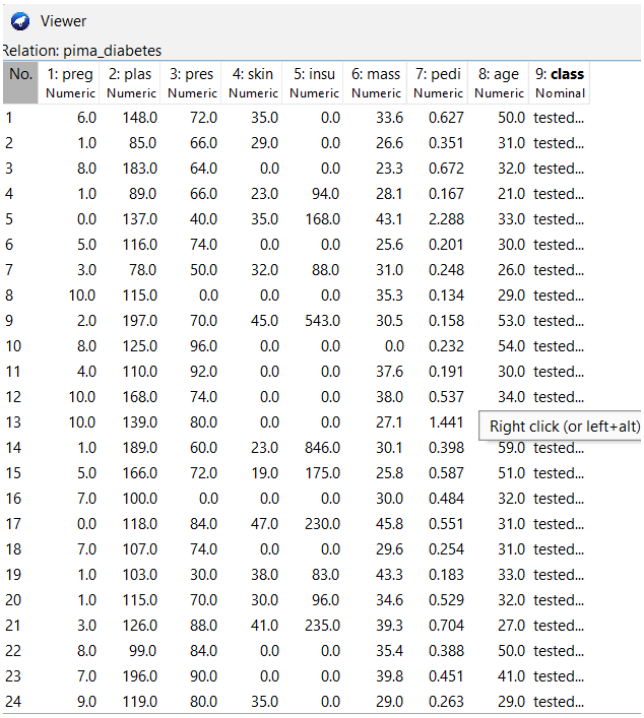


Figure 1: Opening the dataset in Weka Explorer



No.	1: preg Numeric	2: plas Numeric	3: pres Numeric	4: skin Numeric	5: insu Numeric	6: mass Numeric	7: pedi Numeric	8: age Numeric	9: **class** Nominal
1	6.0	148.0	72.0	35.0	0.0	33.6	0.627	50.0	tested...
2	1.0	85.0	66.0	29.0	0.0	26.6	0.351	31.0	tested...
3	8.0	183.0	64.0	0.0	0.0	23.3	0.672	32.0	tested...
4	1.0	89.0	66.0	23.0	94.0	28.1	0.167	21.0	tested...
5	0.0	137.0	40.0	35.0	168.0	43.1	2.288	33.0	tested...
6	5.0	116.0	74.0	0.0	0.0	25.6	0.201	30.0	tested...
7	3.0	78.0	50.0	32.0	88.0	31.0	0.248	26.0	tested...
8	10.0	115.0	0.0	0.0	0.0	35.3	0.134	29.0	tested...
9	2.0	197.0	70.0	45.0	543.0	30.5	0.158	53.0	tested...
10	8.0	125.0	96.0	0.0	0.0	0.0	0.232	54.0	tested...
11	4.0	110.0	92.0	0.0	0.0	37.6	0.191	30.0	tested...
12	10.0	168.0	74.0	0.0	0.0	38.0	0.537	34.0	tested...
13	10.0	139.0	80.0	0.0	0.0	27.1	1.441	57.0	tested...
14	1.0	189.0	60.0	23.0	846.0	30.1	0.398	59.0	tested...
15	5.0	166.0	72.0	19.0	175.0	25.8	0.587	51.0	tested...
16	7.0	100.0	0.0	0.0	0.0	30.0	0.484	32.0	tested...
17	0.0	118.0	84.0	47.0	230.0	45.8	0.551	31.0	tested...
18	7.0	107.0	74.0	0.0	0.0	29.6	0.254	31.0	tested...
19	1.0	103.0	30.0	38.0	83.0	43.3	0.183	33.0	tested...
20	1.0	115.0	70.0	30.0	96.0	34.6	0.529	32.0	tested...
21	3.0	126.0	88.0	41.0	235.0	39.3	0.704	27.0	tested...
22	8.0	99.0	84.0	0.0	0.0	35.4	0.388	50.0	tested...
23	7.0	196.0	90.0	0.0	0.0	39.8	0.451	41.0	tested...
24	9.0	119.0	80.0	35.0	0.0	29.0	0.263	29.0	tested...

