

STOCK ASSAULT 2.0

Artificial Intelligence Stock Market Software

SideKick Edition

Welcome To Stock Assault 2.0 SideKick Edition!

Stock Assault 2.0 SideKick Edition is a software program intended for the prediction of situations originating in the markets and generating predictions for proper purchase and sale moments.

The off-the-shelf algorithmic developments in neural net theory, together with the optimally calculated parameters of prediction models, have allowed us to create a unique software product. Complete visualization of data, the possibility of saving/restoring/creating new neural net models, complete on-line support and updated stock quote data through the Internet are just some of the possibilities of Stock Assault 2.0 SideKick Edition.

Stock Assault 2.0 SideKick Edition will show you all the power of artificial intelligence and non-standard analytical abilities in such difficult areas as the predictability of a market change.

Stock Assault 2.0 SideKick Edition is an application for traders, investors and brokers. It is an advanced charting tool with a predictive core based on artificial intelligence technology using neural networks. Stock Assault 2.0 SideKick Edition will help you to make decisions on entering and exiting the market, on buying your favorite stock or selling it at the best moment, for maximum profit. A core-complex neural network structure is used for training and forecasting. The data used for building neural network models is stock quote data with complex pre- and post- processing, noise removal, filtration, normalization, dynamic data reduction, etc. This transformed data is used for training models. For more detailed information about neural network principles see chapter [Neural Network Theory Introduction](#).

Stock Assault 2.0 SideKick Edition will help you to feel the pulse of the stock market and get recommendations of the best BUY/SELL moments.

The program is highly tailored and specially developed for both professional stock traders and beginners.

Advanced structures, algorithms and modern research resulting in neural net modeling make it possible to offer this product specially designed for traders and investors.

Applications, based on neural net technology, are often more effective, than the most professional trader. Such applications make heuristic analysis of stock market behavior, predicting a direction for the market and giving buy/sell signals on some time interval. The professional trader, using the given program, can predict with a most exacting degree of success.

The program has been highly effective on the most risky securities markets, in unstable and analytically unpredictable markets.

This professional application allows you to reduce the risk of investing, and enables easier decision making. The series of tests which have been carried out by professional traders, have allowed us to improve the internal structure neural net models, and using the newest algorithms, to achieve excellent results in the prediction of the

movement of the market.

A free online data source allows you to create models online, and update them daily... getting new recommendations for buy, sell or hold decisions.

A professional statistical module can give you all details about your model's accuracy, profitability and profit ratios, ROA, PRR, etc.

Stock Assault 2.0 SideKick Edition has an easy-to-use interface, which gives you the ability to create a model with just a few clicks, train the model and generate forecasting (without a knowledge of neural network theory and deep math analysis.)

You can create your own portfolio of models, open them, update them with new stock quotes, train, estimate accuracy and profitability, get forecasting signals, print stock charts with forecasting results and examine your current open trading positions, all extremely easily, with a user-friendly interface.

Neural Network Theory Introduction

For a long time, the most interesting area of the science known as neural networks was in a shadow. Only recently has it received the wide recognition it deserves, as needs arose to process huge masses of data. In the first years of development, neural networks were considered strategically important, and the products, both software and hardware, were made by order, mostly, for the government and military industry. Today, neural networks are beginning to be much more common. Neural networks are now used in the positioning of battle units (tanks, missiles, etc.), on satellite snapshots from space (military industry), for the prediction of the election returns (sociology), for the detection and diagnosis of disease (medicine), and the prediction of stocks on the basis of historical data (stock market).

In part, the neural network model emulates the operation of the brain of a human. Its structure and the principles of operation are similar. Therefore neural net models are capable of generalizing various sets of data, of catching hidden, occasionally even unpredictable, dependencies, and producing outcome, which is, in large degree, authentic.

The principle of neural net operation consists in the following: you have a set of input data, and according to this data, a set of output data. And you can make the supposition, that between input and output data, there is a hidden link. For example, it is possible to assume, that there is a dependence between today's price for fuel, and the sales volume of automobiles in one month. In the learning process, the network is set up so as to minimize the error obtained at any incongruity of outcome produced by the network. Processing a huge amount of such examples, the network starts to catch dependencies of output data from input data. There is no necessity to be a professional in the area of neural net simulation, technical analysis and/or trading to use this software. Simply download the necessary historical data from the Internet and start the training mode. The neural network will then do the rest.

[You can read more about Neural Network Theory.](#)

Links For More Information About Neural Networks

For more info on the theory of neural networks, please see the following links:

Link 1: <ftp://ftp.sas.com/pub/neural/FAQ.html>

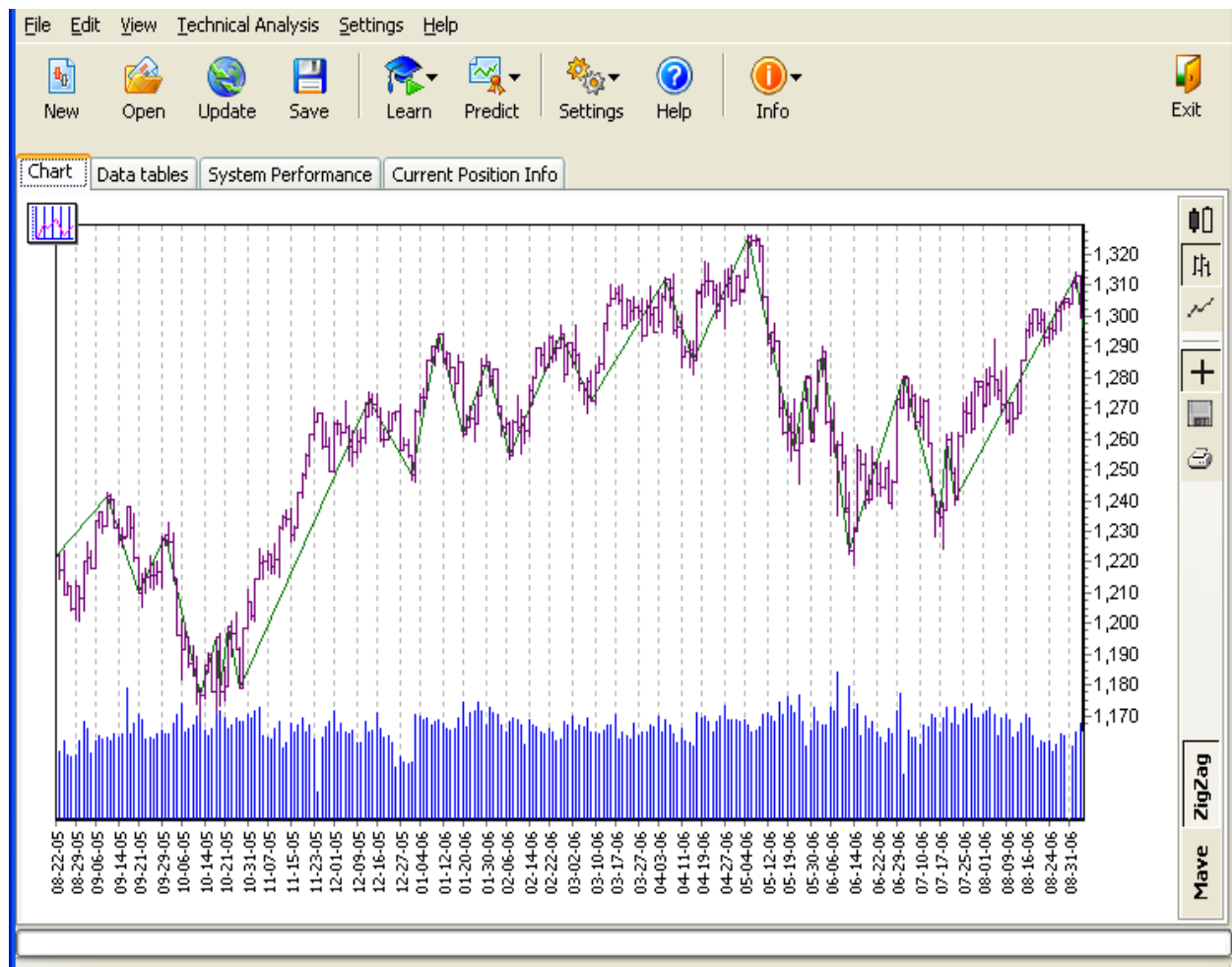
Link 2: <http://www.sigma-research.com/bookshelf/default.htm>

Link 3: <http://kevin-gurney.staff.shef.ac.uk/?p=Home.xml>

Link 4: http://en.wikipedia.org/wiki/Artificial_neural_network

Link 5: http://en.wikipedia.org/wiki/Artificial_intelligence

Chart



The main part of the user interface is the stock chart, on which you can see historical quotes and forecasting results. On the right side of the chart is an additional instrument panel, which allows you to perform basic operations with the stock chart. You can switch among 3 different views of stock data: candlestick view, bar view and line view. Another button gives you the ability to show the crosshair or hide it. With a crosshair, you can easily track detailed information about each bar or candle, by pointing the crosshair over it. Two bottom buttons are used for plotting the most common technical indicators over the chart, and taking a look at current market movement direction: ZigZag and Moving Average.

At the top left corner of the chart, there's the button for showing technical indicators menu. There're 13 customizable technical indicators and the tool for plotting trend lines.

The lower axis of the graph is the time axis and shows the date of the stock market data. The left axis shows the value of the stock market data. You can save your chart or print it, with a single click of the appropriate button on this right panel.

Moving and Zooming Chart

Moving and zooming charts in Stock Assault 2.0 SideKick Edition is very easy. As you can see, there's no scrollbars on the chart, because none are needed. You can move the chart using the mouse. For operating with movement, click and hold the right button of the mouse on the chart and move the mouse in any direction you wish to move the chart.

