

User's Guide For 4Cast XL

Example 1: Times-series Prediction

Step 1: Preparing Data

The input data to neural network is not always hard physical measurements. Scaling the data sometimes is needed so that the neural network can learn better. **4Cast XL** use a sigmoidal activation function as its output. Therefore, user need to scale the input data first so that the target activations to values can be comfortably learned. Please take note that **4Cast XL** only process numeric data.

All the data need to be converted to the between 0 to 1. Neural network will learn better if there is uniformity in the data.

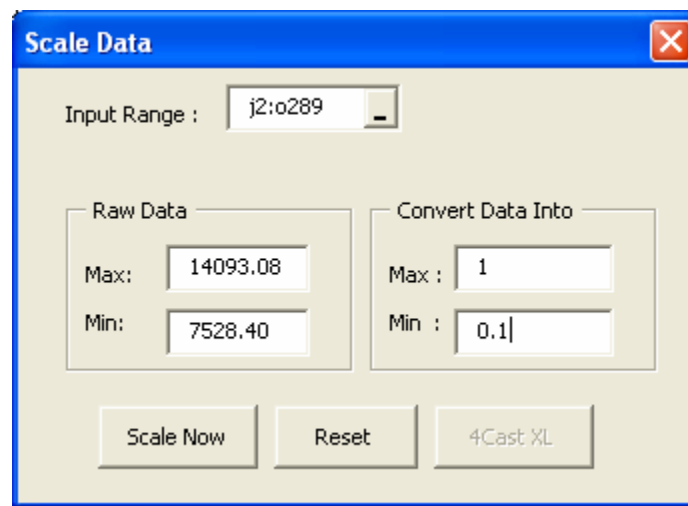
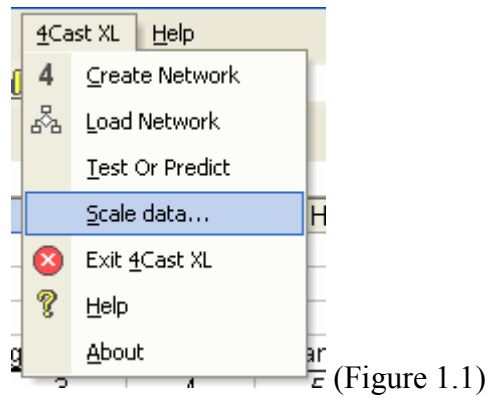
Here, a real life example is use. This data is taken from the DJIA weekly prices, from the period from 22 April 2002 to 22 Oct 2007.

Let's organize the data first. The basic assumption for time series forecasting is that the pattern found from historical data will also hold in the future. In practice, we partition the data into two parts.

The first part is used to train the neural network, while the second part of data is used to test the performance of the model. A model is considered good if the error of out-of-sample testing is the lowest compared with the other models. If the trained model is the best one for validation and also the best one for testing, one can assume that it is a good model for future forecasting.

There are tradeoffs for testing and training. One should not say it is the best model unless he has tested it, but once he has tested it, he has not trained enough. The general partition rule for training and testing set is 70% and 30%, respectively.

a) Scaling of variables and data is automated. Just select **Scale Data** on the **4Cast XL** menu (see Figure 1.1) and the input form below (Figure 1.2) will be shown:



(Figure 1.2)

For column J, the input data consist of the 5 days moving averages of the DJIA weekly prices, column K consist of 10 days moving averages and so on. The highest price during the period from 22 April 2002 to 22 Oct 2007 is 14,093.08 and the lowest price is 7528.4.

288 weekly data is entered accordingly from column J2 to N289. Column O is the output/target data consist of actual weekly closing.

I	J	K	L	M	N	O
Date	MA_5	MA_10	MA_20	MA_60	MA_120	Target
4/22/2002	10206.85	10297.863	10118.79	10127.81	10409.4	10006.63
4/29/2002	10127.38	10301.711	10128.57	10120.15	10396.77	9939.92
5/6/2002	10061.04	10258.817	10123.8	10108.41	10381.91	10353.08
5/13/2002	10093.49	10236.876	10134.6	10117.24	10374.42	10104.26
5/20/2002						9925.25
5/28/2002						9589.67
6/3/2002						9474.21
6/10/2002						9253.79
6/17/2002						9243.26
6/24/2002						9379.5
7/1/2002						8684.53
7/8/2002						8019.26
7/15/2002						8264.39
7/22/2002						8313.13
7/29/2002						8745.45
8/5/2002						8778.06
8/12/2002						8872.96
8/19/2002						8663.5
8/26/2002						8427.2
9/3/2002	8697.434	8614.798	9197.439	9700.845	10161.26	8312.69
9/9/2002	8610.882	8508.117	9117.537	9663.739	10141.97	7986.02
9/16/2002	8452.474	8438.266	9016.507	9620.562	10122.69	7701.45
9/23/2002	8218.172	8406.485	8904.583	9575.308	10096.92	7528.4
9/30/2002	7991.152	8332.886	8763.349	9525.568	10071.2	7850.29
10/7/2002	7875.77	8286.602	8650.651	9482.802	10049.54	8322.4
10/14/2002	7877.712	8244.297	8570.508	9450.829	10032.19	8443.99
10/21/2002	7969.306	8210.89	8513.224	9417.843	10015.49	8517.64
10/28/2002	8132.544	8175.358	8465.396	9393.975	9997.84	8537.13
11/4/2002	8334.29	8162.721	8429.563	9376.163	9978.876	8579.09
11/11/2002	8400.05	8177.04	8300.254	9304.004	9900.000	8004.04

Scale Data ✖

Input Range :

Raw Data

Max:

Min:

Convert Data Into

Max :

Min :

(Figure 1.3A)

Thus we need to convert the data in the range J2:O289 (see Figure 1.3A). That is 6 columns times 288 rows in the worksheet “Raw Data”. Enter J2:O289 into the **Input Range** edit box. Press the Tab key on your keyboard to exit. When you press Tab, **4Cast XL** will automatically load the maximum (14,093.08) and the minimum (7528.4) in the **Raw Data** frame **Min** and **Max** textbox.

By default in the **Convert Data Into** frame, **4Cast XL** use the value 1 for maximum and 0.1 for minimum (see Fig 1.2) above. Of course you can change this. It is advisable not to use the value 0 as the minimum as it represent nothing.

Click on the **Scale Now** button and all the data in the range J2:O289 will be converted to 0.01 to 1. I’ve format the converted values into 2 decimal points. (see Figure 1.3 below)

	I	J	K	L	M	N	O
1	Date	MA_5	MA_10	MA_20	MA_60	MA_120	Target
2	4/22/2002	0.47	0.48	0.46	0.46	0.49	0.43975868
3	4/29/2002	0.46	0.48	0.46	0.46	0.49	0.43061292
4	5/6/2002	0.45	0.47	0.46	0.45	0.49	0.48725604
5	5/13/2002	0.45	0.47	0.46	0.45	0.49	0.45314349
6	5/20/2002	0.45	0.46	0.46	0.46	0.49	0.4286017
7	5/28/2002	0.45	0.46	0.46	0.46	0.49	0.38259458
8	6/3/2002	0.44	0.45	0.45	0.46	0.49	0.36676533
9	6/10/2002	0.42	0.44	0.45	0.45	0.49	0.33654634
10	6/17/2002	0.39	0.42	0.45	0.45	0.49	0.3351027
11	6/24/2002	0.37	0.41	0.44	0.45	0.48	0.35378084
12	7/1/2002	0.35	0.40	0.44	0.44	0.48	0.25850232
13	7/8/2002	0.33	0.38	0.43	0.44	0.48	0.16729559
14	7/15/2002	0.29	0.36	0.42	0.43	0.48	0.20090225
15	7/22/2002	0.26	0.33	0.40	0.43	0.48	0.20758438
16	7/29/2002	0.24	0.30	0.38	0.42	0.47	0.26685429
17	8/5/2002	0.22	0.29	0.37	0.41	0.47	0.27132503
18	8/12/2002	0.22	0.28	0.36	0.41	0.47	0.28433557
19	8/19/2002	0.25	0.27	0.35	0.41	0.47	0.25561916
20	8/26/2002	0.26	0.26	0.34	0.40	0.46	0.22322307
21	9/3/2002	0.26	0.25	0.33	0.40	0.46	0.20752405
22	9/9/2002	0.25	0.23	0.32	0.39	0.46	0.16273847
23	9/16/2002	0.23	0.22	0.30	0.39	0.46	0.12372469
24	9/23/2002	0.19	0.22	0.29	0.38	0.45	0.1
25	9/30/2002	0.16	0.21	0.27	0.37	0.45	0.14413025
26	10/7/2002	0.15	0.20	0.25	0.37	0.45	0.20885527
27	10/14/2002	0.15	0.20	0.24	0.36	0.44	0.22552493
28	10/21/2002	0.16	0.19	0.24	0.36	0.44	0.23562215
29	10/28/2002	0.18	0.19	0.23	0.36	0.44	0.23829417
30	11/4/2002	0.21	0.19	0.22	0.35	0.44	0.24404678

(Figure 1.3)

4Cast XL will also automatically store the minimum (in cell J292) and the maximum (cell J291) value of the raw data in the last row and first column of the raw data. (see Figure 1.4 below).

