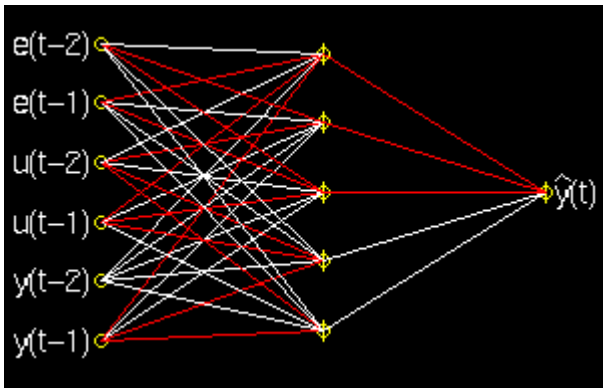
THE NNSYSID TOOLBOX

- FOR USE WITH MATLAB



VERSION 2

CONTENTS

The NNSYSID toolbox is a set of MATLAB tools for neural network based identification of nonlinear dynamic systems. The toolbox contains a number of m and MEX-files for training and evaluation of multilayer perceptron type neural networks within the MATLAB environment. There are functions for training of ordinary feedforward networks as well as for identification of nonlinear dynamic systems and for time-series analysis. Version 2 requires MATLAB 5.3 or higher. For MATLAB 4.2-MATLAB 5.2 it is possible to use the old Version 1.1. In this case the Signal Processing Toolbox must be available. The toolbox is completely independent of the Neural Network Toolbox and the System Identification Toolbox.

The toolbox contains:

- Fast, robust, and easy-to-use training algorithms.
- A number of different model structures for modelling of dynamic systems.
- Validation of trained network models.
- Estimation of the models's generalization ability.
- Demonstration programs.

HOW CAN I LEARN THE THEORY?

The book

Neural Networks for Modelling and Control of Dynamic Systems
by Magnus Nørgaard, O. Ravn, N. K. Poulsen, and L. K. Hansen

is available on Springer-Verlag, London, in the series *Advanced Textbooks in Control and Signal Processing*.

Visit our homepage for the book

LET ME HAVE IT...

Version 2

The toolbox will work under Matlab 5.3 and Matlab 6. No "official" toolboxes are required. Version 2 is not backward compatible with Version 1.1. The toolbox has been zipped into a file of approximately 1.5 Mbytes. This file contains the manual in Postscript and PDF-formats.

Download matrix

General version (zip)	From Windows: Use Winzip From DOS : pkunzip nnsysid20.zip From UNIX: unzip -a nnsysid20.zip
Alternative unix version (gzip+tar)	Use "gunzip nnsysid.tar.gz" followed by "tar -xvf nnsysid.tar" to unpack
Compiled MEX files (zip)	Compiled Mex files for Windows, Intel/Linux, and HPUX

Version 1.1

The toolbox has been compressed and packed into a "zip" file of approximately 0.53 Mbytes. From the matrix below you can download different versions of the toolbox.

Download matrix

Matlab 4.2	General version (zip)	From DOS : pkunzip nnsysid.zip From UNIX: unzip -a nnsysid.zip
Matlab 4.2	Alternative unix version (gzip+tar)	Use "gunzip nnsysid.tar.gz" followed by "tar -xvf nnsysid.tar" to unpack
Matlab 4.2	Alternative PC version (zip)	use pkunzip sysidpc.zip to unzip
Matlab 5	General version (zip)	From DOS : pkunzip nnsysid5.zip From UNIX: unzip -a nnsysid5.zip
Matlab 5	Alternative unix version (gzip+tar)	Use "gunzip nnsysid5.tar.gz" followed by "tar -xvf nnsysid5.tar" to unpack

NOTICE that there is a special PC-version for MATLAB 4.2 As explained in the release notes the "printf" statements works differently under Unix and Windows 3.1. The PC version contains the toolbox with the suggested modification for PCs. Under MATLAB5/Windows 95 this problem has been eliminated.

It appears that problems occur when trying to print the manuals on certain printers. I have therefore used the unix-command 'ps2pdf' to convert the manuals to pdf-format. View tutorial section or reference section. The manuals are included in postscript format in the zip-files above.

MEX files for version 1.1

All functions in the toolbox have been implemented as M-functions. However, to speed up some of the most time consuming functions, a few dublets have been implemented in C and can be compiled to MEX-files. For users that do not have access to a compiler or can't figure out how to use their compiler I have precompiled the MEX-files for a few platforms

Download MEX files		
Matlab 4.2	MEX compiled on HP735 workstation (zip)	Use "unzip hpmex4.zip" to unpack
Matlab 5	MEX compiled on HP735 workstation (zip)	Use "unzip hpmex5.zip" to unpack
Matlab 5	MEX compiled on PC running Win95	Use "pkunzip pcmex5.zip" or "Winzip" to unpack

WHAT'S NEW IN VERSION 2?

