Contents

About the aiNet:
Features
Terms
How to use the aiNet:

Menu commands:
Static menus: (what is a static menu?)
File Menu
View Menu
Options Menu
Window Menu
Help Menu

Dynamic menus: (what is a dynamic menu?)
Edit Menu (Model Vectors View)
Edit Menu (Prediction View)
Edit Menu (Error Distribution View)
Model Vectors Menu (Model Vectors View)
Prediction (Prediction View)
Error Distribution(Error Distribution View & Charts View)

Toolbar:
Toolbar commands

Registration:
All about registration

What Is a Static Menu

Static menus are always present in the menu bar and are almost always accessible. See also <u>dynamic menus</u>.

What Is a Dynamic Menu

Dynamic menus are view-depended. When the **Model Vector View** window is active, for example, than **Edit** menu and **Model Vectors** menu appear on the menu bar and they disappear when some other view becomes active.
See also static menus.

Features of the aiNet

The aiNet application is a very powerful and a very simple tool for solving the problems which are usually solved with artificial neural networks (ANN). All possible tests we had run proved that the results obtained with the aiNet are at least as good as the results obtained with some other ANNs. Let us state some of the aiNets features.

1. The major attribute that distinguishes the aiNet from other ANNs is the analysis speed. Since the aiNet uses an algorithm, which does not require any learning phase, the answers about prediction can be obtained almost immediately.

2. There is also only one coefficient (penalty coefficient), which has a major effect on the results. If we neglect some aspects, we can claim that knowing the right value for this coefficient solves the entire problem.

3. The aiNet can dynamically change the knowledge base. This means that you can add a new data to   neural network (or remove old one), add additional parameters (or remove old ones) and still get answers right away -- there is no time consuming learning phase.

4. Our experiments show us that noisy data is the aiNets favorite. If the data is just noisy, the aiNet will give you excellent results. When you have chaotic data (and you do not know that), than you can not obtain any solution, still the aiNet will assure you   that something is wrong with the data.

5. The aiNet provides you a way to estimate the rate of error in your prediction. If your problem is smooth, i.e. without noise, than this error will represent an estimation for the error in the predicted result. If you have noisy data, than this will represent an estimation for the noise around the predicted result. This means that an error estimation behaves locally.

6. aiNet is very suitable to work with missing values in your data. In real life problems it is usually very difficult to find a perfectly assembled knowledge base -- there is always some data missing. The aiNet handles missing data automatically and you need not worry about how to represent such data.

7. The aiNets graphical user interface is very simple to use. It looks like a spreadsheet application and if you are familiar with any other spreadsheet, the aiNet would not present a problem for you. Almost everything is only a mouse click away, menus are simple and there are also speed buttons.

8. On-line help is there for you. If you are stuck and do not know what to do just press F1 or select Help button and the aiNet will help you find the way out.

9. Several charts are also available. They represent the most natural way to estimate how good your problem data is. They tell you visually if your problem can be generalized and what kind of results you can expect.

We are 100% sure that you will like our aiNet application.

Registration (Thank you for registering the aiNet!)

What Is Your Benefit
How to Register
Disclaimer of Warranty
Registration Form

We did not cripple this program in any way. This means that this is a FULL version of the aiNet application. Nothing has been disabled and there are no extra files. There is only a slight difference in the manual. If you order a hardcopy of the manual, there will be some pictures included, which you can not see in a shareware version of manual. These pictures are in a bit map format and they consume an enormous amounth of memory -- a few megabytes and you know how long it takes to download a few megabytes from an Internet server. Only for this reason we put them out of the manual.

We think that releasing a full version is necessary and it enables you to fully evaluate the aiNet before you decide to register it. This also means that we are relying totally on your honesty to register.

What Is Your Benefit

If you become a registered user, you will be able to receive free upgrades to all future shareware releases of the aiNet application, till (but not including) the next major release. We will number major upgrades in whole numbers and minor in .10 or .01 increments. As we allready said, the minor upgrades will be free. For the new major releases we will give you a 50% discount if you are a registered user.

You will also have a chance to have an influence on the features of the future versions, the same is not guaranteed for unregistered users.