Delete the value in cell O289 because this is the value or weekly price that we want to predict.

281	8/27/2007	0.88	0.91	0.91	0.78	0.66
282	9/4/2007	0.88	0.90	0.91	0.78	0.67
283	9/10/2007	0.89	0.90	0.91	0.79	0.67
284	9/17/2007	0.91	0.90	0.91	0.80	0.67
285	9/24/2007	0.92	0.90	0.92	0.80	0.68
286	10/1/2007	0.94	0.91	0.92	0.81	0.68
287	10/8/2007	0.97	0.93	0.92	0.82	0.69
288	10/15/2007	0.97	0.93	0.92	0.82	0.69
289	10/22/2007	0.97	0.94	0.92	0.83	0.69
290						
291		14093.08				
292		7528.40				
293						
294						

(Figure 1.4)

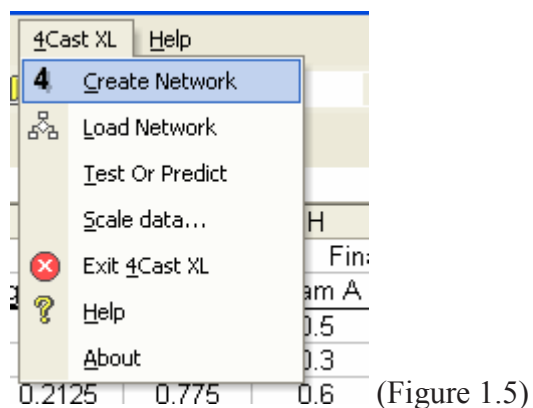
b) After converting the raw data, click on the **X** at the top right of dialog form to exit. Now we need to partition the data into 2 parts.

- i) Part 1, row 2 to row 200 as the training data
- ii) Part 2, row 201 to row 288 as the testing data.

User can start to build the neural network model after this.

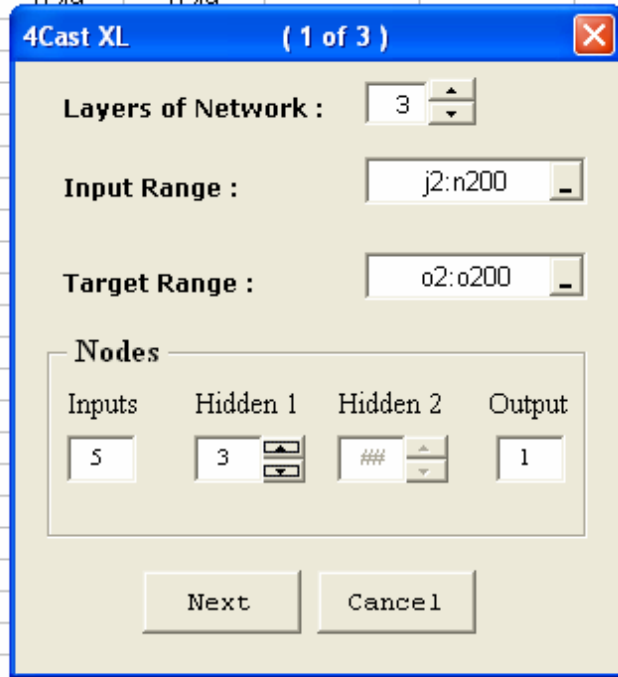
Step 2: Building the architecture of the neural network model.

a) Select **Create Network** on the **4Cast XL** menu (see Figure 1.5) and the dialog box **4Cast XL (1 of 3)** will be displayed (see Figure 1.6).



(Figure 1.5)

J	K	L	M	N	O	P	Q
MA_5	MA_10	MA_20	MA_60	MA_120	Target		
0.47	0.48	0.46	0.46	0.49	0.44		
0.46	0.48	0.46	0.46	0.49	0.43		
0.45	0.47	0.46	0.45	0.49	0.49		
0.45	0.47	0.46	0.45				
0.45	0.46	0.46	0.46				
0.45	0.46	0.46	0.46				
0.44	0.45	0.45	0.46				
0.42	0.44	0.45	0.45				
0.39	0.42	0.45	0.45				
0.37	0.41	0.44	0.45				
0.35	0.40	0.44	0.44				
0.33	0.38	0.43	0.44				
0.29	0.36	0.42	0.43				
0.26	0.33	0.40	0.43				
0.24	0.30	0.38	0.42				
0.22	0.29	0.37	0.41				
0.22	0.28	0.36	0.41				
0.25	0.27	0.35	0.41				
0.26	0.26	0.34	0.40				
0.26	0.25	0.33	0.40				
0.25	0.23	0.32	0.39				
0.23	0.22	0.30	0.39				
0.19	0.22	0.29	0.38				
0.16	0.21	0.27	0.37				
0.15	0.20	0.25	0.37	0.45	0.21		
0.15	0.20	0.24	0.36	0.44	0.23		
0.16	0.21	0.24	0.36	0.44	0.23		

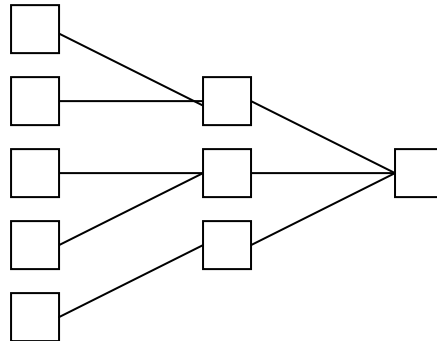


(Figure 1.6)

- Select 3 on the **Layers of Network**. Proven to be very effective on time-series forecasting.
-
- On the **Input Range** edit box, enter the range J2:N200 (five columns of inputs and 199 rows of input pattern, i.e. the training data). Each time you exit, please use the *Tab* key on your keyboard.
-
- On the **Target Range** edit box, enter the range O2:O200 (1 column of output/target and 199 rows of pattern to be learned). *Tab!!!* When you tab to exit, **4Cast XL** will automatically load the number of input column in the **Inputs** textbox on the **Nodes** frame. (Here we've 5 inputs)
-
- On the **Hidden 1** textbox of the **Nodes frame**, enter 3. This mean that we have select 3 nodes for the hidden layer. The general rule to follow here is that we use the pyramidal scheme, *5-3-1*. (you can also try 5-4-1, 5-2-1 etc)
- The **Output** textbox on the **Nodes frame** is already loaded with the value 1 as this is done automatically by **4Cast XL** as well.
- Click the **Next** button.

Graphically, it looks like this: (see below)

Input Layer Hidden Layer Output Layer



Click on the **Next** button and the dialog box **4Cast XL (2 of 3)** below will be shown(see Fig 1.7). The worksheet **Network Output** will be created. This worksheet is very important as it is use to store the the model architecture(display on the **Network Architecture** frame), the weights(w1 to w18) and all other important parameters. (see Figure 1.7)

B	C	D	E	F	G	H	I	J	K	L	M
	MSE				Weights		Output 1		Error 1		Hidd1_1
	AE			w 1							
	Test MSE			w 2							
	No. Of Layer	3		w 3							
	No. Of HL1	3		w 4							
	No. Of HL2	0		w 5							
	Input Nodes	5		w 6							
	Output Nodes	1		w 7							
	Mean			w 8							
	Std Dev			w 9							
				w 10							
				w 11							
				w 12							
				w 13							
				w 14							
				w 15							
				w 16							
				w 17							
				w 18							

Do not edit Column A !!!

