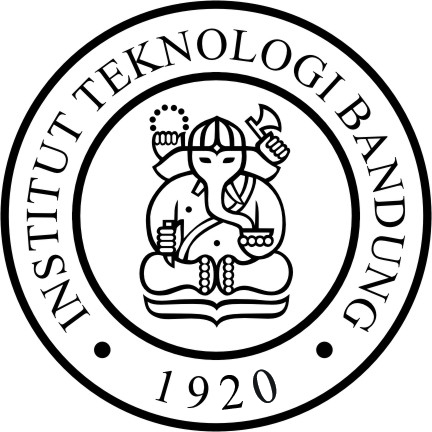
**TUGAS IMPLEMENTASI NEURAL NETWORK**

**IF4071 MACHINE LEARNING**



DISUSUN OLEH:

JAN WIRA GOTAMA PUTRA 13512015

MARIO FILINO 13512055

MELVIN FONDA 13512085

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

TEKNIK INFORMATIKA

2015

# **Hasil Eksekusi**

## **Multilayer Perceptron**

//Configuration

learningRate = 0.1;  
momentum = 0.0;  
mseThreshold = 0.0001;  
maxIteration = 10000;

normalizeAttribute = yes;

activationFunction = Sigmoid;

topology = 1 hidden layer with 3 neurons

randomInitialWeight = yes

evaluation = using test set

### Weather Numeric

NEURAL NETWORK TOPOLOGY

-----------------------------

Hidden Layer 0

-----------------

Node 0 Activation Function = SIGMOID

Bias Weight = -0.2212319162268913

Weight from previous layer neuron0 = -2.783168004000769

Weight from previous layer neuron1 = 0.3088886560666097

Weight from previous layer neuron2 = 2.4780858803884436

Weight from previous layer neuron3 = 0.9907833059597125

Weight from previous layer neuron4 = 5.160125201120365

Weight from previous layer neuron5 = 1.5456170847012958

Node 1 Activation Function = SIGMOID

Bias Weight = 1.1572019305698449

Weight from previous layer neuron0 = 2.0666740111266306

Weight from previous layer neuron1 = 2.86637377995684

Weight from previous layer neuron2 = -3.1356153273624487

Weight from previous layer neuron3 = -1.2477991383847993

Weight from previous layer neuron4 = -5.221674721242067

Weight from previous layer neuron5 = -1.2484890200275776

Node 2 Activation Function = SIGMOID

Bias Weight = -1.6477615204831906

Weight from previous layer neuron0 = -5.506767220188397

Weight from previous layer neuron1 = 5.313130346342439

Weight from previous layer neuron2 = -0.43840711053143067

Weight from previous layer neuron3 = -2.152403554919713

Weight from previous layer neuron4 = -1.59016380314574

Weight from previous layer neuron5 = 7.153827230940083

-----------------

Output Layer

-----------------

Activation Function = SIGMOID

Bias Weight = -0.20734364520026652

Weight from previous layer neuron0 = -4.713801846383771

Weight from previous layer neuron1 = 5.336332923323059

Weight from previous layer neuron2 = 9.232533942779924

Activation Function = SIGMOID

Bias Weight = 0.4682929211210917

Weight from previous layer neuron0 = 4.4514597697567435

Weight from previous layer neuron1 = -5.587191391811598

Weight from previous layer neuron2 = -9.220931794215733

-----------------

Correctly Classified Instances 14 100 %

Incorrectly Classified Instances 0 0 %

Kappa statistic 1

Mean absolute error 0.0149

Root mean squared error 0.0172

Relative absolute error 3.2102 %

Root relative squared error 3.592 %

Total Number of Instances 14

### Weather Nominal

NEURAL NETWORK TOPOLOGY

-----------------------------

Hidden Layer 0

-----------------

Node 0 Activation Function = SIGMOID

Bias Weight = -2.231055988590537

Weight from previous layer neuron0 = -1.7660467688284978

Weight from previous layer neuron1 = 1.849748417145692

Weight from previous layer neuron2 = -1.7312190596144124

Weight from previous layer neuron3 = -0.6002634916575913

Weight from previous layer neuron4 = 0.8996362303942665

Weight from previous layer neuron5 = -1.151715638078737

Weight from previous layer neuron6 = 4.115086837512908

Weight from previous layer neuron7 = 2.1356596287605143

Node 1 Activation Function = SIGMOID

Bias Weight = 1.2470827819196308

Weight from previous layer neuron0 = 3.2126982450325987

Weight from previous layer neuron1 = -0.6114448595307888

Weight from previous layer neuron2 = -0.7531623480225588

Weight from previous layer neuron3 = 0.545642737037223