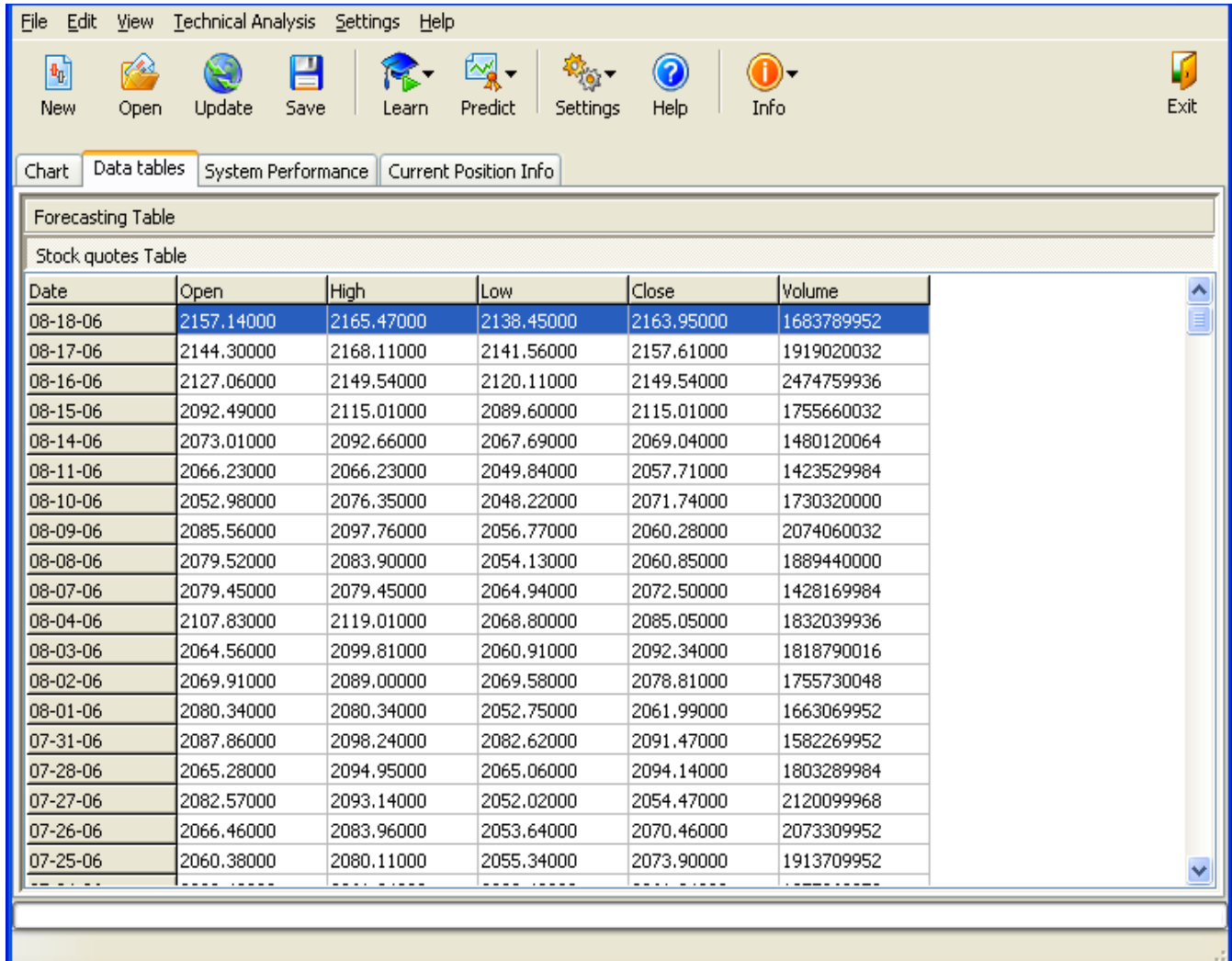
For zooming in, you can simply drag out a rectangle for which you wish to zoom-in on. Dragging out a rectangle is performed with the left mouse button. Press it and hold and then select the rectangle for zooming, by moving the mouse from the top left direction to the bottom right. After releasing the button, the selected rectangle will be zoomed in.

And, for un-zooming and going into the initial charting position, the same procedure is used, but selecting the direction should be the opposite, from the bottom right corner to the top left. After a few tries to move, zoom, and un-zoom, you will find that this way is much more usable than standard scrollbars and additional zooming buttons.

Data Tables

On the next program tab, you will find data tables for all the graphical data you see on the chart. Here you'll see the stock quotes table, with historical prices of selected stock, and the forecasting table, which stores forecasting results that were made in the past.

Stock quotes page. The data is sorted by DATE field, and you can see the Open, High, Low, Close and Volume values for each market day from the downloaded history. If there is no VOLUME field in this table, it means that the Volume value is not accessible for the ticker you inputted during the creation of the model.



The screenshot shows a software application window with a menu bar (File, Edit, View, Technical Analysis, Settings, Help) and a toolbar with icons for New, Open, Update, Save, Learn, Predict, Settings, Help, Info, and Exit. Below the toolbar are four tabs: Chart, Data tables (selected), System Performance, and Current Position Info. The main area displays two tables: a 'Forecasting Table' (empty) and a 'Stock quotes Table' containing the following data:

Date	Open	High	Low	Close	Volume
08-18-06	2157.14000	2165.47000	2138.45000	2163.95000	1683789952
08-17-06	2144.30000	2168.11000	2141.56000	2157.61000	1919020032
08-16-06	2127.06000	2149.54000	2120.11000	2149.54000	2474759936
08-15-06	2092.49000	2115.01000	2089.60000	2115.01000	1755660032
08-14-06	2073.01000	2092.66000	2067.69000	2069.04000	1480120064
08-11-06	2066.23000	2066.23000	2049.84000	2057.71000	1423529984
08-10-06	2052.98000	2076.35000	2048.22000	2071.74000	1730320000
08-09-06	2085.56000	2097.76000	2056.77000	2060.28000	2074060032
08-08-06	2079.52000	2083.90000	2054.13000	2060.85000	1889440000
08-07-06	2079.45000	2079.45000	2064.94000	2072.50000	1428169984
08-04-06	2107.83000	2119.01000	2068.80000	2085.05000	1832039936
08-03-06	2064.56000	2099.81000	2060.91000	2092.34000	1818790016
08-02-06	2069.91000	2089.00000	2069.58000	2078.81000	1755730048
08-01-06	2080.34000	2080.34000	2052.75000	2061.99000	1663069952
07-31-06	2087.86000	2098.24000	2082.62000	2091.47000	1582269952
07-28-06	2065.28000	2094.95000	2065.06000	2094.14000	1803289984
07-27-06	2082.57000	2093.14000	2052.02000	2054.47000	2120099968
07-26-06	2066.46000	2083.96000	2053.64000	2070.46000	2073309952
07-25-06	2060.38000	2080.11000	2055.34000	2073.90000	1913709952

Model's Performance Estimation

The next tab (System Performance) contains all necessary information for estimating the accuracy and profitability of your model. This is professional detailed trading statistics, with descriptions of all trades. Depending on the quality of your model, you can achieve great results. When you ask the program to predict stock movement and get a set of signals, the performance tab is filled with statistical values and trades, and shows you your total net profit, percent profitable, profit factor, return on investment and many other parameters. Depending on these values, you can make a decision to recreate the model, retrain it or save it for further use.

File Edit View Technical Analysis Settings Help										
New Open Update Save Learn Predict Settings Help Info Exit										
Chart Data tables System Performance Current Position Info										
	NET	LONG	SHORT	Date	Bar#	Name	Price	Volume	Profit	
Total net profit	54705.14	25699.04	29006.10	05/09/06	179	S long exit	2339.708	43	161.250	
Gross profit	55390.66	25699.04	29691.62							
Gross loss	-685.52	0.00	-685.52	05/10/06	180	S short entry	2332.820	43		
Total # of trades	34	15	19	05/25/06	191	B short exit	2182.429	43	6466.813	
Percent Profitable	94	100	89							
Number winning trades	32	15	17	05/30/06	193	S short entry	2202.570	45		
Number losing trades	2	0	2	06/01/06	195	B short exit	2179.820	45	1023.750	
Largest winning trade	6466.81	5040.91	6466.81							
Largest losing trade	-655.64	0.00	-655.64	06/05/06	197	S short entry	2212.349	45		
Average winning trade	1730.95	1713.26	1746.56	06/15/06	205	B short exit	2098.369	45	5129.100	
Average losing trade	-342.76	0.00	-342.76							
Ratio avg win/avg loss	5.05	0.00	5.09	06/19/06	207	S short entry	2136.599	47		
Avg trade(win & loss)	1608.97	1713.26	1526.63	06/21/06	209	B short exit	2109.219	47	1286.906	
Max consec. winners	15	15	11							
Max consec. losers	1	0	1	06/26/06	212	B long entry	2126.458	47		
Avg # bars in winners	6	5	6	07/05/06	218	S long exit	2174.688	47	2266.810	
Avg # bars in losers	3	0	3							
Max intraday drawdown	-6965.13	-4244.15	-6965.13	07/06/06	219	S short entry	2156.300	46		
Profit factor	43.31	0.00	43.31	07/25/06	232	B short exit	2060.378	46	4412.366	
Max # contracts held	49	49	48							
Account size required	6965.13	4244.15	6965.13	07/27/06	234	S short entry	2082.570	48		
Return on account	785.41	605.51	416.44	07/28/06	235	B short exit	2065.280	48	829.919	
PRR	38.96	1270.90	19.21							
Open Position P/L	4365.11	4365.11	0.00	08/02/06	238	S short entry	2069.909	48		
				08/11/06	245	B short exit	2066.229	48	176.639	

There's the descriptions of System Performance Table fields:

Total net profit	The grossprofit + grossloss.
Gross profit	The total profit from all profitable trades.
Gross loss	The total losses from all losing trades.
Total # of trades	The number of trades.
Percent profitable	Percent of profitable trades is Winning Trades/ Total Trades * 100;
Number winning trades	The number of winning trades.
Number losing trades	The number of losing trades.
Largest winning trade	The amount largest winning trade.
Largest losing trade	The amount largest losing trade.
Average winning trade	The amount of the average winning trade.
Average losing trade	The amount of the average losing trade.
Ratio avg win/avg loss	The Average Win to average loss is Average Win / Average Loss
Max consec. winners	The maximum number of consecutive winning trades.

Max consec. losers	The maximum number of consecutive losing trades.
Avg # bars in winners	The average number of bars in winning trades.
Avg # bars in losers	The average number of bars in losing trades.
Max intraday drawdown	The maximum drawdown from a individual trade. Draw down is = Volume * Value of a Tick * (low price during the trade - high price the during trade) / Tick Size
Profit factor	Profit Factor For long trades profitfactor = Long Gross Profit / Long Gross Loss For Short Trades profitfactor = Short Gross Profit / Short Gross Loss
Max # contracts held	The largest volume traded
Account size required	Account Size Required to trade the account. if margin = 100 then the Account Size Required equals the maximum drawdown, else Account Size Required = maximum drawdown / (margin / 100)
Return on account	Return on account Short Account Return = 100 * Short Net Profit / Short Account Size Required Long Account Return = 100 * Long Net Profit / Long Account Size Required
PRR	$\frac{(((\text{Net Win} - \text{pow}(\text{Net Win}, .5)) / \text{Net Trades}) * \text{Net Gross Profit} / \text{Net Win}) / (((\text{Net Loss} + \text{pow}(\text{Net Loss}, .5)) / \text{Net Trades}) * (-1) * \text{Net Gross Loss} / \text{Net Loss})}{1}$ if second half of above equation is zero then PRR is the first half. The PRR is used to determine how good a system is. PRR values from systems can be compared against each other to determine the best trading systems.
Open Position P/L	Profit/Loss value of currently opened position, based on current stock price.

Current Opened Position Info

The last tab shows you your current opened position and recommendations for the next day open price. This recommendation is based on forecasting results, which you can also see on the chart. There, you'll see the information about operating symbol, current position, recommendation on the next day open, current position entry date, current stock price, current position entry price, duration of current position, current profit/loss and annual return account.

Here are the explanations for Current Position and Recommendation for the next open parameters.

Current position field shows you if you're in a SHORT or LONG position at present. So, if you get the signal a couple of days ago, and open the position, if that position is not closed yet, you will see that position in the field.