4Cast XL (2 of 3)

Network Architecture

This is a 3 layers network with 5 input nodes, 3 HL 1, 0 HL 2 and 1 output(s).
199 patterns is feed to the network.

Training Methods

Backpropagation with Momentum
 Conjugate Gradient (Recommended)
 Genetic Optimization

Worksheet name : table(5)

Show MSE Graph

Next Cancel

(Figure 1.7)

Currently, the worksheet is blank as we have not start training yet.

b) Select the training option that you want in the **Training Methods** frame. Training option includes **Backpropagation with momentum**, **Conjugate Gradient** and **Genetic Optimization**. Select the checkbox **Show MSE Graph** to show the training error in each epoch. (*the Genetic Optimization training option will be discussed in Example 2*)

Choose the **Conjugate Gradient** training method. Click the **Next** button again and the **Conjugate Gradient** training method dialog form (see Fig 1.8 below) will be display.

Step 3: Training the network model

F	G	H	I	J	K	L
	Weights		Output 1		Error 1	
w 1	0.252525					
w 2	0.545577					
w 3	-0.44574					
w 4	0.242201					
w 5	-0.39576					
w 6	-0.58331					
w 7	0.327761					
w 8	-0.04065					
w 9	0.258624					
w 10	0.40747					
w 11	0.350927					
w 12	0.145317					
w 13	-0.07325					
w 14	0.572845					
w 15	-0.00294					
w 16	0.276322					
w 17	0.030649					
w 18	-0.0073					

CG (3 of 3)

Network Architecture

This is a 3 layers network with 5 input nodes,
3 HL 1, 0 HL 2 and 1 output(s).
199 patterns is feed to the network.

Epoch:

Precision:

Stop After

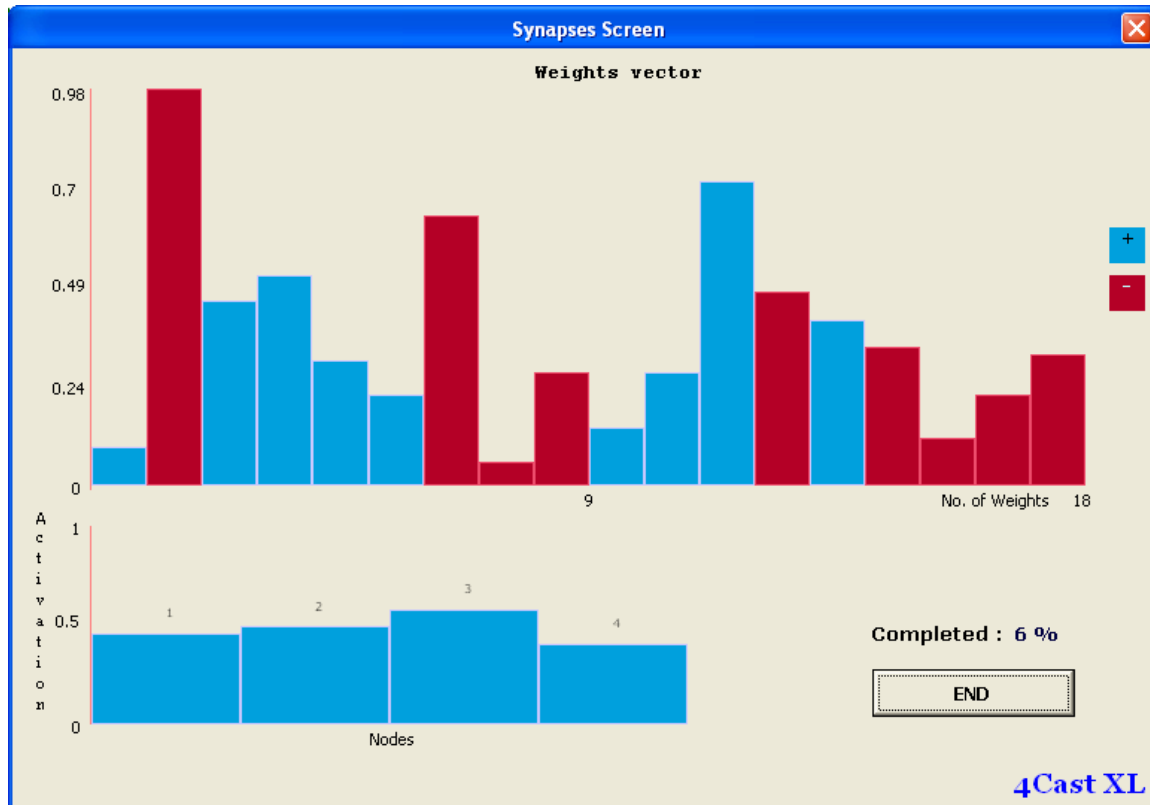
(Figure 1.8)

Select the **Initialize new weights**. 18 rows of value for the weights will be generated in G2:G19. (Note: This is optional as **4Cast XL** will generate the weights for you automatically if you didn't click on this button)

You also have the option to run the program for how many loops / **Epoch** (maximum for this program is 15000) or you can select the option to run the program for how many minutes. (maximum 300 minutes). User can stop training prematurely by clicking the

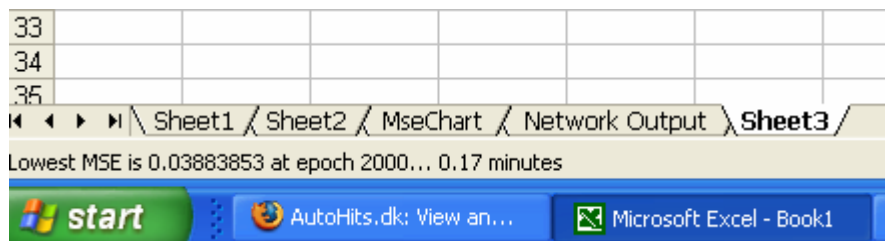
End button.(Note: MSE graph will not be updated if user do this. This might take a few seconds.)

Click the **Train** button to start training the model. The **Synapses Screen** will be shown.(see Fig 1.9 below) and the program will start running.



(Figure 1.9)

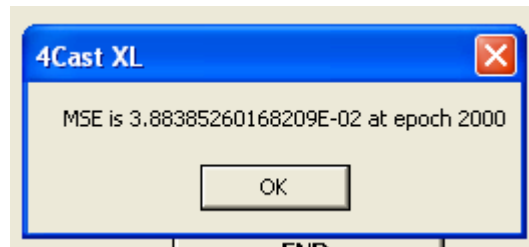
You can see the activities of the neural network model training. For experience user, they will know whether a suitable model has been built, whether enough hidden layers is created, whether the network has good starting weights etc, just by looking at this screen. **4Cast XL** is the only neural network program in the world that has this feature.



(Figure 1.10)

The number of **Epoch** and the **Mean Square Error** (MSE) will be shown in the *Status Bar* (see Figure 1.10 above)

A message box will pop up after the model has finish training for the current session. (see Fig 1.11)



(Figure 1.11)

User can stop training prematurely by clicking the **End** button.(Note: MSE graph will not be updated if user do this. This might take a few seconds.)

Here it show 0.003883 which is not satisfactory. Usually we need the MSE value to be below 0.001 or after training for 30000 epochs. As the result is not satisfactory we need to train the network model again. Before that, take a look at the **Network Output** worksheet. You'll see all the parameters generated on this sheet. Don't change anything here.

A complete analysis is presented in the **Network Output** worksheet. User can view and analyse the network model.(e.g. Each pattern output and error is presented thus, user can determine whether the network has learn well for a particular pattern.) In the hidden layer output, user will know which node is redundant or active. Weights of the network are also reported. User can also use the data here for graphing purposes, for example the *Actual Vs Forecasted chart*, *Sensitivity of Inputs chart*, *error distribution chart*, *correlation graph* etc.