Registered users will have 100% support via e-mail or ordinary mail. This support is for correcting bugs in the software and manuals and does not include advice on how to solve various problems using neural network. We will also inform registered users about new issues of the aiNet or of our similar applications, but only if we are allowed to do so.

A 50% discount will be avaible to all registered users, when they will consult us about solving various problems using the aiNet application. We are also available for contract work on the use of neural networks.

How to Register

If you use the aiNet for more than 21 days - three weeks - you are obligated to register it by paying the registration fee. To do so, fill the registration form, sign it and mail the payment to:

AINET
Ales Krajnc, s.p.
Trubarjeva 42
SI-63000 Celje
Slovenia
Europe

If you are paying with a credit card, than you can also send the registration form via e-mail (ainet@ikpir.fagg.uni-lj.si). Just take a <u>registation form template</u> (ORDER.TXT), fill and drop it into your emailer. This will speed up our responce.

As you will see from the registration form we provide two different ways for the registration. The first way is faster and slightly cheaper; we will mail (or e-mail) you your registration text and code. After you receive the text and code, you will enter them into the registration dialog box. This will remove all registration-demand messages and will explicitly register the aiNet to you (registration text will appear in the caption bar).

If you choose the second way, we will send you the aiNet application on a diskette. You will also receive the registation text and code, which can be used for next minor upgrades. Optionaly, you may order also a hardcopy of the manual.

Currently, we accept two methods of payment: credit cards (preferable) and money checks. Generaly, there are no problems with credit card payment, but some problems may occur with money checks.For this reason, we prefer a credit card payment.

Disclaimer of Warranty

This software and documentation are sold "as is" and without warranties as to performance of merchantability or any other warranties whether expressed or implied. Because of the various hardware and software environments into which this program may be put, no warranty of fitness for a particular purpose is offered. You use the aiNet *entirely at your own risk*, and you supply it to your customers, friends, family or acquaintances *entirely at your own risk*.

In no event shall we (AINET s.p.) be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use the aiNet application, even if we have been advised of the possiblility of such damages.

If these terms are not acceptable to you, then please *DELETE* all the files from your disks *immediately and permanently*.

aiNet 1.0 REGISTRATION FORM (USE CAPITAL LETTERS, PLEASE)

Name:_____

E-MAIL Address:_____

Company/Address:_____

_____

Registration Name:_____(Do not make it too long!)

(This name will appear on the aiNets caption bar. The registration code, you will receive, will be calculated on the basis of the Registration Name)

How do you want to receive a registration:

[ ] I want to receive a registrered version of the aiNet on a diskette.
    Mail it to me in a [ ] 5.25" [ ] 3.5" diskette.

[ ] I want to receive a registration card only (includes Registration Text & Code), but no diskette. Send it to me via [ ]mail, [ ]e-mail.

I order _____ hardcopy(s) of aiNet manual at the price of US$10 for each copy. This makes US$_____ in total.

Price Calculation:

Registration Fee:_____Single user registration fee: US$200.
                         Multiple users registration fee:
                         2-5 users US$160 per user, 6-10 users US$130 per user,
                         11 or more users US$100 per user.

Manuals: _____Fill in the total from manual order (above), or write zero if you do not order any manuals.

Shipping & Handling:_____Zero, if you do not order any manuals or diskette. Otherwise: Europe US$4, other countries US$9.

Total Cost:_____

Credit Card Orders: Please Check

[ ] MasterCard, [ ] Visa, [ ] American Express, [ ] Eurocard, [ ] Diners Club

Cardholder's Name:_____

Address:_____

Credit Card Number:_____

Expiration Date: Month _____ Year _____

        Credit card payments may be sent via mail or E-Mail!

Check Payments: Make check payable in U.S. currency to:

AINET   **Internet:**
Ales Krajnc, s.p.   ainet@ikpir.fagg.uni-lj.si
Trubarjeva 42
SI-63000 Celje
Slovenia, Europe

[ ] I want, [ ] I do not want to be informed about new issues of the aiNet.

The Toolbar

The Toolbar is a row of buttons at the top of the main window which represent application commands. Clicking one of the buttons is a quick alternative to choosing a command from the menu.   Buttons on the toolbar activate and deactivate according to the state of the application.