Figure 2: Visualize the dataset

2. Apply Multilayer Perceptron

Select on the Classify tab and the Choose options and choose the functions MultilayerPerceptron option present inside the function option.

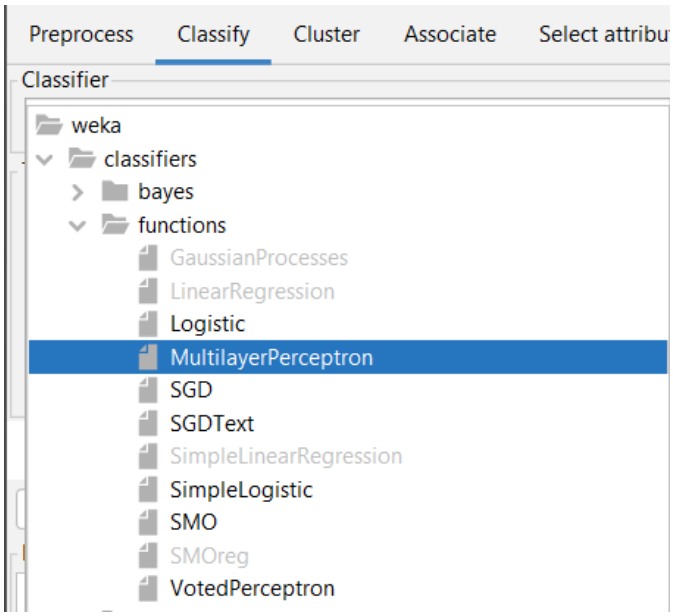


Figure 3: Selecting the MultilayerPerceptron option

3. Configure MLP

Click on MultilayerPerceptron and set the following parameters:

- hiddenLayers = a [auto = (8 attributes + 2 classes)/2 = 5 hidden nodes]
- learningRate = 0.3
- momentum = 0.2
- trainingTime = 1000 (epochs)
- autoBuild = True
- GUI = False

Leave other fields to their default and click on OK.