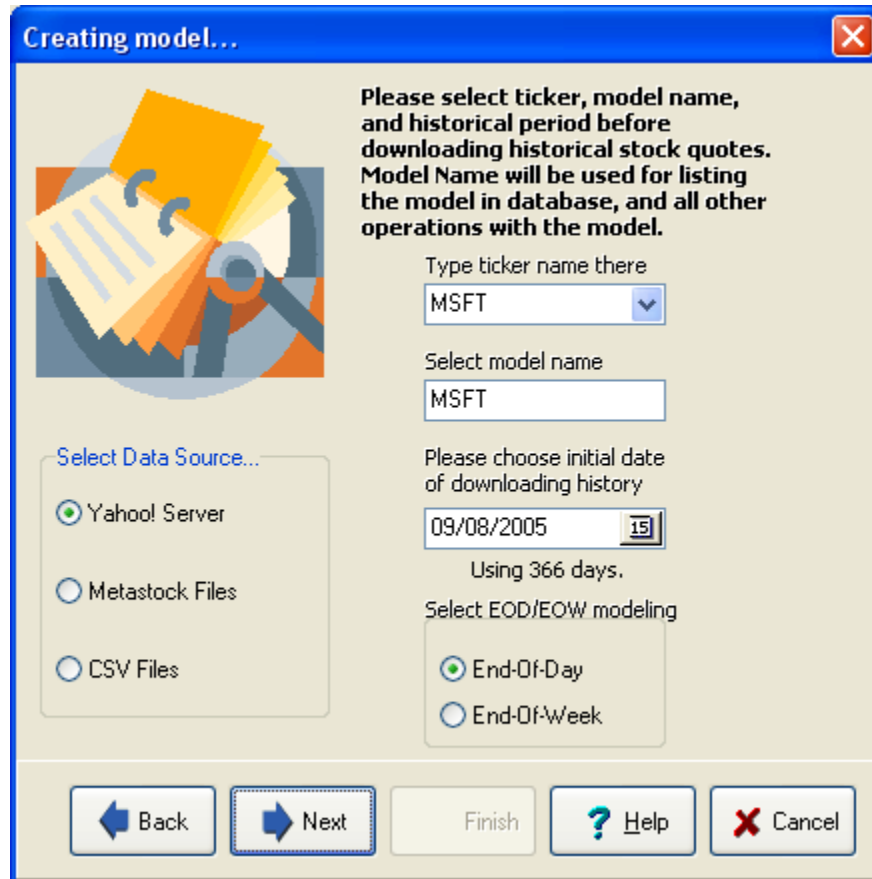
Recommendation for the next open is the current forecasting signal received from the neural network. So, if it says BUY or SELL, that means that the system forecasts that the price is going to move up or down and tomorrow's market open is the best moment to open the position or close the previous one. If Current Position field is NONE, and, you see SELL or BUY as the recommendation, that means that tomorrow at market open it is recommended to open new position, short or long.

If Current Position is LONG or SHORT, and you have SELL or BUY as the recommendation, it means that tomorrow it is recommended to close previously opened position.

Field	Value
Symbol:	NASDAQ
Current Position:	LONG
Recommendation for the next open:	SELL
Current Position Entry Date:	8/12/2006
Current Price:	2163.95
Current Position Entry Value:	2073.01
Duration Of Current Position, Bars:	5
Current Profit:	4386.8578
Current Loss:	...
Annual Return Account:	144.80

Creating Model

When you click the CREATE MODEL button or menu item, you will open the creating model wizard, which helps you to create a profitable model and prepare it for training and forecasting.



Creating model...

Please select ticker, model name, and historical period before downloading historical stock quotes. Model Name will be used for listing the model in database, and all other operations with the model.

Type ticker name there
MSFT

Select model name
MSFT

Please choose initial date of downloading history
09/08/2005 15

Using 366 days.

Select EOD/EOW modeling
 End-Of-Day
 End-Of-Week

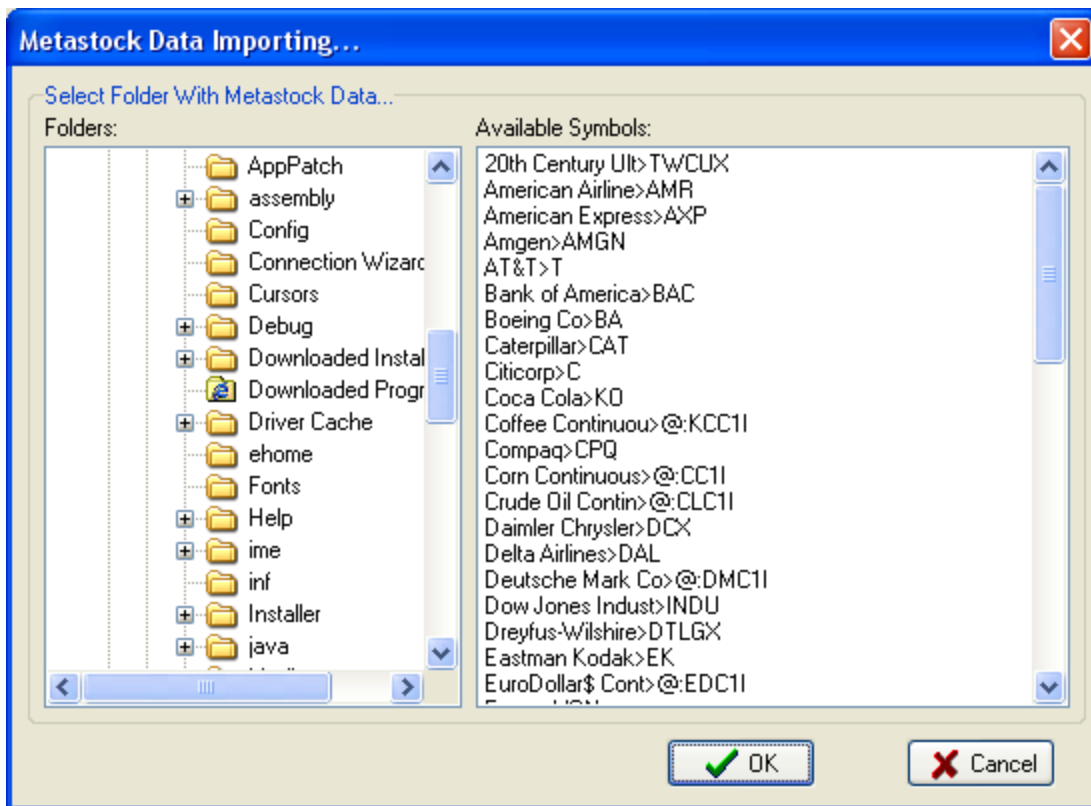
Select Data Source...
 Yahoo! Server
 Metastock Files
 CSV Files

Back Next Finish Help Cancel

Data Source.

You can select the type of the data you wish to use for creating the model and update it during operation. There are 3 data sources available:

- 1. Yahoo server.** When using this data source, historical quotes data for selected symbol is downloaded from Yahoo! finance server.
- 2. Metastock Files.** This data source supposes that you have Equis Metastock database stored at your PC, with historical quotes. When you select this data source, ticker name field is changed so that you will be able to choose which data record to use for building your model.



In the left pane, select the folder which contains your Metastock database. When you select the folder, right pane will be filled out automatically, with the list of available symbols and companies' names for these symbols. Select the symbol and click OK button. Selected DB record will be used for building the model and further updating of the model when you click UPDATE button.

3. CSV Files. This option will allow you to use Comma-Separated-Values data files for building the models and updating them. CSV file is the most common data file format, with the values separated by commas. Stock Assault 2.0 SideKick Edition uses fixed CSV file format, so you may need to rearrange your CSV data to be compatible with Stock Assault 2.0 SideKick Edition.

The format of the file which can be used for importing data and building the models is:

Date, Open, High, Low, Close, Volume, Adjusted Close

where Date is in format of:

d-mmm-yy (for example, "8-Sep-06")

Open, High, Low, Close values are floating point values, with point as decimal separator.

Volume is integer value without any separator.

Adjusted close is floating point value, but IT IS NOT REQUIRED for building the model. Adjusted close value is used for defining stock splits and dividends, and if you do not have such a field in your CSV data file, it will be duplicated by Close value automatically when importing the data from CSV.

You can find the example of CSV file format which can be imported into Stock Assault 2.0 SideKick Edition in "C:\Program Files\Stock Assault 2.0 SideKick Edition\CSV_Example\" directory.

Ticker name

The name of the stock you wish to create a model from. This should be a standard symbol name, for example, if you wish to create SP500 model, you should type ^SPC. If your name is not correct, the program tells you so and advises you to visit the given link for symbol lookup.

Model name

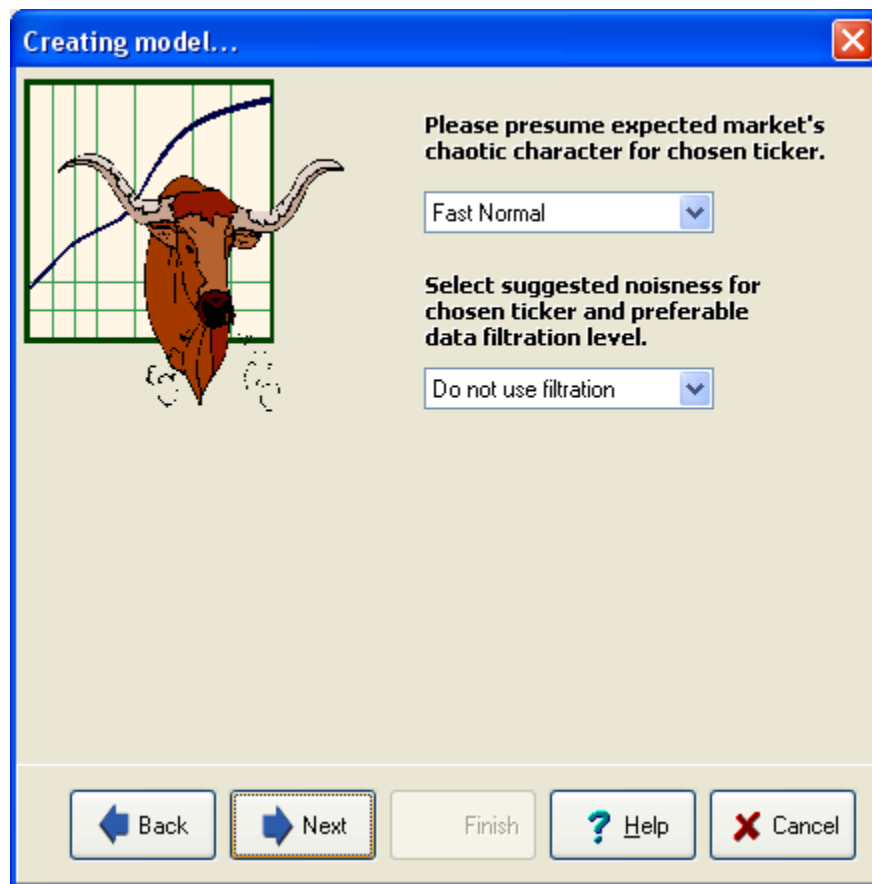
The name of the model with which your model will be saved and used. You can type any name you wish, with any additional info. For example, if you create a model for the SP500, with end of day prices and a small neural network structure, you can name it as SP500_EOD_SMALL. This will help you to operate with your portfolio of created models in further operating steps.

Initial date of downloading history

Here you can select which historical range you wish to download and use for training. The standard range is a 1 year history, but you can experiment with different periods.

End Of Day/End Of Week modeling

You can select which trading period you wish to use in creating a model. If you select End Of Week modeling, downloading data will be preprocessed and converted to a 1 week time frame.



