What is NNMODEL

NNMODEL is a cost effective way of modeling process data, statistical experiments, or historical databases. It can find from simple linear to complex non-linear relationships in empirical data. It is easy to use because it **automatically constructs** mathematical models directly from your data. It enables you to create prototype models quickly and inexpensively.

NNMODEL is designed to help you get maximum benefit from powerful **neural network** modeling techniques without requiring you to learn a complicated software package or statistical language. Thus, you can learn how to use NNMODEL and start solving real world problems within a few hours.

NNMODEL currently contains program modules to:

- Design a statistical experiment - NNMODEL allows you to create a data matrix based on a statistically designed experiment. A designed data matrix will allow you to squeeze the most information from a finite number of observations. The types of designs available are: two level, three level, simplex, star-simplex, central composite and multilevel.
- Keyboard enter, file or clipboard import the data - There are three methods for entering data into NNMODEL: 1) Enter the data directly using the built in data matrix editor, 2) import an ASCII tab or blank delimited file or 3) paste data from the Windows clipboard.
- Run simple statistics and correlation reports - You can generate a report that contains the basic statistics, such as, number of observations, maximum, minimum, average, standard deviation and sum of squares. Or generate a correlation report contains the results Pearson Correlation Coefficients, Probability > |R| under Ho and Rho:=0 / N.
- Graphically analyze the raw data - You can view the data graphically using a variety of plotting routines including: trend plot by observation, XY scatter, frequency distribution, 3 dimensional scatter. Thumbnail views of all the data can be printed for the trend, scatter and distribution plots.
- Load historical data into a designed experiment matrix - A designed data matrix can be created as an empty shell and later loaded by the historical data loader. This imposes a designed experiment onto the historical data to better insure any resulting model's long term success. This method also has two side benefits, you get to see how much of the design space is really represented in the data and it generates a smaller training matrix so the training step proceeds faster.
- Advice on missing observations - After historical data has been loaded into a designed experiment the Missing Advisor can be used to suggest trials or treatments to run that would balance the design space. Thus, extracting more information from the data.
- Add equations or calculated columns to the data matrix - Columns of data can be created by defining an equation based on the other columns. A simple equation parser is built into the data matrix editor. Rows of data can be excluded from reports, graphs or models by using an exclude equation.
- Model the data using neural networks - The whole purpose of NNMODEL is to build neural models. A model can be created and trained in just a few minutes.
- Interrogate the model interactively - After a model has been trained you can immediately ask the model to predict using combination of input levels not seen in the data.

- <u>Analyze the model's performance statistically</u> - A model's performance can be evaluated using standard R square statistics.
- <u>Display the model's predictions graphically including 3D and contour plots</u> - A number of graphs are available for validating a model including: measured vs. predicted, measured overlaid on predicted, residual plots, trends, scatter plots, frequency distributions, XY plots, 3D surface maps and contour plots.
- <u>Test the model on additional external data sets</u> - a test matrix can be loaded from data matrices not originally used to generate the model. This type of testing may be the only way of validating models generated from undesigned data.
- <u>Perform sensitivity analysis</u> - This analysis can show you how sensitive an output variable is to changes made to the inputs. The results are ranked in order with the variables with the most effect at the top of the list.
- <u>Export the neural model as a transportable ASCII file</u> - Trained models can be exported from NNMODEL to any other hardware platform. Neural models can be included with user software by linking with the NNLIB library.

Planned add-ons to NNMODEL:
- <u>Multi-Module Optimizer</u> Combine one or more neural models with algebraic equations to minimize or maximize any combination of inputs, outputs or cost functions. The optimizer utilizes a Monte Carlo started constrained conjugate gradient algorithm to minimize the objective function. The objective function can be constructed from any or all inputs or outputs along with their polarity (min or max) and their relative weight. Inputs can be constrained rectilinearly, outputs are constrained by a penalty function. Results of the optimizations can be viewed using the interactive interrogation module, graphically or by viewing results log.
- <u>Data Mining Utility</u> Allows the user to automatically set up a historical data matrix, identify variables as factors, responses or unknown, time position (up or down stream) in time units, use full dataset for modeling or select records from the database based on goodness of fit to a multi-level design, pick the best factors for inclusion into the model based on model performance, include or exclude factors for any model based on prior knowledge, report results of search. Possible RISC based add-on processor.
- <u>Multi-Module Simulator</u> Combine one or more neural models with interpreted algebraic equations or pre-compiled user subroutines (user creates a DLL file). Simulator is an OLE container that can link with many graphical display modules and VBX controls. The simulator is designed using the source/sink concept. Data sources are ASCII files, OLE or DDE modules, models or equations. Sinks are reports, graphs, meters, equations or models.
- <u>Attribute Data</u> Automatic conversion of Attribute data to a continuous variable based on a user defined rank or conversion to discrete logical variables (1 or 0). The continuous variable simply becomes one input to the model. However, the discrete variable creates as many inputs as there are states.
- <u>Real Time Data Matrix Loader</u> Using the DDE interface automatically load a designed data matrix. The data matrix can be exported to be used to build a neural model. The neural model can then be used to control the process monitored by the DDE source.
- <u>OODB Linkage</u> Allow the data matrix to be created directly from an open OODB database such as Microsoft Access.