<u>Button</u> Action	Menu Equivalent
Create a new neural net. File|<u>New</u>	
Locate and open a neural net File|<u>Open</u>	
Save the file in the active window File|<u>Save</u>	
Cut selected text to Clipboard Edit|<u>Cut</u>	
Copy selected text to Clipboard Edit|<u>Copy</u>	
Paste text from Clipboard Edit|<u>Paste</u>	
Undo previous editor action Edit|<u>Undo</u>	
Switch to Model Vectors View View|<u>Model Vectors</u>	
Switch to Prediction View View|<u>Prediction</u>	
Switch to Error Distribution View View|<u>Error Distribution</u>	
Switch to Charts View View|<u>Charts View</u>	
Calculate prediction, filtration or verification Prediction&Error Distr.|<u>Calculate</u>	
Display help file contents Help|<u>Contents</u>	

File Menu

The file menu provides commands for creating new files (in our case new artificial neural networks), opening existing files, closing files, saving files and exiting the aiNet application.

New   **Create a new, untitled file - neural network.**
Open   **Open an existing file.**
Close Neural Net **Close the current neural network - document.**
Close View   **Close the current view.**
Save   **Save the current neural network.**
Save As   **Save the current neural network under a new name.**
Save Prediction   **Save the data in the Prediction View window.**
Open Prediction **Open an existing prediction file.**
Exit   **Exit the aiNet application.**

New Command

Using this command you order the aiNet to open a brand new neural network. Before the aiNet lets you enter some data into a new neural network, a dialog box pops up. The dialog asks you about the number of model vectors and the number of parameters in the model vector. (You can always add or remove some model vectors or parameters later.)

After you have filled the dialog, the aiNet will ask you about the file format you want to use. Select the one you like and Model Vectors View window should appear on the screen.

Open Command

You use this command when you want to load some existing neural network from the disk. The aiNet will display the File Open dialog. There you select the file you want to load. If the file is loaded successfully, the aiNet will automatically open Model View and display model vectors from the file.

Close Neural Net Command

Close Neural Net command will close all opened views of the current neural network and delete the network from memory. Before the aiNet does that, it will remind you to (give you the last chance) to save the network.

Close View Command

The aiNet will close the active view of the current neural network. If this view is the last opened view, you will be asked to save the neural network to the disk.

Save

Save command will save the current neural network to the disk. It usually takes a moment or two for the aiNet to do this job. However, if the neural network does not have a file name yet and is labeled as Untitled1 or Untitled2 ... than the <u>Save As</u> command will be executed.

Save As

This command saves the current neural network to the disk, but before this is done, a dialog is displayed. In the dialog you must enter a file name and a location of the file.

You can also specify a file format you want to use. You can use either binary format (AIN extension) or ASCII comma delimited format (CSV extension). If you use the CSV format, than the neural network must be denormalized first.

Save As command also changes the text in the title bar of the views -- a new file name will appear within the square brackets.

Save Prediction

This command is enabled only if Prediction View is active. Save Prediction allows you to save the prediction vectors separately from the model vectors. This is very useful if you want to have several different independent sets of prediction vectors. Predictions are always saved in the ASCII comma delimited format, but with a PRD extension.

Open Prediction

This command does the opposite from the Save Prediction command. It is enabled only if Prediction View is active. It will ask you to select prediction file (PRD extension) you want to load. After you select a file, the aiNet will discard previous contents of Prediction View and will load it with the prediction vectors from the file.

Exit

Use Exit when you want to quit the aiNet application. Before the aiNet closes, it will ask you to confirm your decision. If you really want to quit the aiNet, you will be further asked to save the neural networks.

View Menu

This menu allows you to bring up the view you want to see. If the view has not been created yet, than the aiNet creates a window for the selected view and puts it on the top. If the view window already exists and is covered with other windows, than this command puts selected view window to the top. This rule holds for all four views.

Model Vectors View
Prediction View
Error Distribution View
Charts View

Sometimes some of the views are disabled. This usually happens when the model vectors in the Model Vectors View window are not normalized. Selecting the Model Vectors|Normalize + Lock command will enable all views.

Model Vectors View

Model Vectors View is a basic view, where all model vectors are presented in a spreadsheet form. The rows represent model vectors and the columns represent parameters. Usually you can not see all the model vectors and parameters at once, so you must do some scrolling. This view is automatically activated when you create a new neural network or you load an old neural network from the disk. You use this view to edit model vectors, to add some new or remove some old model vectors, etc. In this view you also normalize and denormalize model vectors.