Expected market chaotic character

This option is used for estimating the chaotic character for the selected ticker, from VERY CHAOTIC to VERY

INERT. This parameter determines how many days looking back the system uses for giving one prediction. In unstable markets, it's recommended you choose 'VERY CHAOTIC' and the system will build predictions based on only 1 day back data.

This parameter sets up how far in the past the model will look for building the forecasting. Technically, it means how many deltas will be used for modeling/forecasting, where delta is the difference between current quote and the quote n days earlier. For example, if the model is built to use open, high, low, close historical quotes, delta means current open-previous close or current high-low price 3 days ago and so on.

So, Chaotic settings will put 1 day backward deltas into the model-[current price-previous price].

Fast normal settings will put 2 days backward deltas, such as [current price-previous price] and [current price-2 days ago price].

Slow normal parameter sets up 3 days backward deltas.

Inert - 4 days backward deltas.

Very inert - 5 days backward deltas.

When using deltas, the model looks at the data in the past and based on that data, builds predictions. So, in other words, this parameter means how many previous days data will be used for building current signal (buy or sell recommendation).

The more the parameter - the more the suggestion that the current price is the result of price movements of previous days. So, if you suggest that the quotes are not changed chaotically for specified symbol and the changing of trend is able to be determined several days ago (doesn't matter how, but, previous bars precedes trend change), you may set slow normal, inert and very inert parameter. This way, you suggest that the price turn takes several days for specified ticker.

In case if the stock is not inert and the price turn does not take more than 2 days, you may set the parameter to chaotic/fast normal.

Noisiness and filtration level

This parameter selects estimated noisiness of the chosen ticker, and therefore, the filtration level for the price series.



This parameter describes initial data normalization level. There are several normalization methods used in the program, but for initial raw data, the program uses sigmoid function for normalization and scaling the data. Normalization is very important point in successful neural network learning process.

So this parameter means how the data will be normalized before feeding neural network structure. No filtration means that the data will be simply normalized by sigmoid function. Other parameters means that the stochastic 'peaks' in data will be smoothed. The higher filtration level will give you more peaks/holes smoothing, and therefore, remove the significance of accident prices jumping from neural network modeling. If you feel that there are many price jumps over the historical period for specified ticker, which has no reason and cannot be determined by looking at the past data, you may set this parameter to some filtration level. The more such a peaks/holes (accident unusual price jumps) - the higher filtration level you may wish to choose. By default, the parameter is set to 'no filtration', expecting that there's no accident major drops/raises of the price in selected historical period.

Dominant trading strategy

This parameter sets up the strategy of further forecasting.

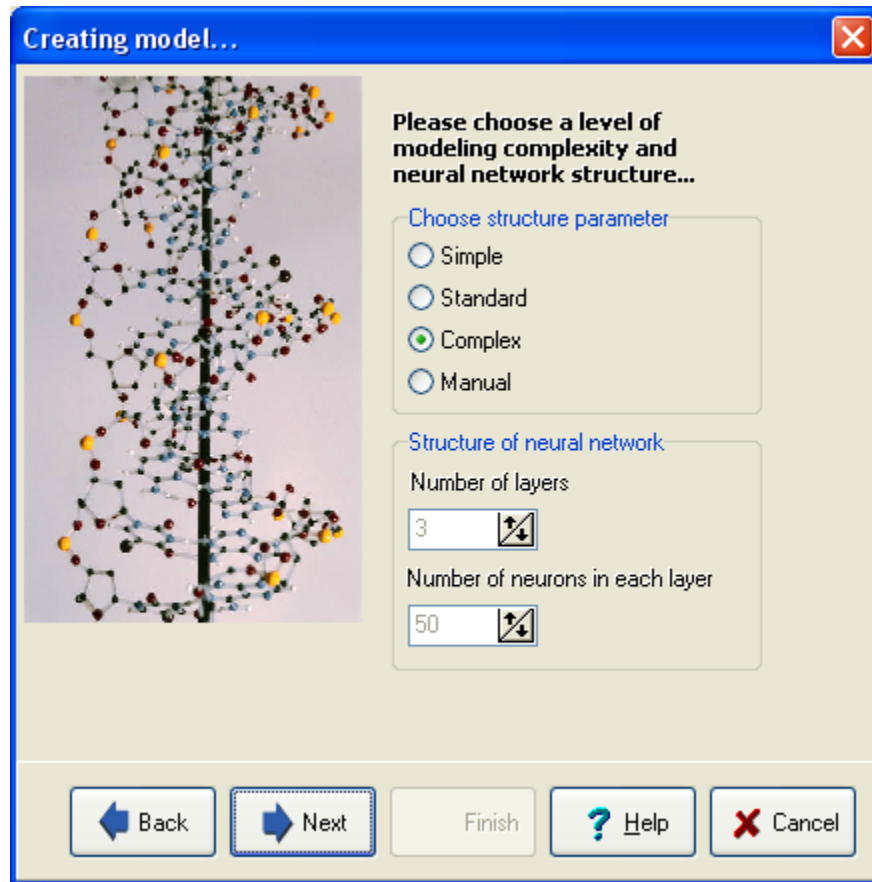
Speculative strategy will give more signals for some period, than Middle or Investment. If you wish to get signals almost every day, choose Speculative. If you wish to estimate investments with this system, choose Investment strategy (the signals will be infrequent).

Desired signals frequency

Here you can select the number of days between generated forecasting signals and control the approximate position duration (days between opening position and closing it).

In data processing module, the principle of waves is used. That means, that based on previously prepared data, there's additional data calculated named 'waves'. It is similar to moving averages, but, the resulting data is oscillation of values around PI value. This tangensoid waves are used in neural network modeling and learning process, and therefore, in forecasting. The effect which brings such a data transformation/applying, is that the more waves are used in modeling - the rarer signals you will get, following the rules of forecasting signal accepting/declining. So, using 1 tangensoid wave will give you more frequent signals because the weight of the signal from wave's signal generating rule will be high. Otherwise, if there're many waves are used in modeling, the signals will be spaced out, and will not be too frequent, as there should be strong signal supporting by every wave in the model.

For easier understanding, we name that parameter as 'desired signals frequency', as there's direct relation between numbers of waves and signals frequency.



Neural Network structure parameter

This parameter describes which neural network structure will be used for modeling.

There are 4 options:

1. Simple - this mode creates the NN structure with 3 internal layers, 15 neurons in each. The structure does not describe first, input layer and last, output layer. First layer may have different number of neurons, equal to input data sets which are used for learning and depends on all other parameters you select in creating model dialog wizard. Output layer may have different number of layers too, and usually, it has buy/sell/none signals at its outputs, from all signals generating rules used in the model.
2. Standard - the internal NN structure is 3 layers, with 30 neurons in each.
3. Complex - internal nn structure is 3 layers, with 50 neurons in each.
4. Manual - there you can select your own NN structure, if you feel that these 3 above is not optimal for your

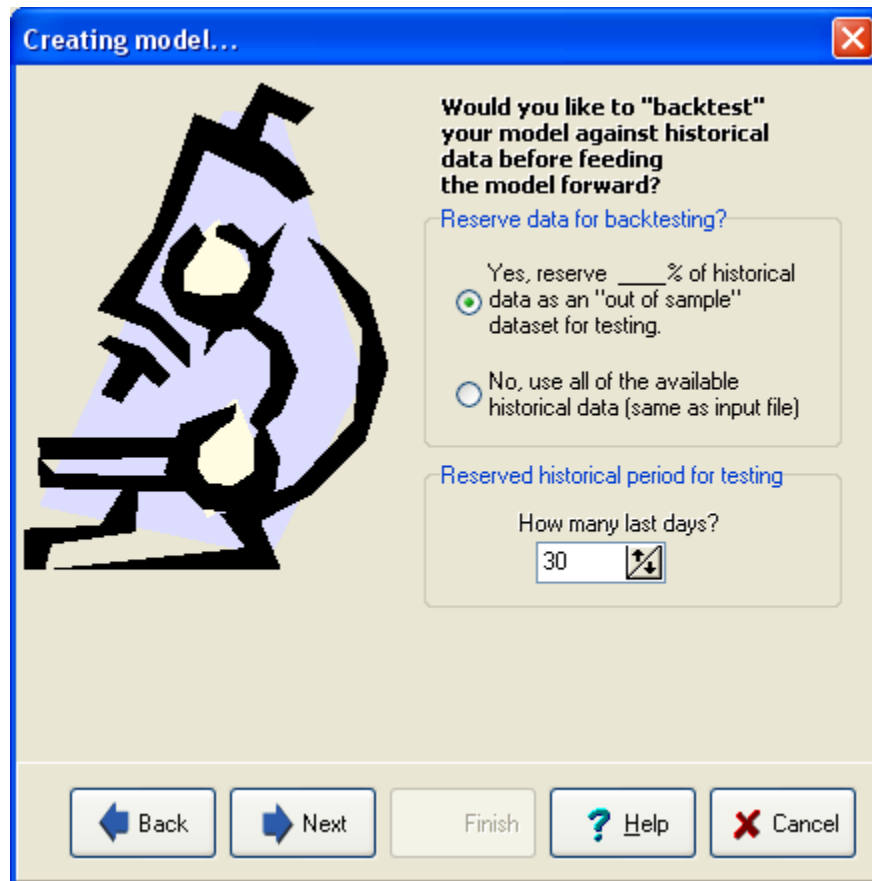
modeling. You can set up any dimension of neural network, with any number of layers and any number of neurons in each. For example, if you type 10-30-50-100-500-300, the program will create the NN model, with 6 internal layers and different number of neurons in each. Large structures take much memory and the time for their learning will increase in geometric progression.

Usually, large structures are used for very complex data, to determine the dependencies between input data sets and output data sets, when the correlation between input and output data verges towards zero. Also, large structures have more 'captious' memory, and therefore can be used for learning on long historical periods (3 years or more).

But, there's another side of using large structures. In case if the data which is used for learning is not too complex and there's no need to recognize millions of input combinations to generate correct output, NN structure will not be stable. So, even after deep learning, it may still generate incorrect output results because the power of such a model does not correspond to simple input/output data correlation and/or number of learning data sets.

If you use 1 year historical data for learning, you may set this parameter up to standard/complex mode, based on other parameters in modeling you select. For shorter periods, it may be successful to use simple structure. And for longer periods, you may select your own structure of neural network manually.

Reserve data for backtesting



This parameter allows you to select the part of the historical period which will not be used in training, but will be used for estimating a model's accuracy and profitability. In the case of using the backtest period, the last days are marked as an 'out of sample' period. If you select USE ALL DATA, all available data will be used in training.

