n r l n twor p for t v

Us r's G i

V rsion: 0.1.9.1

Mi e D S mid

nu 2 9

Contents

1 Introduction **2**

1.1 Compatibility to Matlab's ^{TM} Neural Network Toolbox . . . 2

1.2 Version numbers . . . 2

1.3 Known incompatibilities . . . 2

1.3.1 Function names . . . 2

2 Neural Network Package for Octave **3**

2.1 Available Functions . . . 3

2.1.1 min max . . . 3

2.1.2 new . . . 3

2.1.3 prestd . . . 4

2.1.4 poststd . . . 4

2.1.5 saveMLPStruct . . . 4

2.1.6 sim . . . 4

2.1.7 subset . . . 4

2.1.8 train . . .

2.1.9 trastd . . . 6

2.2 Transfer functions . . . 6

2.2.1 logsig . . . 6

2.2.2 purelin . . . 6

2.2.3 tansig . . . 7

3 Examples

hapte 1

I r u i

1.1 Co pati ility to Matla ' rall t or ooll ox

hapte

eur l e w r ge f r e

his hapter des ribes all fun tions a ailable in the neural network pa ka e of ta e. E en thou h it will be as ompatible as possible to the one of B( ).

2.1 ailable n tion

2.1.1 min max

min_ma

PF: Performance function default = "mse"

Example :

```
net = new ( x2 [2 1 )
net = new ( x2 [2 1 {"tansig" "purelin"});
net = new ( x2 [2 1 {"tansig" "purelin"} "trainlm");
net = new ( x2 [2 1 {"tansig" "purelin"} "trainlm" "notUsed" "mse");
```

Comment :

In this version you can have as much output neurons as you want. The same with the number of hidden layers. This means you can have one input layer, unrestricted number of hidden layers and one output layer. That's it.

2.1. nest

2.1. stst

2.1. save LP struct

Description:

Left Hand Side:
m_train: (+) x matrix with input rows output rows and columns where = N.
m_test: (+) x matrix with input rows output rows and columns where = N.
m_all: (+) x U matrix with input rows output rows and U columns where U = N. nd U an
only exist if also exist.

an et data fa trainin
[Nat used ai ht naw anll fa aampatibilit with atlab
[Nat used ai ht naw anll fa aampatibilit with atlab
allidatian data. Cantains input and tan et alues

Exa plle :

net = train(Pnet P)
net = train(Pnet P [[ )

P must ha e the same333(sai J3(m)2b)-
.[(P)83 must ha e the same333(sai J3(m)2b)-

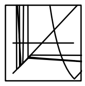

Figure 2.4: Linear transfer function [illegible]

2.2. tansi

hapte

le

.1 xa pl 1

u an find this example in the *t sts* *LP* di e t f ea h elease f m the sub e si n ep sit . I will d (e less) a line b line walkth u h s after this sh uld be e e thin lea . I assume that u ha e s e expe ien e with multi a e pe epti ns.

.1.1 Int ucti n

(w)-23(9) a 43 8 ed with a d B eas (e)24 7(w)78(e)244 n-28 w e

in eas n de eas n su fa h w an p e a ult- a e (ept)-316((sh t .)-46(s.a)-317 this)-316(ea s)-317an

with with.

```
[nR w , nC lumn ] = ize(m ata);
```

```
    # th  la t c lumn i  th   ut ut,
    # r m    c lumn  , 8 and 12
    # 8  r w .

mOut ut = m ata(:,em );
mIn ut = m ata(:,1:em  1);
mIn ut(:,[  8 12]) = []; # d l t  c lumn  , 8 and 12

m ata = [mIn ut mOut ut];

# n w   lit th  data matrix in    i c  , train data, t];
m ata
# th inith  tr, ut tand[ 5d(m (i) ata)]T0g0 0.250. 80.5 rg0.250. 80.5 R 0 11. 55Td[( 10 8)]T0g0
traini
#i th #18 1 and
(];)]T0g0 0.250. 80.5 rg0.250. 80.5 R [ 525(#) 51/2  ,# 8  =i  trair
5ata @ata=
```

```
00082 [mT  InputN] = ra  (mT  (: nd ,:),cM anInput,cSdInput);
0008
0008  [ imOut] = i  (n ,mT  InputN);
00085  imOut
```

.2. alkth u h

he d ene e t the example starts below the line number 3 .

i li gr y

[1 John . Eaton

NU ta e anual Edition 3 P D - ersion ebruar 1997

[2 he ath orks In .

B Help B ersion 7.1 (14 P3) Neural Network oolbox ersion 4. .6 (14 P3)

[3 Christopher . Bishop

Neural Networks for Pattern e o nition U Uni ersit Press reat Clarendon treed xford 2 6 P BN -19-8 3864-2 2 2

[4 artin . Ha en Howard B. Demuth ark H. Beale

NEU NE K DE N P Publishin Compan 2 Park Pla a Boston 2116-4324 BN 3494332-2 1996