For example, if you want to built the Actual VS Forecasted chart, goto the Network Output worksheet and copy the output (Output 1 in column I) and paste it beside the Target column in the data sheet. (see Figure 1.11B below)

O	I	J	K
Target	Output 1		Error 1
0.44	0.453812		0.000198
0.43	0.436887		3.94E-05
0.49	0.423779		0.004029
0.45	0.431013		0.00049
0.43	0.425122		1.21E-05
0.38	0.426693		0.001945
0.37	0.409374		0.001816
0.34	0.390229		0.002882
0.34	0.344814		9.43E-05
0.35	0.311844		0.001759
0.26	0.292765		0.001174
0.17	0.265261		0.009597
0.20	0.228966		0.000788
0.21	0.210953		1.13E-05
0.27	0.197308		0.004837
0.27	0.189527		0.006691
0.28	0.194472		0.008075
0.26	0.213718		0.001756
0.22	0.225673		6E-06
0.21	0.232364		0.000617
0.16	0.226195		0.004027
0.12	0.213583		0.008075
0.10	0.196101		0.009235
0.14	0.183363		0.001539
0.21	0.178335		0.000931
0.23	0.179971		0.002075
0.24			

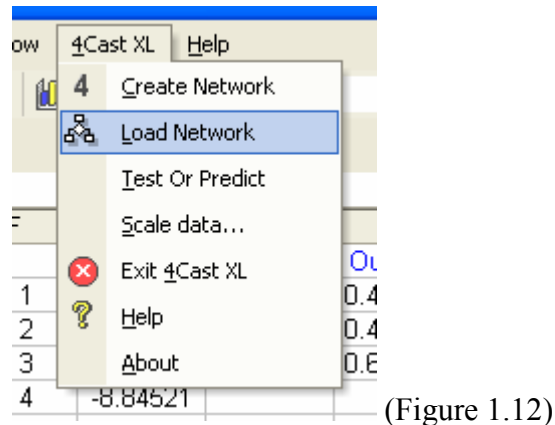
(Figure 1.11B)

As the result is not satisfactory, we need to continue training the network model.

Step 4: Retraining the Network Model.

Here we do not need to rebuild the whole network again. Make sure you have select the sheet that contain the training data as the active sheet before execute the step below.

- Select **Load Network** on the **4Cast XL** menu (see Fig 1.12 below) and the dialog box **4Cast XL (2 of 3)** will be displayed (see Figure 1.7 above)



(Figure 1.12)

- Choose the **Conjugate Gradient** training method and go through the whole process again until you've train the model for 20000 to 30000 epoch or when the MSE reach below 0.001.

Do not click on the **Initialize New Weights** unless you want to change a new set of weights. Here **4Cast XL** will use the weights contain in the weight i.e.G2:G19 generated from the previous training session.

(For advance user, you can change the weights vector by selecting the **Initialize New Weights** to initialize new weights if you feel that the current model does not have good starting weights)

If you have no confidence on the network model and need to change or create a new model altogether with different architecture then goto Step 5 else goto Step 6.

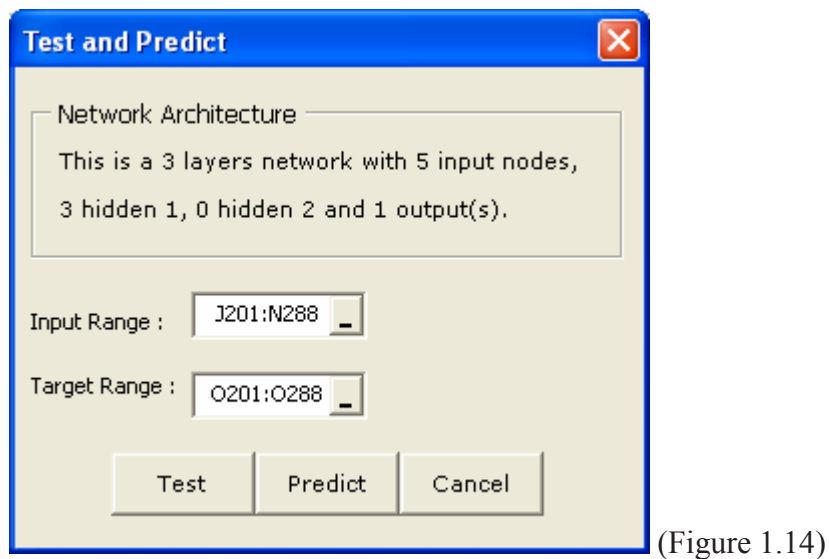
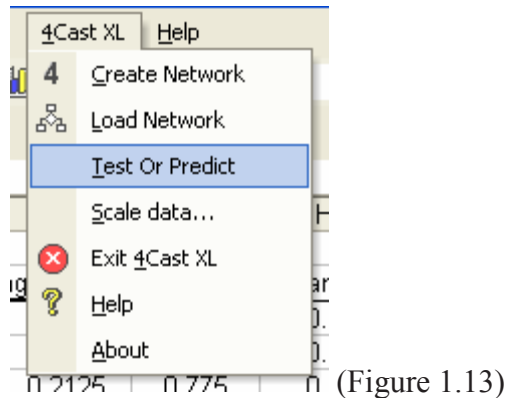
Step 5 : Re-Build the Neural Network Model

- i) It is advisable for user to use different workbook to create different network architectures.
- ii) Or user can rename the *Network Output* worksheet if new network architecture is being created in the same workbook. **4Cast XL** will also prompt user to rename the *Network Output* worksheet if the new model is being built on the same workbook.
- iii) Go through Step 2 , 3, 4 again until you are satisfied with the result.
- iv) If you are satisfied with the result then go to Step 6.

Step 6: Testing

After you are satisfied with the training result, its time to test the neural network model. For our current example, the network model was trained for 10,000 epochs and the MSE has reach 0.00113.

- i) Select **Test or Predict** on the **4Cast XL** menu (see Fig 1.13) and the dialog box **Test and Predict** will be display. (see Figure 1.14)



- ii) Enter J201:N288 in the **Input Range** edit box
- iii) Enter O201:O288 in the **Target Range** edit box

Remember that we have allocated row 201 to 288 as the test set.

- iv) Select the **Test** button. **4Cast XL** will run the test and pops up a message box displaying the Test Error and will also be stored on the **Network Output** worksheet (see Fig 1.15)

B	C	D	E
	MSE	0.00113	
	AE	0.224802	
	Test MSE	0.00676	
	No. Of Layer	3	
	No. Of HL1	3	
	No. Of HL2	0	
	Input Nodes	5	
	Output Nodes	1	
	Mean	0.203511	
	Std Dev	0.223358	

(Figure 1.15)

Here the test error is 0.00676 (see Figure 1.15) which is also display on the **Network Output** sheet. The MSE is 0.00113. The general rule here is that the test error and the actual MSE should not have a big difference. For example MSE = 0.00067 and Test error is 0.00089, this can be consider good modeling and suitable to be used as a forecasting model in real life scenario.

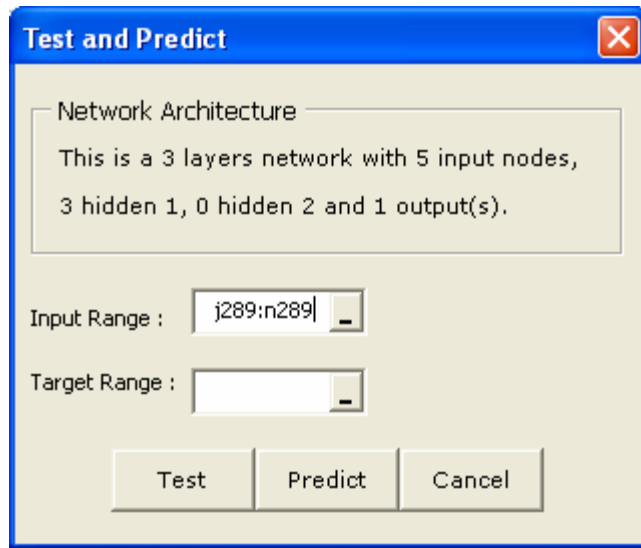
If you have no confidence on the result or after training for 20,000 to 30,000 epochs and the network model show no improvement, then you need to re-built the neural network and go through Steps 2,3,4 and 6. *Neural network forecasting requires lots of training and testing before you can apply it to real life situations.*

You will need some time to learn and to experience before you can gain the confidence to use neural network as a forecasting tool. Once you've became an expert user, you will be amaze when neural network shows its magic!!!

The result of the current model is quite satisfactory, after training and testing, its time now to do the prediction.