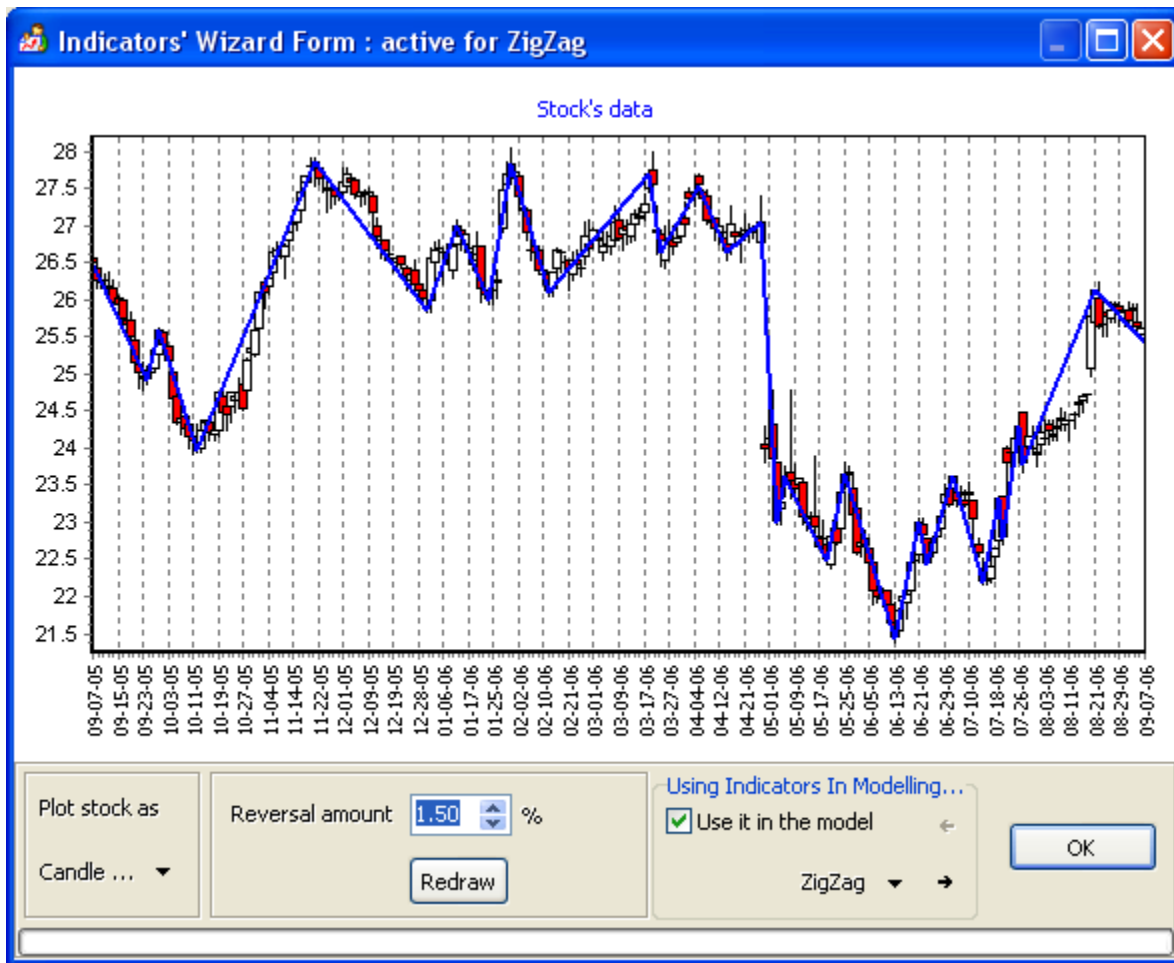
In creating a model, your computer must be connected to the Internet in order to download the required data.

After setting up all parameters for the model and clicking the FINISH button, the program will download historical prices from the Internet, make all necessary pre-processing, filtration, noise reduction, normalization, dynamic data reduction/induction, post-processing and save your model with the name you chose. After the model is created, you have to train it before it will be used for predicting the stock.

Configuring Technical Indicators In Creating Model Dialog Wizard

After creating a model, you can see historical quotes for your chosen ticker in the window. You can set up some parameters for technical indicators, applied in the system. Choose a reversal amount percentage for the ZigZag indicator, the type of Moving Average (Simple, Exponential) and the time period for the Moving Average. After pressing the OK button, all the data, necessary for the neural network model, are present. Further steps consist of training your model and getting forecasting results.

ZigZag Indicator Settings

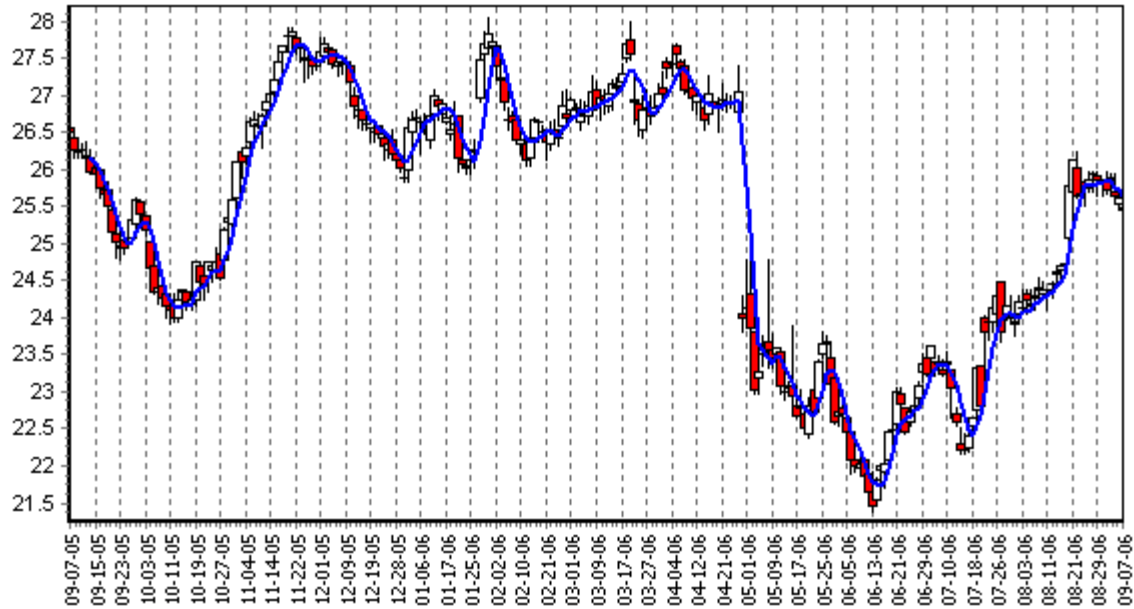


Moving Average Settings

Indicators' Wizard Form : active for Moving Average



Stock's data



Plot stock as

Candle ...

Moving Average

Simple

Time period

5

Redraw

Using Indicators In Modelling...

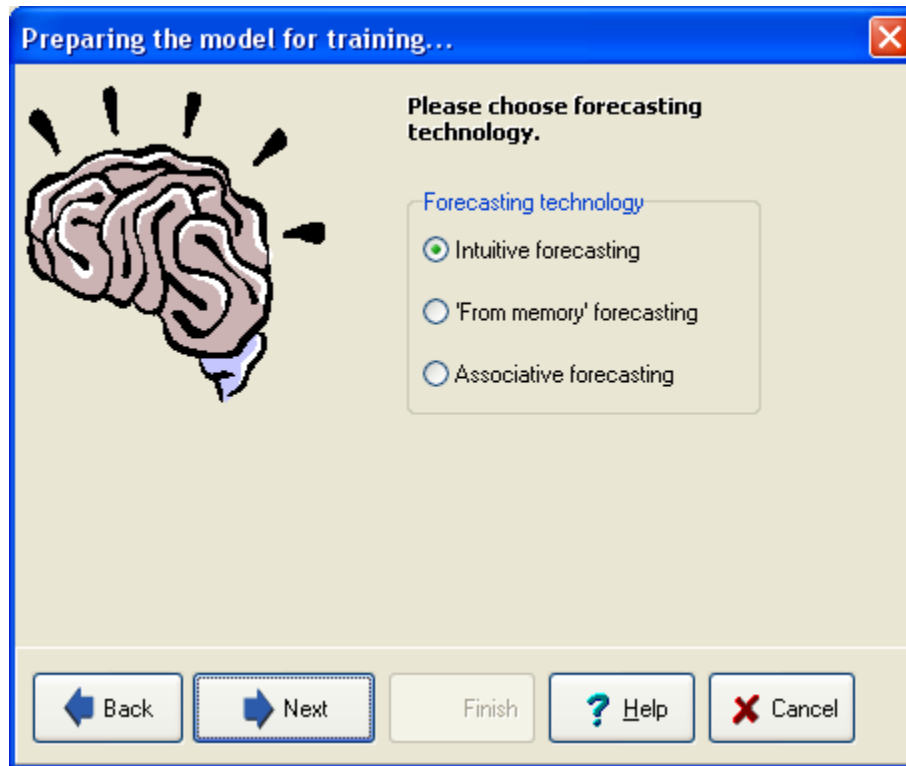
Use it in the model

Moving Average

OK

Training Model

Before the first use of your model in predicting a stock, you have to generate the initial training. Click the LEARN button or select START OR CONTINUE. Then a learning mode wizard will appear, which will help you to set up the learning parameters.



Forecasting technology

This parameter sets up how intensive your model will be learned.

The most important parameters which are changed by selecting different forecasting mode, is the number of iterations and error limit.

The number of iterations sets up how many iterations will be processed in learning, in other words, how many times the data sets will be pushed through the neural network and how strong they will effect NN structure memory.

Error limit parameter sets up the level when NN structure is considered to be learned already.

When using long historical periods (longer than 0.5 years) and large NN structures, it is recommended to use 'strong' and 'aggressive' learning mode, which is **Associative** forecasting mode.

In that mode, the number of iterations is 15.000 and error limit is smallest. This way, the learned model will use associations for forecasting and based on current input data set, it will try to associate it with all previous data sets which were used in learning process. If there's no association the model will not give any output result.

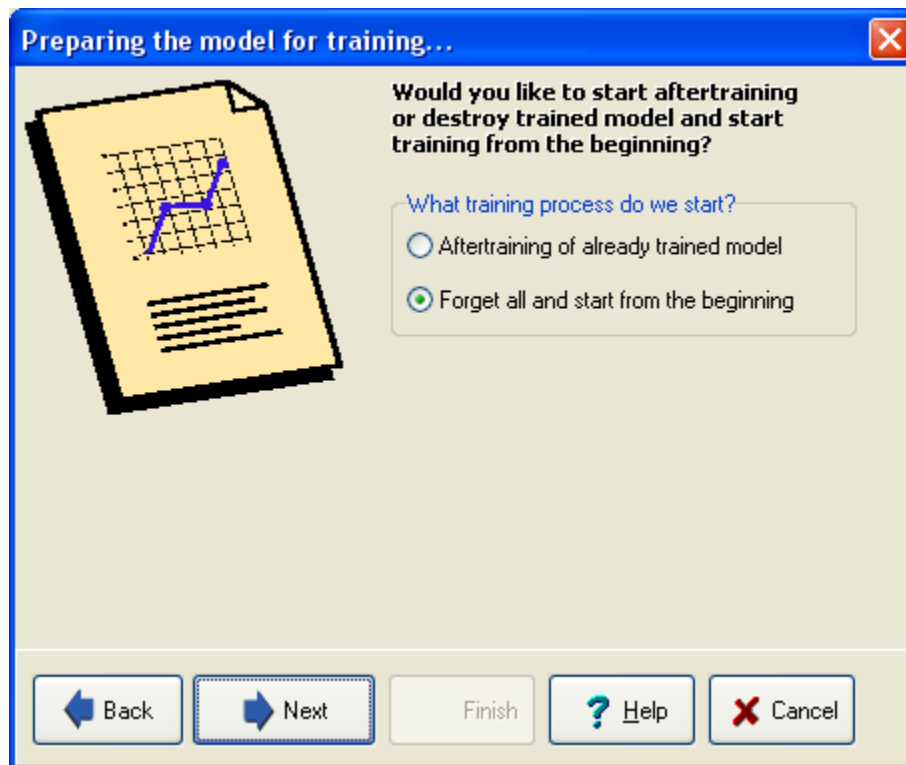
From memory forecasting mode has 5000 iterations and middle error limit. The model will try to recognize current input data set, by comparing it with all previously seen, which were used in learning process and by generalization of difference between this unknown data set and in-sample data sets.