Weight from previous layer neuron4 = -0.6593872236732067

Weight from previous layer neuron5 = 1.3460025982929773

Weight from previous layer neuron6 = 1.2409922960606943

Weight from previous layer neuron7 = -1.1191617312568383

Node 2 Activation Function = SIGMOID

Bias Weight = -3.70557282860813

Weight from previous layer neuron0 = -4.262943559802429

Weight from previous layer neuron1 = 3.3349689810125063

Weight from previous layer neuron2 = -2.0775929223118963

Weight from previous layer neuron3 = -0.7869392826308839

Weight from previous layer neuron4 = 1.6222170977264414

Weight from previous layer neuron5 = -3.5799022913238963

Weight from previous layer neuron6 = 7.989884752196615

Weight from previous layer neuron7 = 5.038450852164147

-----------------

Output Layer

-----------------

Activation Function = SIGMOID

Bias Weight = -3.0844680069925237

Weight from previous layer neuron0 = 3.914142675502906

Weight from previous layer neuron1 = -3.625345086748739

Weight from previous layer neuron2 = 9.206282406964515

Activation Function = SIGMOID

Bias Weight = 2.8863095344451235

Weight from previous layer neuron0 = -4.001334365397016

Weight from previous layer neuron1 = 3.844296502396978

Weight from previous layer neuron2 = -9.105116289412893

-----------------

Correctly Classified Instances 14 100 %

Incorrectly Classified Instances 0 0 %

Kappa statistic 1

Mean absolute error 0.011

Root mean squared error 0.0154

Relative absolute error 2.3769 %

Root relative squared error 3.2157 %

Total Number of Instances 14

### Iris

NEURAL NETWORK TOPOLOGY

-----------------------------

Hidden Layer 0

-----------------

Node 0 Activation Function = SIGMOID

Bias Weight = 1.7050050010462219

Weight from previous layer neuron0 = -1.6053858173147002

Weight from previous layer neuron1 = 3.3673095761436254

Weight from previous layer neuron2 = -5.243317538766451

Weight from previous layer neuron3 = -4.8525359746623975

Node 1 Activation Function = SIGMOID

Bias Weight = -23.087580978279785

Weight from previous layer neuron0 = -2.163185151385694

Weight from previous layer neuron1 = -7.923296032831107

Weight from previous layer neuron2 = 18.580237774185218

Weight from previous layer neuron3 = 21.963412394205758

Node 2 Activation Function = SIGMOID

Bias Weight = -2.405727162513938

Weight from previous layer neuron0 = 2.768979430404254

Weight from previous layer neuron1 = -2.9669264423277615

Weight from previous layer neuron2 = 5.774718494581189

Weight from previous layer neuron3 = 5.502658213133793

-----------------

Output Layer

-----------------

Activation Function = SIGMOID

Bias Weight = 0.6880390863821987

Weight from previous layer neuron0 = 6.025991958851746

Weight from previous layer neuron1 = -5.5824956619240025

Weight from previous layer neuron2 = -6.97907440436522

Activation Function = SIGMOID

Bias Weight = 0.1473532373630191

Weight from previous layer neuron0 = -6.697831449096516

Weight from previous layer neuron1 = -19.32870727932157

Weight from previous layer neuron2 = 8.596343647664167

Activation Function = SIGMOID

Bias Weight = -6.9788519046724415

Weight from previous layer neuron0 = -4.092824942323376

Weight from previous layer neuron1 = 19.288347805036523

Weight from previous layer neuron2 = -1.5942358132735421

-----------------

Correctly Classified Instances 148 98.6667 %

Incorrectly Classified Instances 2 1.3333 %

Kappa statistic 0.98

Mean absolute error 0.0134

Root mean squared error 0.0914

Relative absolute error 3.0075 %

Root relative squared error 19.3967 %

Total Number of Instances 150

## **Single Perceptron Configuration**

learningRate = 0.1;