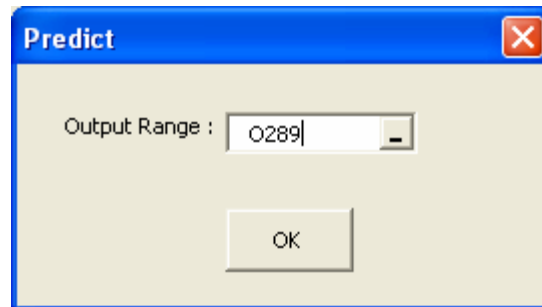
Step 7: Predicting

i) Select **Test or Predict** on the **4Cast XL** menu (see Fig 1.13 above) and the dialog box **Test and Predict** will be display. (see Figure 1.16 below). Enter J289:N289, i.e. the 5 input factors to predict the outcome/result (i.e. 1 row of MA5, MA10, MA20, MA60 and MA120). Leave the **Target Range** blank



(Figure 1.16)

- ii) Click on the **Predict** button to bring up the **Predict** dialog box (Figure 1.17), enter O289 so that the result/output of the prediction will be contained on this cell.



(Figure 1.17)

- iii) Click the **OK** button and the result will be displayed on the cell O289 in blue color font. The Confidence Level of 86.29% is also shown next to it. (Figure 1.18)

287	10/8/2007	0.97	0.93	0.92	0.82	0.69	0.92		
288	10/15/2007	0.97	0.93	0.92	0.82	0.69	0.94		Confidence Level
289	10/22/2007	0.97	0.94	0.92	0.83	0.69	0.93		86.29%
290									

(Figure 1.18)

- iv) Next we need to convert this result/output back to the Dow Jones weekly price. (as 0.93 doesn't mean anything to us)

Select **Scale Data** on the **4Cast XL** menu (see Figure 1.1 above) and the dialog form below (Figure 1.19) will be shown: (we also do this step when we first convert the raw data)

271	6/18/2007	0.93	0.93	0.94	0.73	0.63	0.91	
272	6/25/2007							
273	7/2/2007							
274	7/9/2007							
275	7/16/2007							
276	7/23/2007							
277	7/30/2007							
278	8/6/2007							
279	8/13/2007							
280	8/20/2007							
281	8/27/2007							
282	9/4/2007							
283	9/10/2007							
284	9/17/2007							
285	9/24/2007							
286	10/1/2007							
287	10/8/2007	0.97	0.93	0.92	0.82	0.69	0.92	
288	10/15/2007	0.97	0.93	0.92	0.82	0.69	0.94	Confidence Level
289	10/22/2007	0.97	0.94	0.92	0.83	0.69	0.93	86.29%
290								
291		14093.08						
292		7528.40						

(Figure 1.19)

Remember that **4Cast XL** has stored the minimum in cell J292 and maximum in cell J291 of the raw data when we first convert the data in Step 1. Now we reverse the process. (see Figure 1.19)

- v) Click the **Scale Now** button and we have the predicted result as **13582.49**, (**0.93** become 13582.49...see Figure 1.20 below)

287	10/8/2007	0.97	0.93	0.92	0.82	0.69	0.92	
288	10/15/2007	0.97	0.93	0.92	0.82	0.69	0.94	
289	10/22/2007	0.97	0.94	0.92	0.83	0.69	13582.49	
290								

(Figure 1.20)

To conclude, **4Cast XL** predict that next week DJIA closing price is **13582.49**.

Example 2: Sales Forecasting

Here is a real life data sample taken from a fashion store selling women clothing. 2 years i.e. 104 weeks of historical data are used to train and predict the weekly and daily sales of the fashion store. This example will show neural network as a non-linear application can predict better than classical linear model due to their ability to find non-linear dependencies in data.

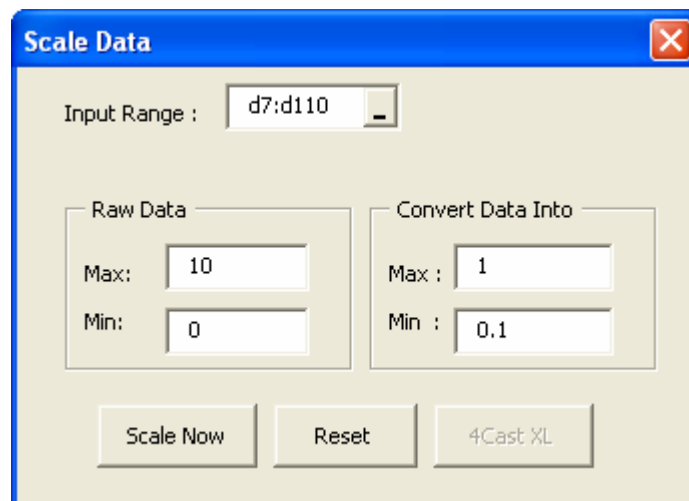
Sales distribution among the weeks and days appear to be random with no definite rules. After proper training the network is capable of forecasting weekly and daily sales much better than classical models. (Open the file Sales_Forecasting)

In the worksheet “Raw Data”, I’ve manually transform the data from Column B to column E into numerical value so that they can be read by **4Cast XL**. For example, in Column B for the “Season Influence” input, a “Low” category is transform to the value 0.25, a “Medium” category is transform into the value 0.5 and so on. User can view all the transform in the worksheet “Transform Data”.

Now you need to transform the numerical value in Column D to G into the range 0 to 1 as **4Cast XL** use a sigmoidal activation function as its output. Therefore, user need to scale the input data first so that the target activations to values can be comfortably learned. However you need to do this one column by one column as the data are categorical different unlike Example 1 above. Let’s start with Column D.

Step 1: Preparing the Data

Like Example 1 above, select **Scale Data** on the **4Cast XL** menu and the input form (see Figure 2.1 below) will be shown. Enter D7:D110 in the Input Range and press the TAB button on your keyboard and **4Cast XL** will automatically load the minimum and maximum value of the Column D. Convert the raw data into 1 for maximum and 0.1 for minimum. (See example in figure 2.1 below)



The image shows a dialog box titled "Scale Data" with a close button in the top right corner. The "Input Range" field contains "d7:d110". Below this are two columns of input fields. The left column, labeled "Raw Data", has "Max:" set to "10" and "Min:" set to "0". The right column, labeled "Convert Data Into", has "Max:" set to "1" and "Min:" set to "0.1". At the bottom of the dialog are three buttons: "Scale Now", "Reset", and "4Cast XL".

(Fig 2.1)

Click the **Scale Now** button, and the raw data will be converted. (see Figure 2.2). The raw data minimum and maximum value will be stored automatically below 2 rows of the last non-blank cell, and in this case cell D112 for the raw data maximum and D113 the raw data minimum.

D	E
0.37	0
0.64	2
0.82	2
0.19	1
0.55	2
0.1	0
0.73	1
0.91	2
0.37	1
0.55	2
0.28	0
0.19	2
0.64	0
0.55	1
0.82	3
0.73	3
0.55	1
0.73	0
0.91	1
0.55	1
0.46	1
0.37	0
0.64	1
0.46	1
0.1	1
0.1	0
0.55	0
0.37	0

(Figure 2.2)

Continue this process of scaling or transforming the data for Column E to Column G. For the Column I and J which are the targeted value, you can scale them in one go as they comprise values from the same category i.e. sales.

All the scale data are stored in the worksheet “Transform Data” and these values are use as the training data for our neural network model. Now we need to partition the data into 2 parts.

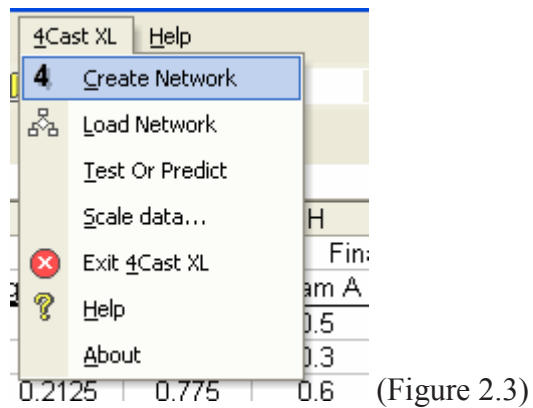
Part 1 - row 7 to row 87 as the training data

Part 2 - row 88 to row 110 as the testing data.