Intuitive forecasting mode has 3000 iterations and error limit is largest. The effect of learning in that mode is not so aggressive for model and it learns to give intuitive forecasting on unknown data. The memory effect is not

strong and there's almost no defined dependencies between in-sample data sets. So, NN structure gives some kind of 'suggestions' based on unknown input data set, it tries to classify data set using degraded classification clusters.

So, when using large periods and complex models, associative forecasting is recommended. For middle range of structures (standard NN structure) you may wish to try 'from memory' mode.

For small NN structures and short historical periods (half a year of historical quotes, for example and small/normal NN structure), it may be useful to select 'intuitive' forecasting technology, as the number of data sets is not too large, and NN structure is small and can be set up to stable working state without thousands of iterations, which will put NN structure into overlearned state, which means that the NN model will not be able to give any adequate forecasting/results.



Initial Training/Aftertraining option

The Training process is automatic.

If you start learning for the created model, select FORGET ALL AND START TRAINING FROM THE BEGINNING.

If you wish to aftertrain the model with new updated historical data, select AFTERTRAIN.

If you already have a trained model and just wish to retrain it without any 'forgetting' of the previous training, choose 'Aftertraining'.

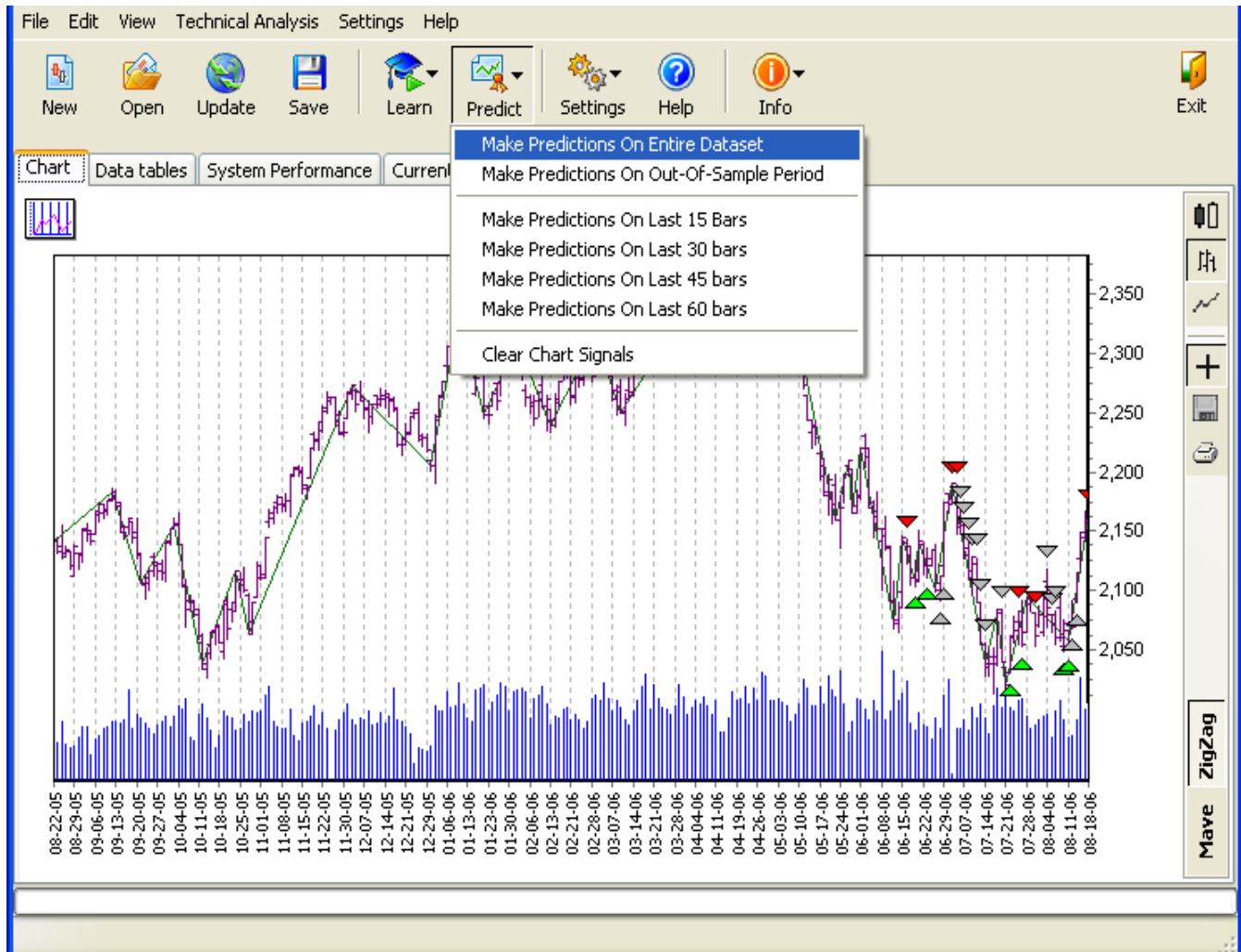
If you wish to train the newly created model, choose the training method with forgetting and randomizing the system.

After clicking the FINISH button, your model will be prepared for training and learning will start. It takes some time to learn the model with the data and it can vary depending on the settings you select in creating the model parameters and the learning parameters. Keep in mind that large NN structures take more time than small

structures. And, models that contain a long historical period of stock data can increase learning time.

Getting Predictions On Created And Learned Models

After the two first steps are done and you end up with a created, saved and learned model, you can get the forecasting results from your model.



Click the PREDICT button and in the popup menu, you can select different historical periods which are used for building the forecasting results.

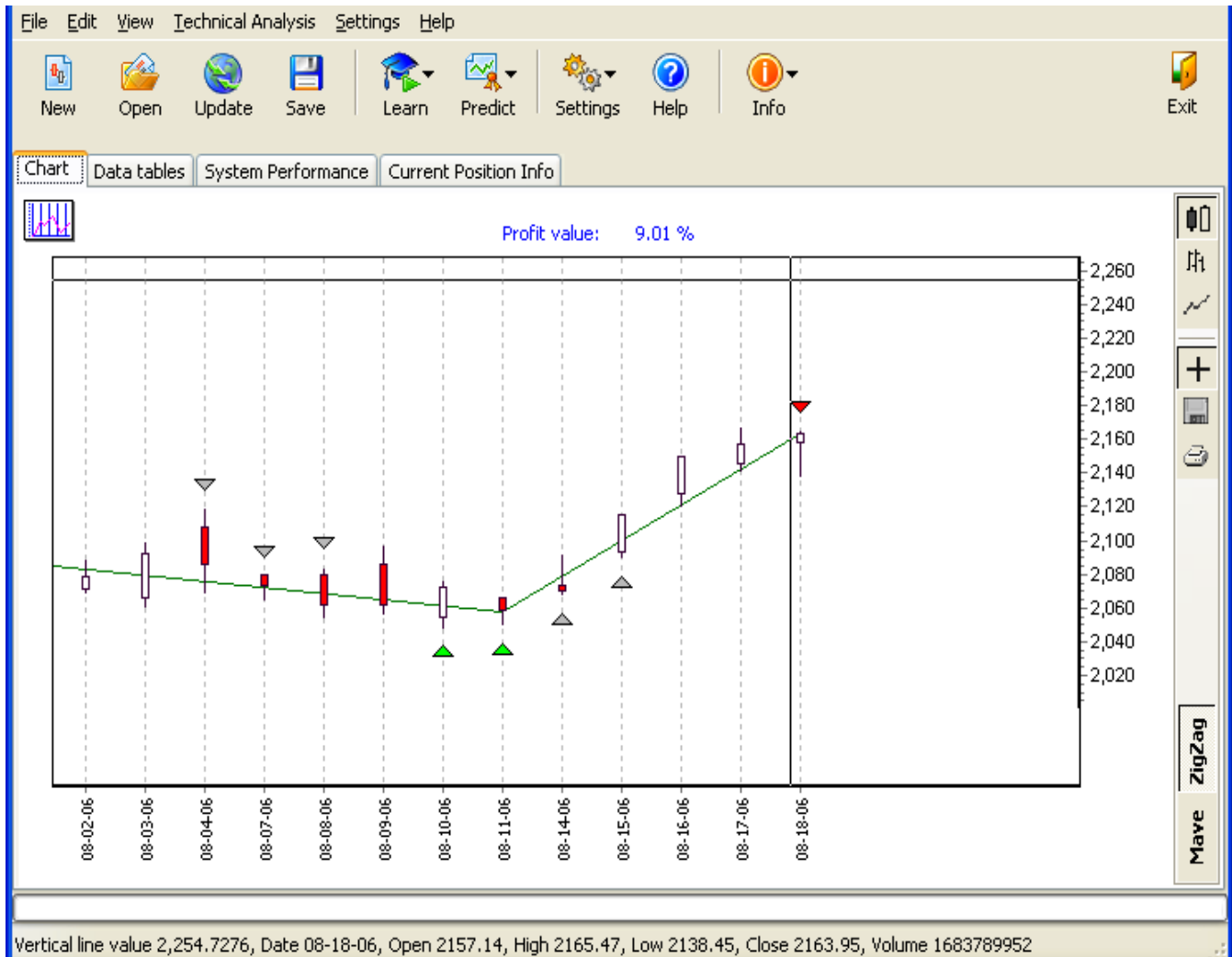
There're several options under prediction menu - for 15 bars, 30, 45 and 60. There's no difference between them, except one - that the forecasting signals are built for the last n days. You may use these options to determine if the model is successfully learned on different periods and the performance on these periods seems to be similar.

Other forecasting options - predict on entire dataset and predict on out of sample period - will give you the ability to determine the overall model's performance on the whole period used in modeling (entire dataset), on out-of-sample period which was not used in learning process and which is 'unknown' for the system like its freshly updated data. Looking at entire data set forecasting results, you can see how good is your model, how strong is its memory filled during learning and if it has successfully learned for recognizing data sets of in-sample period.

Out-of-sample period predictions show you how successfully you selected the model's parameters and if it will be able to work with unknown data and give correct forecasting results. And of course, how successful your model was learned. So, selecting this option will give you forecasting signals for last n days which were reserved as 'unknown' by the system and which can be used for estimating 'real' model's performance and accuracy.

If you selected to use out-of-sample data (backtesting period), there will be an option available to predict using the Out-Of-Sample period. This will give you the ability to generate predictions on the data which was not used in learning and estimate a

model's quality of generated signals.



The given results appear as BUY/SELL signals for the chosen period. The signals appear on the Chart page. The signal under the last day's data is the presumed market movement.