Prediction View

Prediction View is used when you want to run some predictions. This view is very similar to Model Vectors View and allows you to take similar actions. You can edit the input part of the predictions vectors and than use the aiNet to calculate the output part. Prediction view is therefore the most useful view.

Error Distribution View

When the aiNet calculates the prediction it usually makes some mistakes -- the predictions are not 100% correct. Thanks to the special algorithms, the aiNet can besides prediction itself predict also the ratio of the error in the prediction result. Before the aiNet can do that, so called filtration and verification process must run over all model vectors. Error Distribution View shows the results of the filtration and verification. Like both views above, this view also uses spreadsheet-like window to present the results. This view is Read only view, which means that you can browse through the results only, but you can not change them.

Charts View

Charts View is an extension of Error Distribution View, because this view represents the same results as Error Distribution View, but in the graphical ways. This enables you to estimate these results visually. Two different charts are currently avaible. The first one is some kind of correlation chart. On the horizontal axis are output values from model vectors (correct values) and on the vertical axis are values calculated by the aiNet. If the calculated and the correct value for some model vector are the same (or almost the same), than the point of this model vector lies exactly on the diagonal line. In other words this means, closer the dots are to the diagonal, less error is present in the verification (red dots) or filtration (green dots). The second chart is a bar chart. It shows the verification or filtration error of each model vector. Model vectors are enumerated on the horizontal axis and the error is shown on the vertical axis.

Options Menu

Comment Width

When you want to change the width of the comment column in Model Vectors View, than you can do that by selecting this command.

Each model vector can have comments. Although you usually do not need them, you will encouner occasions, when they will be pure gold. Imagine you are collecting model vectors from some experiment, which is time dependent. You put these model vectors into the neural network. Now you make analysis and you notice that there is something wrong with some of the model vectors. If these vectors do not have a comment, it is hard to tell when were they measured. But if they do have a comment, a date and time, then it is easy.

Fonts

In the aiNet version 1.0 this command is disabled. In later versions it will allow you to use different fonts in the views.

Register

If you decided to register the aiNet application, we will send you your unique registration code. You enter your name and code in a dialog box, which is invoked by Register command. After successful registration, the aiNet stops displaying registration messages.

Window Menu

The Window menu provides commands to control the position and layout of the aiNet's windows.

Cascade	**Resize and position all windows in an overlapping pattern.**
Tile	**Resize and position all windows in an non overlapping pattern.**
Arrange Icons	**Align all iconized windows along a grid.**
Close All	**Close all windows.**

Help Menu

The Help menu provides access to the help system and the about dialog.

<u>Contents</u>   Help topic contents.

Window Arrange Icons Command

The **Window|Arrange Icons** command arranges all iconized windows into rows along the bottom of the application's main window.

Window Cascade Command

The **Window|Cascade** command arranges all document windows from the top-left position of the application's main window so that the title bar of each is visible.

Window Close All Command

The **Window|Close All** command closes all document windows open in the application.

Window Tile Command

The **Window|Tile** command arranges all document windows side-by-side in a non-overlapping pattern.

Window Help table of contents

The **Help|Contents** displays the help contents page.

Edit Menu (Model Vectors View)

The Edit menu provides several commands to edit model vectors.

Add rows (model vectors)
Add columns (parameters)
Insert row/column
Delete row/column
Delete empty rows (model vectors)
Undo, Cut, Copy, Paste

Add rows (model vectors)

Select this command to add some extra model vectors to Model Vector View. The aiNet will open a simple dialog. The dialog will ask you about the number of rows (model vectors) you want to add. Type the number and close the dialog with the OK button. The selected number of new rows will appear at the end of Model Vector View.

Add columns (parameters)

If you realized that your model needs more parameters, use this command to add some. A dialog will pop up asking you for the number of new parameters. All new parameters will have the exclude status turned on. (Exclude means neither input or output.)

Insert row/column

This command will be available only if you have previously selected a model vector (row) or a parameter (column). If a row has been selected, than a new row (model vector) will be inserted before the selected row. The same is valid for the columns; a new column (parameter) will be inserted before the selected column. The new parameter will have the exclude status turned on.