User can start to build the neural network model after this.

Step 2: Building the architecture of the neural network model.

- a) Select **Create Network** on the **4Cast XL** menu (see Figure 2.3) and the dialog box **4Cast XL (1 of 3)** will be displayed (see Figure 2.4).



E	F	G	H	I	J
Input 5	Input 6	Input 7		Target 1	Target 2
No of Special Offers	Advertising Budget	Number of Ads		Sales	Sales
Max = 5 Min = 0	Max = 49800 Min = 3400	Max = 27 Min = 1		(Weekly)	(Daily Sales)
0.1	0.963146552	0.757692308		0.82	0.17
0.46	0.910531403	0.961530463		0.51	0.12
0.				0.50	0.12
0.				0.70	0.15
0.				0.41	0.11
0.				0.44	0.11
0.				0.59	0.13
0.				0.44	0.11
0.				0.61	0.14
0.				0.59	0.14
0.				0.42	0.11
0.				0.68	0.15
0.				0.37	0.10
0.				0.48	0.12
0.				0.39	0.11
0.				0.56	0.13
0.				0.42	0.11
0.				0.71	0.15
0.				0.38	0.11
0.				0.52	0.13
0.				0.62	0.14
0.1	0.511422414	0.576923077		0.57	0.13

4Cast XL (1 of 3)

Layers of Network :

Input Range :

Target Range :

Nodes

Inputs	Hidden 1	Hidden 2	Output
<input type="text" value="6"/>	<input type="text" value="12"/>	<input type="text" value="4"/>	<input type="text" value="2"/>

(Figure 2.4)

- Select 4 on the **Layers of Network** by clicking the spin button.
-
- On the **Input Range** edit box, enter the range B7:G87 (6 columns of inputs and 81 rows of input pattern, i.e. the training data). Each time you exit, please use the *Tab* key on your keyboard.
-
- On the **Target Range** edit box, enter the range I7:J87 (2 columns of outputs/target and 81 rows of pattern to be learned). *Tab!!!* When you tab to exit, **4Cast XL** will automatically load the number of input column in the **Inputs** textbox on the **Nodes** frame. (Here we've 6 inputs and 2 outputs. Most commercially available neural network software can only accommodate one output node)
-
- On the **Hidden 1** textbox of the **Nodes** frame, enter 12. This mean that we have select 12 nodes for hidden layer 1. The general rule to follow here is that we use the pyramidal scheme or the diamond rule.
-

- On the **Hidden 2** textbox of the **Nodes frame**, enter 4. This mean that we have select 4 nodes for hidden layer 2. The general rule to follow here is that we use the pyramidal scheme or the diamond rule. Here we have use 6 -12 - 4 - 2 (diamond rule) as the network architecture.
-
- The **Output** textbox on the **Nodes frame** is already loaded with the value 2 as this is done automatically by **4Cast XL**.
- Click the **Next** button.

Click on the **Next** button and the dialog box **4Cast XL (2 of 3)** below will be shown (see Fig 2.5). The worksheet **Network Output** will be created. This worksheet is very important as it is use to store the model architecture (display on the **Network Architecture** frame), the weights (w1 to w128) and all other important parameters. (see Figure 2.5)

C	D	E	F	G	H	I	J	K	L	M
MSE				Weights		Output 1	Output 2		Error 1	Error 2
AE			w 1							
Test MSE			w 2							
No. Of Layer	4		w 3							
No. Of HL1	12		w 4							
No. Of HL2	4		w 5							
Input Nodes	6		w 6							
Output Nodes	2		w 7							
Mean			w 8							
Std Dev			w 9							
Iterations:	-1		w 10							
			w 11							
			w 12							
			w 13							
			w 14							
			w 15							
			w 16							
			w 17							
			w 18							
			w 19							
			w 20							
			w 21							
			w 22							
			w 23							
			w 24							
			w 25							
			w 26							
			w 27							
			w 28							

4Cast XL (2 of 3)

Network Architecture

This is a 4 layers network with 6 input nodes, 12 HL 1, 4 HL 2 and 2 output(s). 81 patterns is feed to the network.

Training Methods

Backpropagation with Momentum
 Conjugate Gradient (Recommended)
 Genetic Optimization

Worksheet name : Tranform Data

Show MSE Graph

lit Column A !!!

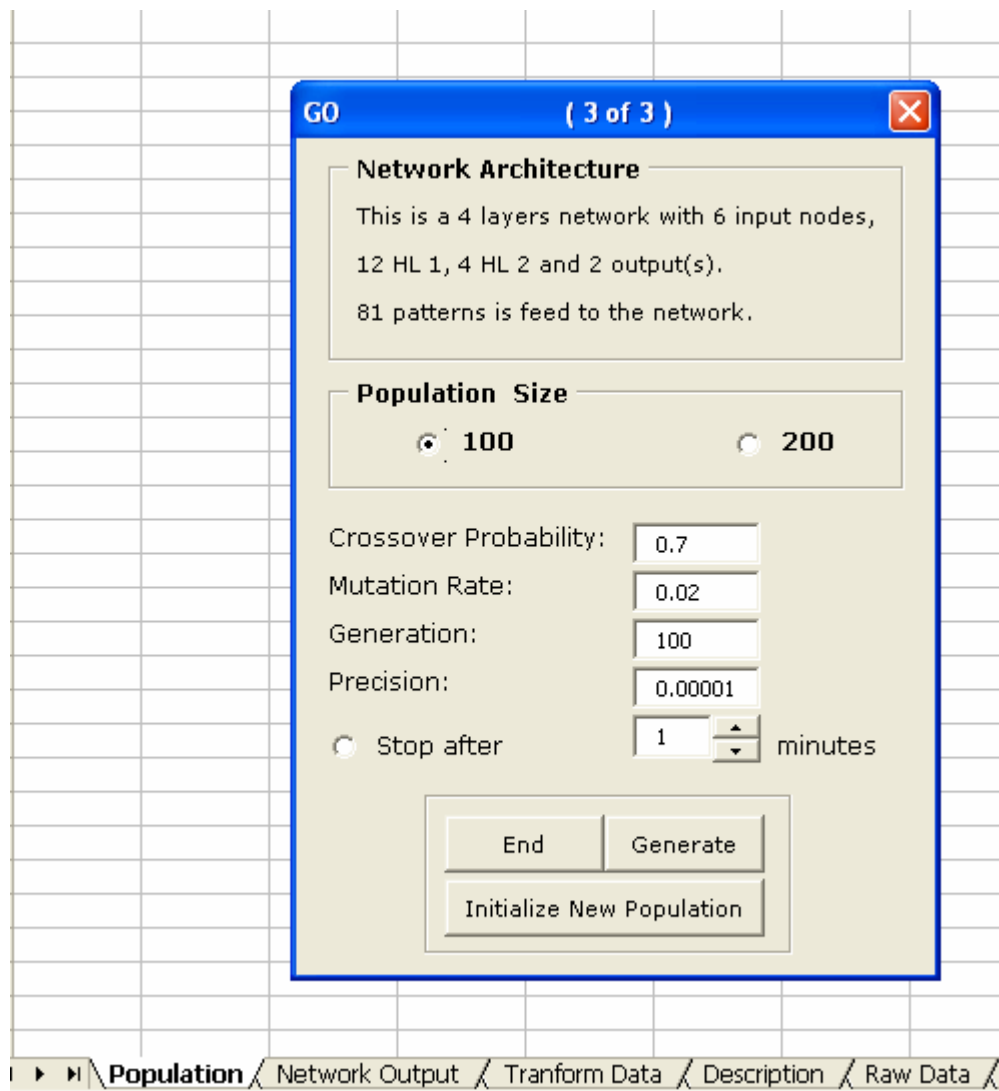
(Figure 2.5)

Currently, the worksheet is blank as we have not start training yet.

b) Select the training option that you want in the **Training Methods** frame. Training option includes **Backpropagation with momentum**, **Conjugate Gradient** and **Genetic Optimization**. Select the checkbox **Show MSE Graph** to show the training error in each epoch.

Choose the **Genetic Optimization** training method. Click the **Next** button again and the **Genetic Optimization** training method dialog form (see Fig 2.6 below) will be displayed.