RED and GREEN signals are active signals which gives you the recommendation to enter the market or exit. Grayed arrows are inactive signals, generated by neural network, but, they are used for supporting previous active signals. So, when the first GREEN arrow appears at the chart, you get the recommendation to go LONG and to BUY some shares tomorrow at open (you also can see this recommendation at CURRENT POSITION INFO tab). You buy some shares then, update the model and get new forecasting. But, a new bar appeared after updating, there's a BUY signal again, which is supporting signal telling that you're on right side and as you're in LONG position already, you do nothing tomorrow at open and that BUY signal is grayed as inactive. As soon as the first SELL signal appears after several BUY signals, you get active RED arrow and SELL recommendation at CURRENT POSITION INFO tab. This is your recommendation to close previously opened LONG position and sell the shares tomorrow at open. After the shares are sold, you wait for the next active signal, red or green, for going SHORT or LONG.

If there's a SELL signal, it means that the market moves down. If BUY, the market moves up. If there's no signal, it means that the system gives a HOLD signal. Taking into consideration your model's training level and out-of-sample testing results, you can make decisions about your further market activity. You also can use embedded technical analysis module for plotting indicators, which may be used for supporting your decision to go with given recommendations.

Technical Analysis Module In Stock Assault 2.0 SideKick Edition

Here you can find basic definitions of technical indicators which are used in Stock Assault 2.0 SideKick Edition. This TA module does not take effect on forecasting core, but, included for making your own analysis and for supporting your decision of following BUY/SELL signal generated by the Neural Network.

MACD

MACD, which stands for Moving Average Convergence / Divergence, is a technical analysis indicator created by Gerard Appel in the 1960s. It shows the difference between a fast and slow exponential moving average (EMA) of closing prices. Fast means a short-period average and slow means a long period one, the standard periods are 12 and 26 days.

A signal or trigger line is then formed by smoothing this with a further EMA. The standard period for this is 9 days.

The difference between the MACD and the signal line is often calculated and shown not as a line but a solid block histogram style. This construction was made by Thomas Aspray in 1986. The calculation is simply histogram = MACD - signal.

The set of periods for the averages, often written as say 12,26,9, can be varied. Appel and others have experimented with various combinations.

Elder Ray Bear Power

A technical indicator developed by Alexander Elder that measures the amount of buying and selling pressure in the market. This indicator consists of two separate indicators known as "bull power" and "bear power". These figures allow a trader to determine the position of the price relative to a certain exponential moving average (EMA).

Linear Regression Oscillator

Linear regression is a statistical tool used to predict future values from past values. In the case of security prices, it is commonly used to determine when prices are overextended.

A Linear Regression trendline uses the least squares method to plot a straight line through prices so as to minimize the distances between the prices and the resulting trendline.

Linear regression is widely used in biological and behavioural sciences to describe relationships between variables. It ranks as one of the most important tools used in these disciplines. For example, early evidence relating cigarette smoking to mortality came from studies employing regression. Researchers usually include several variables in their regression analysis in an effort to remove factors that might produce spurious correlations. For the cigarette smoking example, researchers might include socio-economic status in addition to smoking to ensure that any observed effect of smoking on mortality is not due to some effect of education or income. However, it is never possible to include all possible confounding variables in a study employing regression. For the smoking example, a hypothetical gene might increase mortality and also cause people to smoke more. For this reason, randomized experiments are considered to be more trustworthy than a regression analysis.

MAVEs

A moving average, in finance and especially in technical analysis, is one of a family of similar statistical

techniques used to analyze time series data.

A moving average series can be calculated for any time series, but is most often applied to stock prices, returns or trading volumes. Moving averages are used to smooth out short-term fluctuations, thus highlighting longer-term trends or cycles. The threshold between short-term and long-term depends on the application and the parameters of the moving average will be set accordingly.

Mathematically, each of these moving averages is an example of a convolution. These averages are also similar to the low-pass filters used in signal processing.

A simple moving average (SMA) is the unweighted mean of the previous n data points. For example, a 10-day simple moving average of closing price is the mean of the previous 10 days closing prices.

In technical analysis there are various popular values for n , like 10 days, 40 days or 200 days. The period selected depends on the kind of movement one is concentrating on, such as short, intermediate or long term. In any case moving average levels are interpreted as support in a rising market or resistance in a falling market.

In all cases a moving average lags behind the latest price action, simply from the nature of its smoothing. An SMA can lag to an undesirable extent and can be influenced too much by old prices dropping out of the average. This is addressed by giving extra weight to recent prices, as in the WMA and EMA below.

One characteristic of the SMA is that if the data has a periodic fluctuation, then applying an SMA of that period will eliminate that variation (the average always containing one complete cycle). But a perfectly regular cycle is rarely encountered in economics or finance.

An exponential moving average (EMA), sometimes also called an exponentially weighted moving average (EWMA), applies weighting factors which decrease exponentially. The weighting for each day decreases by a factor, or percentage, on the one before it.

Commodity Channel Index

The Commodity Channel Index (CCI) is an oscillator originally developed by Donald Lambert. He introduced this new oscillator in an article published in the October 1980 issue of Commodities magazine (now known as Futures magazine).

The CCI is calculated as the difference between the price of a commodity and its simple moving average, divided by the normal deviation of the price. The index is usually scaled by a factor of $1/0.015$ to provide more readable numbers. Therefore, it is essentially a MACD, normalized the deviation.

Since its introduction, the indicator has grown in popularity and is now a very common tool for traders to identify cyclical trends not only in commodities but also equities and currencies.

The CCI can be adjusted to the timeframe of the market traded on by changing the averaging period.

Like most oscillators in technical analysis, the CCI was developed to determine overbought and oversold levels. Although its value is based on closing prices, the indicator really measures the variation from the mean of the price over n periods and thus a better classification for it would be momentum oscillators.

The Commodity Channel Index is often used for detecting divergences from price trends as an overbought/oversold indicator and to draw patterns on it and trade those patterns. In this respect, it is similar to bollinger bands, but is presented as an indicator rather than as overbought/oversold levels.

The CCI typically oscillates above and below a zero line. Normal oscillations will occur within the range of +100 and -100. Readings above +100 imply an overbought condition, while readings below -100 imply an oversold condition. As with other overbought/oversold indicators, this means that there is a large probability that the price will correct to more representative levels.

Relative Strength Index

The Relative Strength Index (RSI) is a technical analysis oscillator showing price strength by comparing upward and downward close-to-close movements.

The RSI is popular because it is relatively easy to interpret. It was developed by J. Welles Wilder and published in *Commodities* magazine (now called *Futures* magazine) in June 1978, and in his *New Concepts in Technical Trading Systems* the same year.

Note that the term relative strength also refers to the strength of a security in relation to the overall market or to its sector. For instance XYZ might rise 2% when the rest of the market rises 1%. This is sometimes called relative strength comparative to avoid confusion. It's unrelated to the Relative Strength Index described here. Wilder recommended a smoothing period of $N=14$. This is by his reckoning of EMA smoothing (see the EMA article on that), ie. $\alpha=1/14$.

Wilder considered a security overbought if it reached the 70 level, meaning that the speculator should consider selling. Or conversely oversold at the 30 level. The principle is that when there's a high proportion of daily movement in one direction it suggests an extreme and prices are likely to reverse. Levels 80 and 20 are also used or may be varied according to market conditions (eg. a bull market may have an upward bias).

Large surges and drops in securities will affect RSI, but it could just be a false buy or sell. The RSI is best used as a complement with other technical analysis indicators.

Stochastic Oscillator

The stochastic oscillator is a technical analysis oscillator (or two oscillators) showing the latest closing price in relation to the trading range of the past n days. This concept is unrelated to a stochastic in mathematics or statistics.

These stochastics were described by George Lane, in the 1950s. It's unclear whether the idea was original to him, he may have learnt of it from a previous employer Ralph Dysant and indeed he in turn from earlier traders.

Two oscillator lines are calculated, called %K and %D, each ranging from 0 to 100. %K is the closing price within the past n days trading range, ranging from 0 when the latest close is a new n day low, up to 100 for a new n day high.

%D is a 3-day simple moving average of %K.

The usual " n " is 14 days, ie. a fortnight's worth of past data, but this can be varied. Levels near the extremes 100 and 0, for either %K or %D, indicate strength or weakness (respectively) with prices making or approaching new n day highs or lows.

Levels above 80 and below 20 can be interpreted as overbought or oversold, but not on their own, only with other factors. Lane recommended waiting for a return back through those thresholds, ie. when the oscillator goes above 80, wait for it to fall below 80 before selling; or vice versa on going below 20 wait for a rise back above 20 before buying; which in effect means waiting for a bit of a reversal. Or alternately levels 80 and 20 might be traded when some other technical indicator suggests a non-trending market.

%D acts as a trigger or signal line for %K. A buy signal is given when %K crosses up through %D or a sell signal when it crosses down through %D. Such crossovers can occur too often and to avoid repeated whipsaws one can wait for crossovers occurring together with an overbought/oversold pullback or only after a peak or trough in the %D line.

Some traders consider the basic %K and %D too volatile, giving too many signals and too many whipsaws. This is addressed by forming "slow" stochastics. %K values are first smoothed by a 3-day simple moving average and then the %D formed by a further 3-day SMA on that. This "slowed" %K is the same as the "fast" %D, but it's easiest just to think of the slow form as first inserting an extra smoothing.

%K is the same as Williams %R, though on a scale 0 to 100 instead of -100 to 0, but the terminology for the two are kept separate.

Bollinger Bands

Bollinger Bands is a technical analysis tool invented by John Bollinger in the 1980s. They evolved from the concept of trading bands and can be used to measure the relative highness or lowness of price.

Bollinger Bands consist of:

- A middle band being a N-period simple moving average
- An upper band at K times a N-period standard deviation above the middle band
- A lower band at K times a N-period standard deviation below the middle band

Typical values for N and K are 20 and 2, respectively.

The bands cannot, as some have supposed, be used to make reliable statements regarding what fraction of an equity's prices will lie within a certain distance of the mean value. This is because an individual equity's price does not obey known distribution functions (see stochastic process). For example, if the bands for plus or minus two standard deviations are computed, it is wrong to suppose that ~95% of an equity's closing prices will, on average, lie within the bollinger bands. That would require, among other things, that the prices be normally distributed, which they are generally not. It would further require that the true standard deviation be known. The standard deviation calculated as above, however, is only an uncertain estimate of the true standard deviation. Furthermore, it should be realized that the "standard deviations" of stock prices for finite time periods are not fixed parameters as required to apply classical statistical theory, but instead are variables in constant flux depending on price volatility. Nevertheless, the bands have proved useful in the technical analysis of stock prices. The bands give a reliable visual picture of a stock's price volatility. No particular significance, however, should be attached to a price touching the upper or lower band, as Bollinger himself has pointed out. These occurrences should be considered in relation to other factors before making investment decisions.

It is of interest to note that faulty interpretation of a price touching or breaching a band based on incorrect statistical assumptions has become so widespread that some traders now use these events alone as trading signals and by so doing may have unwittingly injected significance into these band-touching events that would otherwise be absent.

When the bands lie close together a period of low volatility in stock price is indicated. When they are far apart a period of high volatility in price is indicated. When the bands have only a slight slope and lie approximately parallel for an extended time the price of a stock will be found to oscillate up and down between the bands as though in a channel. When this behavior is found to regularly repeat in conjunction with a fairly steady broad market, a traders may, with some validity, use a touch or near touch of the upper or lower band as an indication that a stock's price is nearing the limit of its trading range and therefore a price reversal is probable.