Delete row/column

This command will be available only if you have previously selected a model vector (row) or a parameter (column). Before the selection is deleted, the aiNet asks you to confirm the command.

Delete empty rows (model vectors)

This command deletes all model vectors with no entries. It sometimes happens that you have initially created too many rows or you may have added too many rows later. Nevertheless, it would be a good idea to delete such empty rows. This will not only free some memory, but it will speed up the computation as well.

Undo, Cut, Copy, Paste

Undo works within Edit window only, which is located in the speed button bar. The same is valid for the Cut, Copy and Paste commands.

Future releases of the aiNet application will probably further implement these commands.

Edit Menu (Prediction View)

The Edit menu provides several commands to edit predict vectors.

Show local error
Hide local error
Add rows (predictions)
Insert row
Delete row
Delete empty rows (predictions)
Undo, Cut, Copy, Paste

Show local error

The aiNet can besides prediction calculate also an estimation for the error (or the noise rate) for every single prediction result. Show local error command displays these estimations in the Prediction View window.

Please note that if you really want to see these estimations you must calculate error distribution first. If error distribution is not calculated, than only empty lines are displayed.

Hide local error

Does the opposite of the Show local error command. It hides error estimations in the Prediction View window.

Add rows (predictions)

This command is the same as the Add rows (model vectors) command, except that it acts in the Prediction View window.

Insert row

A row must be selected, before this command can be used. It will insert a new prediction vector before selected row.

Delete row

A row must be selected, before this command can be used. It will delete the selected row. You will have to confirm this command.

Delete empty rows (predictions)

This command is the same as Delete empty rows (model vectors) command, except that it acts in the Prediction View window.

Edit Menu (Error Distribution View)

The Edit menu in the Error Distribution View window provides just two commands:

Show difference
Show predicted value

Error Distribution View is used to show the results of filtration (FE:) and verification (VE:). These results can be presented in two ways:

- As the difference between an initial (correct) and a calculated result **(Show difference** command),

- as a predicted value - calculated result (**Show predicted value** command).

Model Vectors Menu (Model Vectors View)

This menu is avaible only when Model Vectors View is the active window. Here are the commands it provides:

Normalize + Lock
Denormalize + Unlock
Normalization settings

Normalize + Lock

Before you can do any computation with the aiNet, you must normalize the model vectors. After the normalization is done, the aiNet will enable a lot of commands, which you will need to perform calculations. This command also locks the model vectors - you can not edit them.

Denormalize + Unlock

This command reverses normalized model vectors back to their original values. It also unlocks them and thus allows you to edit them. But do not forget that when model vectors are unlocked, you can not perform any calculations.

Normalization settings

Before the aiNet can perform any kind of calculation, all model vectors must be normalized. This means that some real physical dimensions will be transformed to something else.

Two normalization methods are available:
- <u>regular,</u>
- <u>statistical.</u>

How to choose the right kind of normalization? Here is the rule we use (it is not an ultimate rule):

* Use regular normalization if the data in model vectors is already normalized in some way -- if you know for sure that all values will fall within some physical limits.

* When you can not say for sure that all the values will be between some limits, use statistical normalization.

In most cases the normalization type does not reflect significaly on the results. Please note, that **normalization type does have an influence on the optimal value of penalty coefficient**. Statistical normalization usually requires larger values for the penalty coefficient.

Regular Normalization

If you choose a regular type of normalization, than the aiNet will perform the following actions during normalization:

1. Scan trough all model vectors and find the largest (maximum) and the smallest value (minimum) for each parameter.

2. For each parameter calculate normalization factors in such a way, that all values in model vectors will be between -1 and 1 after normalization. (The maximum will be 1 and the minimum will be -1.)

3. Normalize all model vectors. Now all values in model vectors will be between -1 and 1.

Statistical Normalization

Here are the steps that the aiNet will perform during statistical normalization:

1. Run through all model vectors and calculate an average and a standard deviation for each parameter respectively.

2. For each parameter calculate normalization factors in such a way that values, which are exactly one standard deviation apart from the mean value, will become 1 (or -1) after normalization.

3. Normalize all model vectors. All values that are now less than |1| are those within one standard deviation, all others (greater than |1|) are more than one standard deviation apart from the mean value.