Step 3: Training the network model



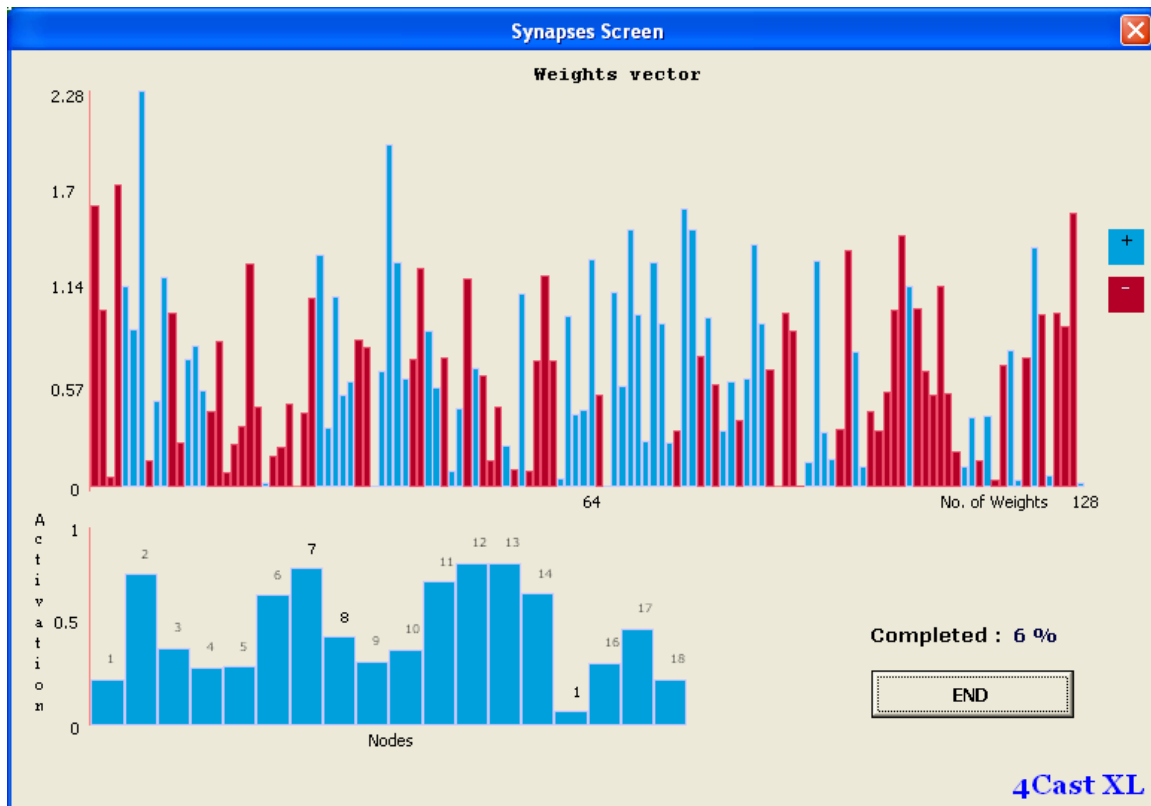
(Figure 2.6)

At the same time, the worksheet “Population” will also be created. (see Figure 2.6). It is recommended that you use the default value on this dialog form.

Click on the **Initialize New Population**. 128 rows of value for the weights and 100 (as indicated as the **Population Size**) columns of individual group of weights will be generated in this worksheet “Population”.

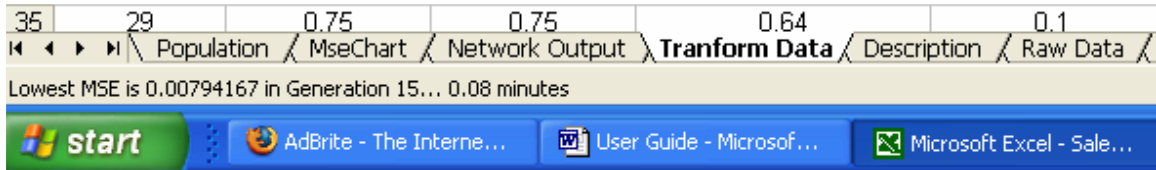
You also have the option to run the program for how many loops / **Generation** (maximum for this program is 15000) or you can select the option to run the program for how many minutes. (maximum 300 minutes). User can stop training prematurely by clicking the **End** button. (Note: MSE graph will not be updated if user do this. This might take a few seconds.)

Click the **Generate** button to start training the model. The **Synapses Screen** will be shown (see Fig 2.7 below) and the program will start running.



(Figure 2.7)

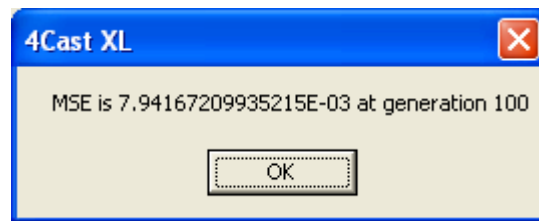
You can see the activities of the neural network model training. For experience user, they will know whether a suitable model has been built, whether enough hidden layers is created, whether the network has good starting weights etc, just by looking at this screen. **4Cast XL** is the only neural network program in the world that has this feature.



(Figure 2.8)

The number of **Generation** and the **Mean Square Error** (MSE) will be shown in the *Status Bar* (see Figure 2.8 above)

A message box will pop up after the model has finish training for the current session. (see Fig 2.9 below)



(Figure 2.9)

User can stop training prematurely by clicking the **End** button.(Note: MSE graph will not be updated if user do this. This might take a few seconds.)

Here it show 0.00794167 which is not satisfactory. Usually we need the MSE value to be below or close to 0.001 or after training for 10000 generations. As the result is not satisfactory we need to train the network model again. Before that, take a look at the **Network Output** worksheet. You'll see all the parameters generated on this sheet. Don't change anything here. Also, check out the Population worksheet.

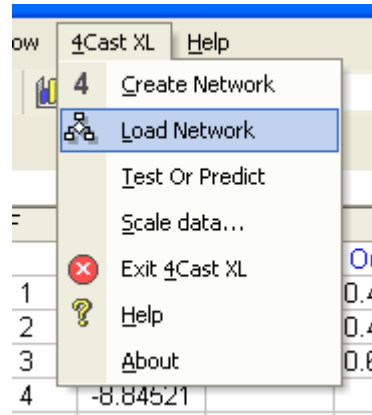
A complete analysis is presented in the **Network Output** worksheet. User can view and analyse the network model.(e.g. Each pattern output and error is presented thus, user can determine whether the network has learn well for a particular pattern.) In the hidden layer output, user will know which node is redundant or active. Weights of the network are also reported. User can also use the data here for graphing purposes, for example the *Actual Vs Forecasted chart*, *Sensitivity of Inputs chart*, *error distribution chart*, *correlation graph* etc.

As the result is not satisfactory, we need to continue training the network model.

Step 4: Retraining the Network Model.

Here we do not need to rebuild the whole network again. Make sure you have select the sheet that contain the training data in this case the “Transform Data” as the active sheet before execute the steps below.

- Select **Load Network** on the **4Cast XL** menu (see Fig 2.10 below) and the dialog box **4Cast XL (2 of 3)** will be displayed (see Figure 2.5 above)



(Figure 2.10)

- Choose the **Genetic Optimization** training method and go through the whole process again until you’ve train the model for 10000 generations or when the MSE reach below or close to 0.001.

Do not click on the **Initialize New Population** button unless you want to change a new set of population of weights. Here **4Cast XL** will use the weights population contain in the worksheet **Population** generated from the previous training session.

(For advance user, you can change the weights vector by selecting the **Initialize New Population** to initialize new weights population if you feel that the current model does not have good starting weights)

If you have no confidence on the neural network model and need to change or build a new model altogether with different architecture then goto Step 5 else goto Step 6.

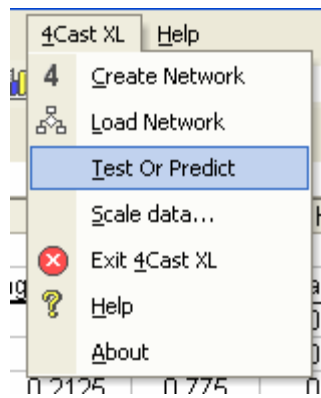
Step 5 : Re-Build the Neural Network Model

- i) It is advisable for user to use different workbook to create different network architectures.
- ii) Or user can rename the *Network Output* worksheet if new network architecture is being created in the same workbook. **4Cast XL** will also prompt user to rename the *Network Output* worksheet if the new model is being built on the same workbook.
- iii) Go through Step 2 , 3, 4 again until you are satisfied with the result.
- iv) If you are satisfied with the result then go to Step 6.