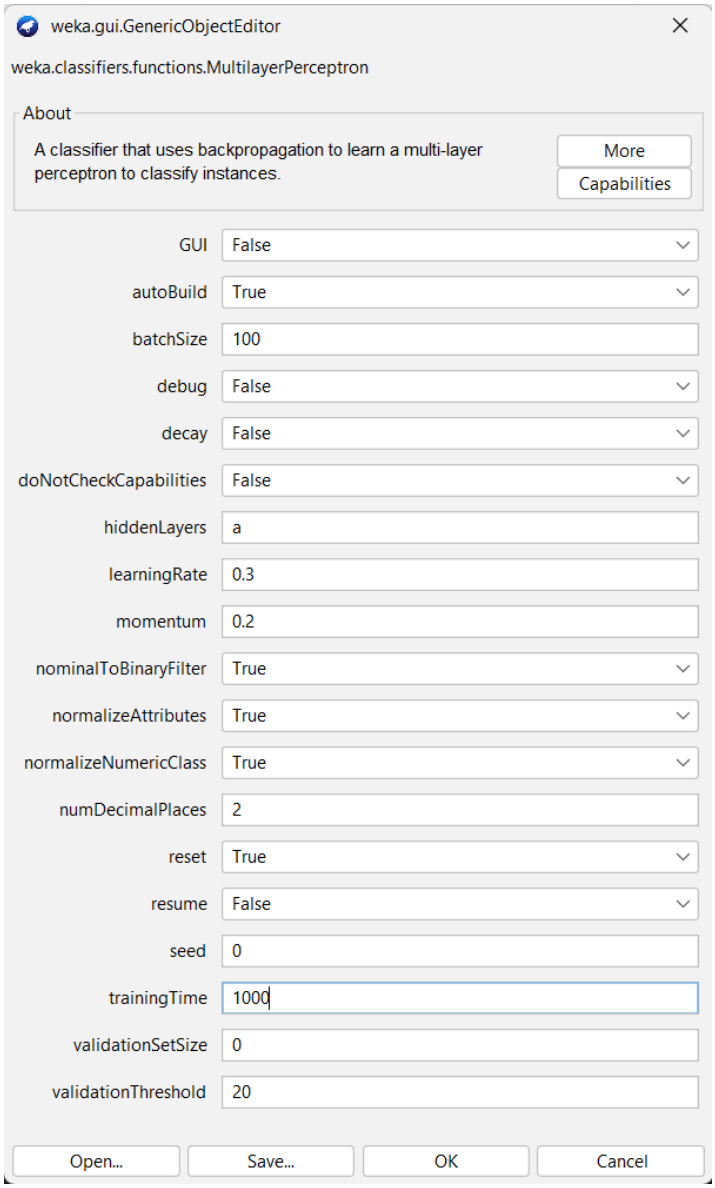


Figure 4: Settings for MLP

4. Test Options

Set the Cross-validation field to 10 folds and Click Start. WEKA will then generate the result in Classifier output section.