Prediction Menu (Prediction View)

This menu is avaible only when Prediction View is the active window. Here are the commands it provides:

Calculate prediction
Penalty settings
Normalization settings
Local error settings

Calculate prediction

When you have entered all input values in the prediction vectors, you can calculate the output values selecting the Calculate prediction command. In a moment or two results will appear in the output parameters of the Prediction View window. If the error distribution was calculated before, than the estimation of the errors will also be calculated. The error distribution will be shown, if Edit|Show local error is selected.

Penalty settings

This is the most important command and we are sure you will use it a lot. When you select this command a dialog appears on the screen. The dialog asks you for the value of penalty coefficient and also lets you select a type of the coefficient.

The penalty coefficient value

Choosing the right value of the penalty coefficient usually solves the problem.

We can give you only informative instructions for selecting the right value. Our experiences can be summarized in the following simple rules:

* **The more model vectors you have, the smaller the optimal value is.**
* **The more parameters you have, the larger the optimal value is.**
* **The more noise your data has, the larger the optimal value should be.**
* **The best thing you can do is try a few values. Do not forget to try some stupid values also.**
* **Small penalty value leads to overfitting (overtraining) and**
* **Large penalty value leads to underfitting (overgeneralizing).**

The penalty coefficient type

The dialog which pops up gives two possible choices for the penalty coefficient type:

* **static**
* **dynamic**

Please note that switching from the static to the dynamic type (or opposite) reflects also on an optimal penalty coefficient value. The dynamic type usually leads to smaller values.

Static penalty coefficient type

The static coefficient type means that penalty coefficient value will be the same all over the hyper-dimensional space. Use this type when your model vectors are approximately uniformly (regularly) distributed. If your data is noisy, than this selection is also recommended.

Dynamic penalty coefficient type

This coefficient type behaves dynamically. When prediction, filtration and verification are calculated, than actual value of penalty coefficient depends on the density of model vectors. If there are a lot of model vectors in the surroundings of prediction vector, than penalty coefficient decreases -- fits little more. However, if model vectors are rare in the surroundings of prediction vector, than penalty coefficient increases -- generalizes little more. Use the dynamic coefficient type, when model vectors are grouped in clusters and are not uniformly distributed over hyper-dimensional space.

Local error settings

Using the Local Error Settings command you control the calculation of the error distribution. You can turn the error distribution on or off by selecting between these two possibilities:

* Enable (Yes, calculate local error),
* Disable (Do not calculate local error).

You can also select which type of local error you want to use. There are two types:

* Unsigned (absolute) local error,
* Signed local error.

Use an unsigned (absolute) type if you have noise in your model vectors. When you know that your problem is smooth and without any noise, than you may try with signed local error type.

The selection of local error type will have the greatest effect on prediction process, more precisely on estimation of local error in the prediction process.

So far we have been using two terms, which are related and have the same background:

an error distribution and
a local error (or local error estimation).

Error Distribution

Error distribution is used together with the terms filtration and verification. Actually, the error distribution is the result of filtration and verification. We named it distribution, because filtration and verification calculate an error for each model vector. When we know the error for each model vector, we also know how error is distributed over all model vectors, hence the term distribution.

Local Error (or Local Error Estimation)

Local error estimation is associated with prediction. If we know the error distribution and we run the prediction process, then we can predict not only the output parameters, but also the error for every single prediction. Because such an error prediction can behave locally (it is not a constant in the hyperspace), we use the term local error. Since error distribution is always noisy, this error prediction can not be very accurate, hence the term estimation.

Error Distribution Menu

This menu is avaible only when Error Distribution View or Charts View is the active window. Here are the commands this menu provides:

Calculate error distribution
Global error report
Penalty settings
Normalization settings
Local error settings

Calculate error distribution

With this command you engage the calculation of error distribution. In a couple of moments results will appear in the output parameters of the Error Distribution View window. If there are many model vectors, the calculation of the error distribution can take a considerable amount of time.

(NOTE: When you double the number of model vectors, than the calculation will take four times longer.)

Global error report

If you want to see global error estimates (root mean square error = RMS) than select this command. A dialog will be popped up, where the total RMS error will be shown, as well as the RMS errors for single output parameters.