Several things have changed in Version 2. This means that hardly any of the functions will be compatible with Version 1.1. However, only minor changes in the function calls must be made. Some of the major new features are:

- The toolbox is no longer dependent on the Signal Processing Toolbox.
- The training is more automatic (better stopping criteria have been introduced).
- Easier call of training algorithms.
- Options to training algorithms changed to an object oriented like fashion.
- Bug fixes and fine-tuning.

PLEASE NOTE

Please bear with me. This is not a commercial product and thus I cannot spare the time for supporting it. BUT, if you should find a major bug do let me know and hopefully I can correct it in a future release.

We encourage all users of the NNSYSID toolbox to write us about their successes (and failures?). We are very interested in hearing where the toolbox is used and for what type of applications. Since your comments very well may influence future releases of the toolbox this is also in your own interest! You can e-mail your experiences to the address listed at the bottom of this page.

AN ADD-ON FOR CONTROL DESIGN

If you are interested in neural networks for control we recommend that you download our NNCTRL toolkit. See our NNCTRL toolkit page for supplementary information.

FUNCTION OVERVIEW

The toolbox functions grouped by subject

FUNCTIONS FOR TRAINING	
batbp	Batch version of the back-propagation algorithm
incbp	Recursive (/incremental) version of back-propagation
igls	Iterated Generalized Least Squares training of multi-output nets
marq	Levenberg-Marquardt method
marqlm	Memory-saving implementation of the Levenberg-Marquardt method
rpe	Recursive prediction error method

FUNCTIONS FOR PREPARATION OF DATA	
dscale	Scale data to zero mean and variance one

FUNCTIONS FOR TRAINING MODELS OF DYNAMIC SYSTEMS	
lipschit	Determine the lag space
nnarmax1	Identify a Neural Network ARMAX (or ARMA) model (Linear MA filter)
nnarmax2	Identify a Neural Network ARMAX (or ARMA) model
nnarx	Identify a Neural Network ARX (or AR) model
nnarxm	Identify a multi output Neural Network ARX (or AR) model.
nnigls	Iterated Generalized LS training of multi-output NNARX models.
nniol	Identify a Neural Network model suited for I-O linearization control
nnoe	Identify a Neural Network Output Error model
nnrarmx1	Recursive counterpart to NNARMAX1
nnrarmx2	Recursive counterpart to NNARMAX2
nnrarx	Recursive counterpart to NNARX
nnssif	Identify a NN State Space Innovations form model

FUNCTIONS FOR PRUNING NETWORKS	
netstruc	Extract weight matrices from matrix of parameter vectors
nnprune	Prune models of dynamic systems with Optimal Brain Surgeon (OBS)
obdprune	Prune feed-forward networks with Optimal Brain

	Damage (OBD)
obsprune	Prune feed-forward networks with Optimal Brain Surgeon (OBS)

FUNCTIONS FOR EVALUATING TRAINED NETWORKS	
fpe	FPE estimate of the generalization error for feed-forward nets
ifvalid	Validation of models generated by NNSSIF
ioleval	Validation of models generated by NNIOL
kpredict	k-step ahead prediction of dynamic systems.
loo	Leave-One-Out estimate of generalization error for feed-forward nets
nneval	Validation of feed-forward networks (trained by marq,rpe,bp)
nnfpe	FPE for I/O models of dynamic systems
nnloo	Leave-One-Out estimate for NNARX models.
nnsimul	Simulate model of dynamic system from sequence of inputs
nnvalid	Validation of I/O models of dynamic systems
wrescale	Rescale weights of trained network
xcorrel	Calculates high-order cross-correlation functions

MISCELLANOUS FUNCTIONS	
crossco	Calculate correlation coefficients.
drawnet	Draws a two layer neural network
getgrad	Derivative of network outputs w.r.t. the weights
pmntanh	Fast tanh function
settrain	Set parameters for training algorithms.

DEMOS	
test1	Demonstrates different training methods on a curve fitting example
test2	Demonstrates the NNARX function
test3	Demonstrates the NNARMAX2 function
test4	Demonstrates the NNSSIF function
test5	Demonstrates the NNOE function
test6	Demonstrates the effect of regularization by weight decay
test7	Demonstrates pruning by OBS on the sunspot benchmark

problem

MORE INFORMATION

For more information, please contact

toolbox@magnusnorgaard.dk

LINKS

- Me
- Homepage for the book "Neural Networks for Modelling and Control of Dynamic Systems.
- The MathWorks, Inc. The creator of MATLAB.
- MATLAB Central, file exchange. A lot of free MATLAB functions available for downloaded
- NNCTRL toolkit. An add-on to NNSYSID for control system design.
- ENTOOL. is a software package for ensemble regression modelling. The toolbox uses a subset of the NNSYSID toolbox.
- Some demo-files for "Predictive Control with Neural Networks" provided by Alexander Bollig.
- NNCTRL toolkit. An add-on to NNSYSID for control system design.
- The Automation Section at Ørsted-DTU.
- Section for Digital Signal Processing at the Department of Informatics and Mathematical Modelling (IMM), DTU. Main neural network group at DTU. Download publications etc.

Last Updated at 04/02/2003 16:33:32
Mail to Web Master
Ole Ravn