mseThreshold = 0.01;//bila mse masih besar, lanjutkan iterasi

maxIteration = 10;

initialWeight = 0.0;

useNormalization = true; //normalize attribute

## **Single Perceptron – Perceptron Training Rule**

### Weather Numeric

Last Weight: [-0.200000, 0.000000, -0.400000, 0.200000, 0.266667, 0.296774, -0.600000]

Correctly Classified Instances 12 85.7143 %

Incorrectly Classified Instances 2 14.2857 %

Kappa statistic 0.6585

Mean absolute error 0.1429

Root mean squared error 0.378

Relative absolute error 30.7692 %

Root relative squared error 78.8263 %

Total Number of Instances 14

### Weather Nominal

Last Weight: [0.400000, 0.400000, -0.600000, 0.600000, 0.400000, -0.200000, 0.200000, -0.800000, -0.400000]

Correctly Classified Instances 13 92.8571 %

Incorrectly Classified Instances 1 7.1429 %

Kappa statistic 0.8511

Mean absolute error 0.0714

Root mean squared error 0.2673

Relative absolute error 15.3846 %

Root relative squared error 55.7386 %

Total Number of Instances 14

### Iris

Not Supported

## **Single Perceptron – Delta Rule Batch**

### Weather Numeric

Last Weight: [-333.801938, -125.703796, -90.118883, -117.979258, -166.080442, -193.404278, -221.924883]

Correctly Classified Instances 9 64.2857 %

Incorrectly Classified Instances 5 35.7143 %

Kappa statistic 0

Mean absolute error 0.3571

Root mean squared error 0.5976

Relative absolute error 76.9231 %

Root relative squared error 124.6354 %

Total Number of Instances 14

### Weather Nominal

Last Weight: [-453.553583, -159.611254, -121.979192, -171.963137, -122.159073, -190.344633, -141.049877, -253.861411, -289.513848]

Correctly Classified Instances 9 64.2857 %

Incorrectly Classified Instances 5 35.7143 %

Kappa statistic 0

Mean absolute error 0.3571

Root mean squared error 0.5976

Relative absolute error 76.9231 %

Root relative squared error 124.6354 %

Total Number of Instances 14

### Iris

Not Supported

## **Single Perceptron – Delta Incremental**

### Weather Numeric

Last Weight: [0.186176, 0.225840, -0.238001, 0.198338, 0.210396, 0.360829, -0.398020]

Correctly Classified Instances 10 71.4286 %

Incorrectly Classified Instances 4 28.5714 %

Kappa statistic 0.3171

Mean absolute error 0.2857

Root mean squared error 0.5345

Relative absolute error 61.5385 %

Root relative squared error 111.4773 %

Total Number of Instances 14

### Weather Nominal

Last Weight: [0.447144, 0.286206, -0.251215, 0.412152, 0.385618, -0.037430, 0.098956, -0.365548, -0.336561]

Correctly Classified Instances 13 92.8571 %

Incorrectly Classified Instances 1 7.1429 %

Kappa statistic 0.8372

Mean absolute error 0.0714

Root mean squared error 0.2673

Relative absolute error 15.3846 %

Root relative squared error 55.7386 %

Total Number of Instances 14

### Iris

Not Supported

# **Deskripsi**

Ekplorasi menggunakan library weka telah dikuasai dengan baik. Implementasi *Single Perceptron* diuji menggunakan data latihan kelas (excel tanpa normalisasi) dan sudah benar. Implementasi *Multilayer Perceptron* diuji dengan membandingkan dengan ANN pada weka dan menghasilkan performa yang cukup mirip. Impelmentasi *Multilayer Perceptron* (*custom*) sudah mampu dimodifikasi untuk banyak *hidden layer*, maupun kelas *numeric*. Apabila kelas merupakan kelas *numeric*, maka *neuron* pada *output layer* akan memiliki fungsi aktivasi *linear*.