New Neural Network Dialog

The dialog asks you to specify a neural network size. The size is defined by the number of model vectors (learn samples) and by the number of parameters (units) in a model vector.

Example

Assume, we have a multiplication problem:

x * y = z

A model vector would in this case have two input parameters (**x** and **y**) and one output (**z**). This means tree parameters in total.
We want to train the network on 20 model vectors. An example is below:

```
x       y       z
1       1       1
1       2       2
2       3       6
...
4       5       20
```

The neural network size for this example would be:
20 model vectors,
3 parameters

Add Some Rows or Columns Dialog

Enter a number of new columns (parameters) or new rows (model vectors) which will be added (appended) to the base.

All new cells will be declared as missing values -- the cells will be empty.
The new parameters will have EXCLUDE status turned on.

Input value is out of range

This is only a warning. The aiNet keeps track of the minimum and maximum for each parameter. If you have entered a value, which is not in the minimum-maximum interval, the aiNet asks you to confirm your entry.

This *safety* is somehow annoying in the first few entries, but later prevents you from entering accidental values.

Parameter Description Dialog

This dialog enables you to change some attributes of selected parameter.

Name:
Enter a new name for the parameter. Do not make it to long. The new name will appear on the screen when the dialog is closed with the OK button.

Status:
Three different status types are available: input, output and exclude.

Input   Select input status, when you want that this parameter will be an input parameter.
Output Select output, when you want that this parameter will be an output parameter.
Exclude Select exclude status, when you want to temporary exclude this parameter from the model vectors and prediction vectors. The aiNet then behaves as this parameter do not exist.

Discrete:
Check discrete option to specify that this parameter will have discrete values only. Discrete values can be any real values, however integers are preferred. In the other words, discrete type enables clustering data.
An Example ...:

Range:
Shows current minimal and maximal value of the parameter.

Numer of decimals:
Enables you to specify a number of decimals you want to see on the screen.

Alignment:
Allows you to select an alignment type.

Cell width:
Here you enter the number of characters (approximately) you want to be displayed on the screen

Global RMS (root mean square) Error Dialog

The Global RMS dialog box displays all global errors which the aiNet calculates.

You see:
*   <u>total RMS errors</u> and
*   <u>RMS errors on parameters</u>

To see an error of a specific output parameter, select desired parameter in the output parameter list box. As you change the selection in the list box, numbers on the right change accordingly.

Registration Dialog

In this dialog you enter the registration text and code. By doing that you explicitly register the aiNet application to you or your company.

How you get the text & code. Fill in the registration order and send it to us. On the registration order you specify the exact text, you want to appear in the aiNets caption bar. When we get your order, we will calculate the registration code on the basis of the text you send to us. Then both (your text and registration code) will be send back to you.

If you have already received the registration text and code, than enter the text and code exactly as it was send back to you. If you have entered everything correctly, than close the dialog with the OK button. The aiNet will thank you for your registration and all nagging messages will disappear.

More about registration ...

Total RMS error

The total error is used as a measure of an error in the model of the whole phenomena (problem) including all output parameters. See Part 2, Chapter 3 in the Users Manual for more detail.

RMS Error on Parameters

The parameter RMS error is used as a measure of an error in the selected output parameter. See Part 2, Chapter 3 in the Users Manual for more detail.

Discrete Option Example

Imagine, that we have an input parameter, which describes printer types: dot-matrix, laser, ink-jet, daisy-wheel, ... We want that different printer types will be handled separately, which means that data associated with selected printer type, will not interfere with the data of other printer types.

Now we want to do some predictions. In the input part of the prediction vector is also the discrete printer type parameter and we specify the ink-jet printer for this parameter. As prediction runs, all model vectors which have discrete parameter not equal to ink-jet are ignored. Thus, the result is calculated only on the basis of the ink-jet printer data.

IMPORTANT NOTE: DISCRETE OPTION ONLY WORKS WITH THE INPUT STATUS TURNED ON. IF YOU TURN ON THE OUTPUT STATUS, THAN DISCRETE OPTION WILL BE IGNORED.

Possible future improvements:
Now we are testing discrete option, which will enable the user to select the rate of the interference of the discrete parameters. (Currently, this rate is set to zero - no interference at all.)