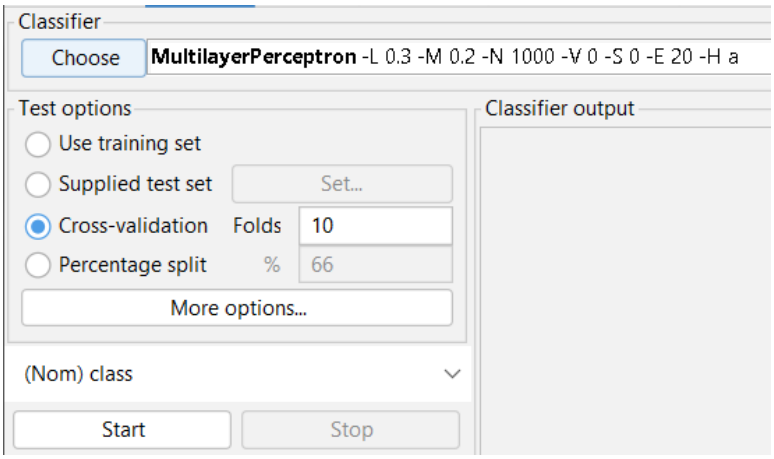


Figure 5: Setting the cross validation

Output:

```
=== Run information ===
Scheme:       weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N
1000 -V 0 -S 0 -E 20 -H a
Relation:     pima_diabetes
Instances:    768
Attributes:   9
              preg
              plas
              pres
              skin
              insu
              mass
              pedi
              age
              class
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

Sigmoid Node 0
    Inputs    Weights
    Threshold    -2.624286069298867
    Node 2    1.4138823475815228
    Node 3    9.686822102945891
    Node 4    2.2864949391672003
    Node 5    3.130727661417809
    Node 6    -3.0052164845409015
```

```
Sigmoid Node 1
    Inputs    Weights
    Threshold    2.624286069340684
    Node 2    -1.4138823475959446
    Node 3    -9.686822107366241
    Node 4    -2.2864949392137763
    Node 5    -3.130727661465087
    Node 6    3.0052164846383143
Sigmoid Node 2
    Inputs    Weights
    Threshold    -3.9405806254241162
    Attrib preg    -12.144356498027216
    Attrib plas    -12.628539057979989
    Attrib pres    3.578880554381653
    Attrib skin    3.1061815950072416
    Attrib insu    -10.914690025462486
    Attrib mass    -14.535326419883615
    Attrib pedi    -5.551575724318274
    Attrib age    14.805450391414613
Sigmoid Node 3
    Inputs    Weights
    Threshold    -10.90963316862385
    Attrib preg    1.5364665402353654
    Attrib plas    -4.020395954624456
    Attrib pres    5.6049570233510035
    Attrib skin    -11.137172826254371
    Attrib insu    0.551635590264021
    Attrib mass    -7.240200120761481
    Attrib pedi    15.429360331594044
    Attrib age    1.6314901902558805
Sigmoid Node 4
    Inputs    Weights
    Threshold    0.9029357140313415
    Attrib preg    1.1866260109410856
    Attrib plas    -18.80844207009834
    Attrib pres    -8.702536995103436
    Attrib skin    5.0932692497201675
    Attrib insu    -4.408004930215059
    Attrib mass    -12.550098005652977
    Attrib pedi    -6.990490359985216
    Attrib age    13.311155262952074
Sigmoid Node 5
    Inputs    Weights
    Threshold    -3.0626298401428516
    Attrib preg    11.92855029222295
    Attrib plas    -18.24495489494624
    Attrib pres    7.460216510364904
    Attrib skin    -1.373341882181475
    Attrib insu    3.8516570718218643
```

```
    Attrib mass    -6.577732294551473
    Attrib pedi    0.1335451930800631
    Attrib age    -25.396804834946206
Sigmoid Node 6
    Inputs    Weights
    Threshold    1.952528567321735
    Attrib preg    19.50262807455562
    Attrib plas    -8.385005035356954
    Attrib pres    -2.994444403097937
    Attrib skin    1.065244552462684
    Attrib insu    -2.5351325345775195
    Attrib mass    0.37644690937466013
    Attrib pedi    9.435195414908467
    Attrib age    -12.324834417705212
Class tested_negative
    Input
    Node 0
Class tested_positive
    Input
    Node 1

Time taken to build model: 0.39 seconds
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         576               75      %
Incorrectly Classified Instances       192               25      %
Kappa statistic                          0.4411
Mean absolute error                      0.2939
Root mean squared error                  0.4279
Relative absolute error                 64.6656 %
Root relative squared error             89.7754 %
Total Number of Instances              768

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.826    0.392    0.797      0.826    0.811      0.442    0.784     0.841     tested_negative
                 0.608    0.174    0.652      0.608    0.629      0.442    0.784     0.655     tested_positive
Weighted Avg.    0.750    0.316    0.747      0.750    0.748      0.442    0.784     0.776

=== Confusion Matrix ===

   a   b   <-- classified as
 413  87 |   a = tested_negative
 105 163 |   b = tested_positive
```

Figure 6: Summary result of the MLP performance

Discussion

The Multilayer Perceptron classifier performed reasonably well on the Diabetes dataset, achieving an overall accuracy of 75% with 10-fold stratified cross-validation. The model exhibited a strong ability to correctly identify patients who tested negative for diabetes, recording a true positive rate of 0.826 and precision of 0.797 for the tested_negative class. This indicates that when the neural network predicts "no diabetes," it is correct in approximately 80% of cases, making it reliable for screening out non-diabetic individuals.

For the tested_positive (diabetic) class, the recall was 0.608, meaning the model successfully detected around 61% of actual diabetic patients, while missing the remaining 39%. Although this sensitivity is moderate, it is a common outcome on this benchmark dataset due to class imbalance and the presence of biologically plausible zero values in several physiological attributes. The precision for the positive class was 0.652, showing that when the model predicts diabetes, it is correct in nearly two-thirds of cases. The weighted average F-measure of 0.748 and a Kappa statistic of 0.4411 reflect fair-to-moderate agreement beyond chance.

The area under the ROC curve of 0.784 demonstrates that the Multilayer Perceptron possesses good discriminative power overall, effectively separating the two classes despite the inherent challenges of the dataset. The network converged quickly (under 0.4 seconds for the full model) even with 1000 training epochs, confirming the efficiency of backpropagation on standardized numeric data.

Conclusion

The Multilayer Perceptron was successfully implemented and trained on the Diabetes dataset using Weka Explorer. With an automatically determined hidden layer size ("a" → 5 hidden neurons), learning rate 0.3, momentum 0.2, and 1000 epochs, the neural network delivered a stable classification model with 75% accuracy, a weighted ROC area of 0.784, and a Kappa of 0.4411. The experiment clearly illustrates the capability of Multilayer Perceptrons to learn complex non-linear relationships among physiological variables and produce meaningful predictive performance on real-world medical data. These results confirm the effectiveness of artificial neural networks as a powerful and practical tool for binary classification tasks in healthcare and predictive analytics.