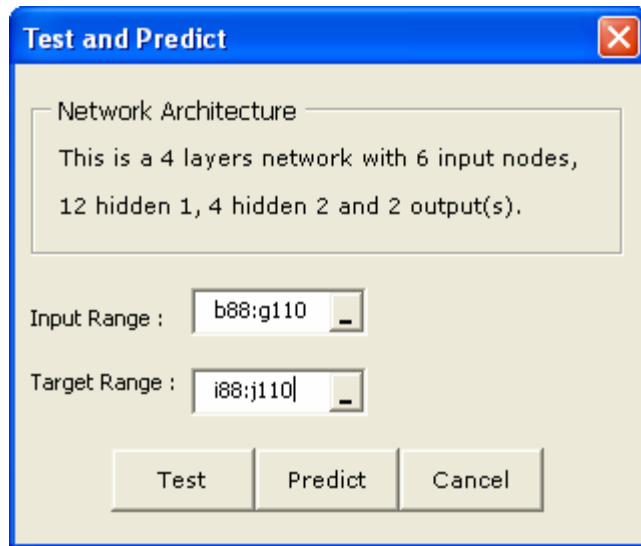
Step 6: Testing

After you are satisfied with the training result, its time to test the neural network model. For our current example, the network model was trained for 7,000 generations and the MSE has reach 0.00186, which is satisfactory.

- v) Select **Test or Predict** on the **4Cast XL** menu (see Fig 2.11) and the dialog box **Test and Predict** will be display. (see Figure 2.12)



(Figure 2.11)



(Figure 2.12)

- vi) Enter B88:G110 in the **Input Range** edit box
- vii) Enter I88:J110 in the **Target Range** edit box

Remember that we have allocated row 88 to 110 as the test set.

- viii) Select the **Test** button. **4Cast XL** will run the test and pops up a message box displaying the Test Error and will also be stored on the **Network Output** worksheet (see Fig 2.13)

B	C	D
	MSE	0.00186
	AE	1.300809
	Test MSE	0.00365
	No. Of Layer	4
	No. Of HL1	12
	No. Of HL2	4
	Input Nodes	6
	Output Nodes	2
	Mean	0.241542
	Std Dev	0.276124
	Iterations:	7000

(Figure 2.13)

Here the test error is 0.00365 (see Figure 2.13) which is also display on the **Network Output** sheet. The MSE is 0.00186. The general rule here is that the test error and the actual MSE should not have a big difference. For example MSE = 0.00067 and Test error is 0.00089, this can be consider good modeling and suitable to be used as a

forecasting model in real life scenario. We can use the current model as the difference between Test error and MSE is negligible.

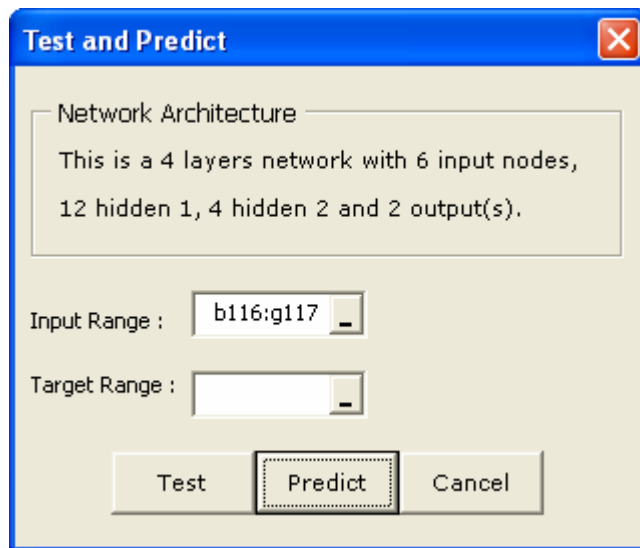
If you have no confidence on the result or after training for 7000 generations and the network model show no improvement, then you need to re-built the neural network and go through Steps 2,3,4 and 6. *Neural network forecasting requires lots of training and testing before you can apply it to real life situations.*

You will need some time to learn and to experience before you can gain the confidence to use neural network as a forecasting tool. Once you've become an expert user, you will be amaze when neural network shows its magic!!!

The result of the current model is quite satisfactory, after training and testing, its time now to do the prediction.

Step 7: Predicting

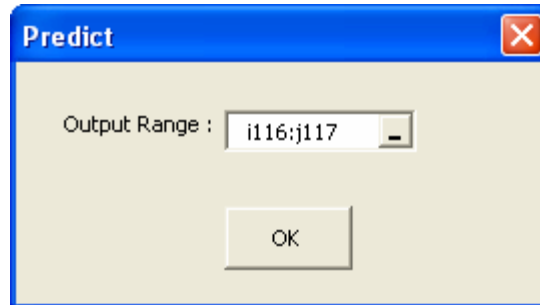
i) Select **Test or Predict** on the **4Cast XL** menu (see Fig 2.12 above) and the dialog box **Test and Predict** will be display. (see Figure 2.14 below). Enter B116:G117, i.e. the 6 input factors to predict the 2 outcome/result (weekly and daily sales). I've prepared these value in the cells indicated. Leave the **Target Range** blank



(Figure 2.14)

vi) Click on the **Predict** button to bring up the **Predict** dialog box (Figure 2.15), enter I116:J117 so that the result/output of the prediction will be contained on these cell. User can execute a batch prediction with **4Cast XL** as shown in

this case. Most of the available commercial neural network software do not have this feature.



(Figure 2.15)

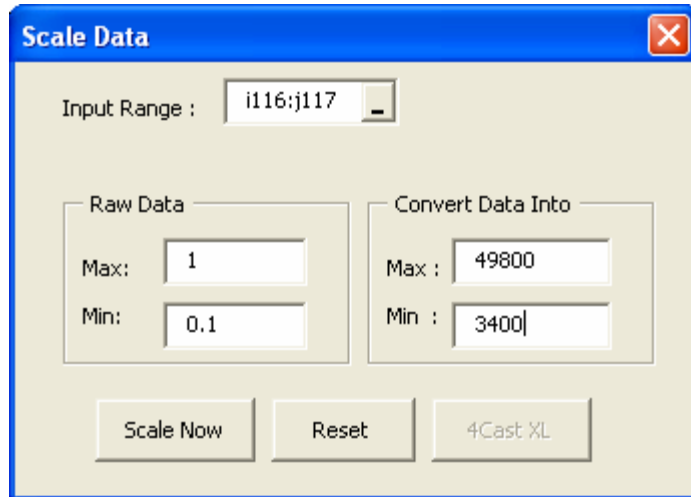
- vii) Click the **OK** button and the result will be displayed on the cell I116:J117 in blue color font. The Confidence Level of are also shown next to it. (Fig 2.16)
- viii) Here we have 0.612081 with 91.02% for the weekly sale and 0.12973 with 83.22% confidence level for the daily sale and so on.

114					
115				Confidence Level	
116	0.612081302	0.129735312		91.02%	83.22%
117	0.465511169	0.122578186		79.14%	90.32%
118					

(Figure 2.16)

- ix) Next we need to convert this result/output back to the raw data as the current values do not mean anything to us.

Select **Scale Data** on the **4Cast XL** menu (see Figure 2.1 above) and the dialog form below (Figure 2.17) will be shown: (we also do this step when we first convert the raw data)



(Figure 2.17)

Remember that **4Cast XL** has stored the minimum in cell F113 and maximum in cell F112 of the raw data when we first convert the data in Step 1. Now we reverse the process. Enter the value shown above. (see Figure 2.17)

- x) Click the **Scale Now** button and we have the predicted result in cells I116:J117 (see Figure 2.18 below)

	29800.636	4933.020555
	22244.13137	4564.030929

(Figure 2.18)

To conclude, **4Cast XL** predict the weekly and daily sales as the values show above in Figure 2.18