Quadrant Lines

Quadrant Lines are a series of horizontal lines that divide the highest and lowest values (usually prices) into four equal sections.

Quadrant Lines are primarily intended to aid in the visual inspection of price movements. They help you see the highest, lowest, and average price during a specified period.

An interesting technique is to display a Linear Regression trendline and Quadrant Lines. This combination displays the highest, lowest and average price, as well as the average slope of the prices.

Williams %R

Williams %R, or just %R, is a technical analysis oscillator showing the current closing price in relation to the high and low of the past n days (for a given n). It was developed by trader and author Larry Williams and is normally used just in the stock market.

The oscillator is on a negative scale, from -100 (lowest) up to 0 (highest). Such a scale is a little unusual and is sometimes found altered (by adding 100), but needn't cause any confusion. A value of -100 is the close today at

the lowest low of the past n days and 0 is a close today at the highest high of the past n days.

Williams used a 10 trading day period and considered values below -80 as oversold and above -20 as overbought. But they were not to be traded directly, instead his rule to buy an oversold was %R reaches -100%.

Five trading days pass since -100% was last reached
%R rises above -95% or -85%.

or conversely to sell an overbought condition

%R reaches 0%.

Five trading days pass since 0% was last reached
%R falls below -5% or -15%.

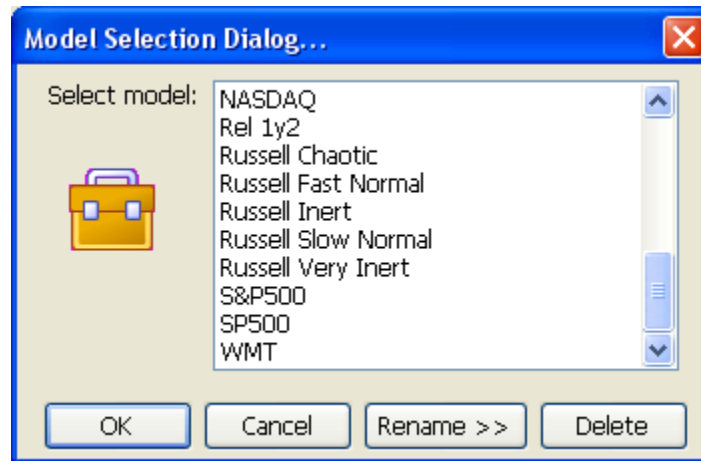
The timeframe can be changed for either more sensitive or smoother results. The more sensitive you make it, though, the more false signals you will get. The "close-position within a range" in the %R indicator is the same as the %K stochastic oscillator, on a different scale.

Saving Model

For saving your opened model, press the *SAVE* button or choose *File/Save Model* in the main menu. You can save your active model during the training process or in the working state. The Save procedure saves all the parameters of your model, its characteristics, forecasting results and so on. If option 'AutoSave on Exit' is turned on, your model will be automatically saved before the program closes.

Opening Already Saved Model

Click the OPEN button, and you will see the Model Selection Dialog. There you can rename the already saved models, delete them from your portfolio of the models or open the model in the system.

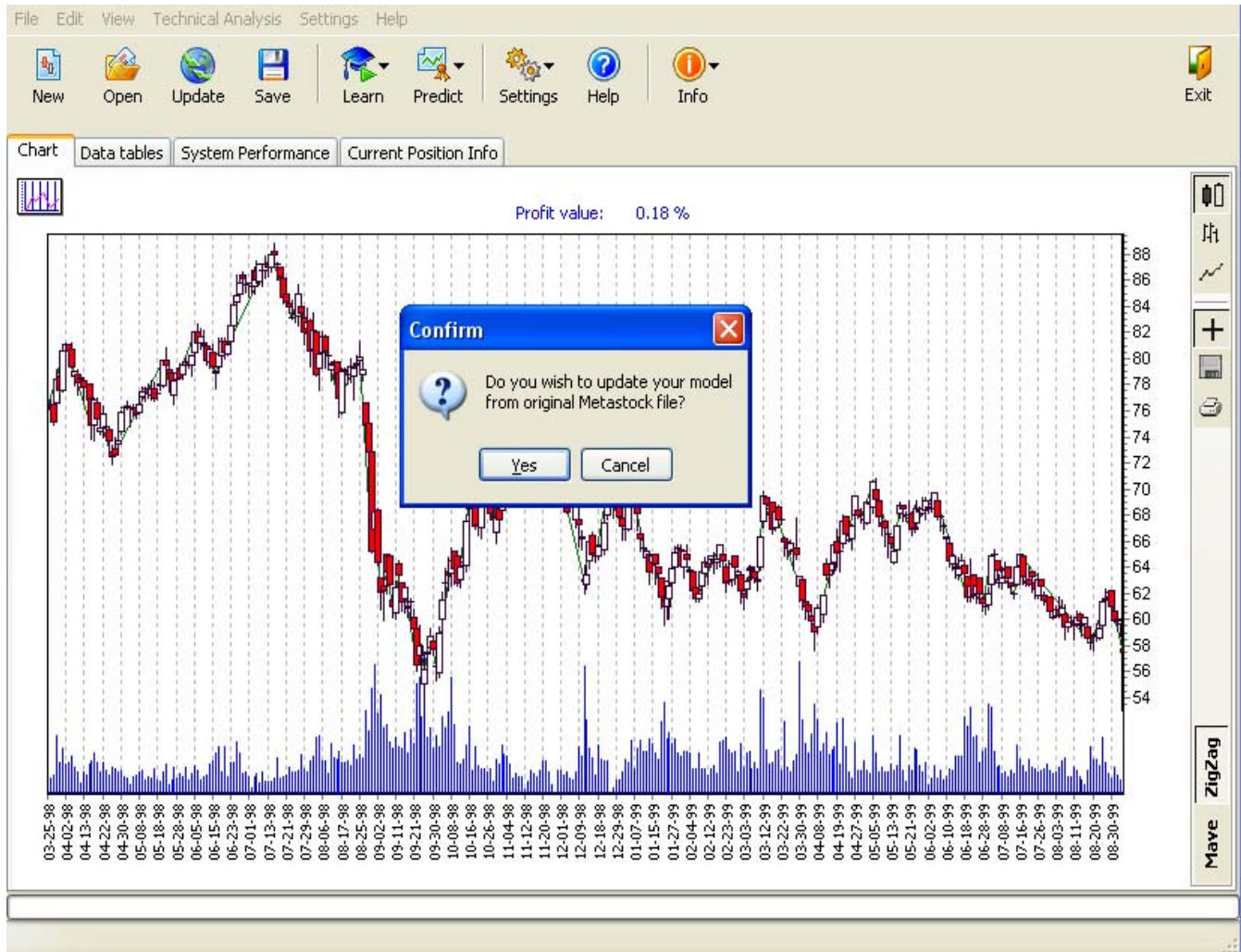


Updating Model

You can update your models daily, with a single click on the UPDATE button. Then the program connects to the server, downloads fresh data and adds this data into the current model. After updating, you can get new forecasting on the newly downloaded data. This will give you step-by-step testing of the already created model and estimate its accuracy and profitability in real time mode.

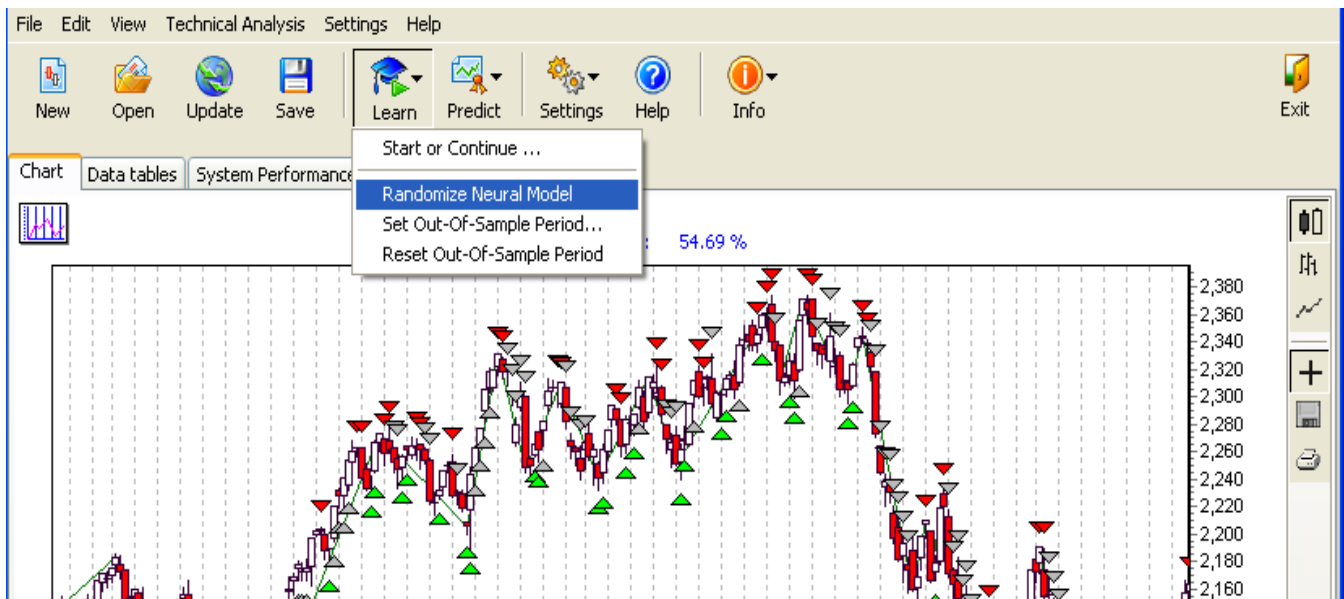
If you used Metastock database for creating the model, original data file will be used for updating the model.

In case if CSV file was used for building the model, file selection dialog will appear, asking for pointing the CSV file which should be used for extracting the data which does not present in the model.



Randomizing Model

If your model seems to be not that accurate, you can randomize it and relearn it from the beginning, using another learning setting. Also, you can set up out of sample period or reset it. If you choose RESET OUT OF SAMPLE PERIOD, this period is set to zero and all available data will be used in learning.



Step-By-Step Operating Recommendations

There you can find basic recommendations, how to set up your trading system for getting excellent forecasting results and build profitable models. Basic steps of operating is:

1. Creating model
2. Learning model
3. Estimating model's performance
4. Using model for getting forecasting signals on new data

If model's performance seems to be not too high for you, that means that you have to try to create another model, with different parameters set. The most important part is to define 'optimized' model for selected ticker symbol. For example, the model with No Filtration Level shows greater results than 2nd Filtration Level, or, Neural Network larger structure gives worse results than smaller structure because the data used in learning, is not sufficient for such a large NN structure.

The way how to get optimized model, is to create several models with different parameters for selected symbol, learn them all, and compare the performance of each model. The model with highest performance is your 'optimized' model, which may be used in further forecasting. For example, you can create 5 models for IBM, changing filtration level, or, trading strategy, or, desired signals frequency (see [Creating Model chapter](#) for detailed explanations of model's parameters), with different models' names, such as "IBM_INVEST_FILTR3_FREQ5", "IBM_MIDDLE_FILTRNO_FREQ3" etc. This way, you will have the description of created model in its name. You also always can see detailed model's parameters by clicking FILE/INFORMATION ABOUT ACTIVE NEURAL MODEL menu item, under main menu.

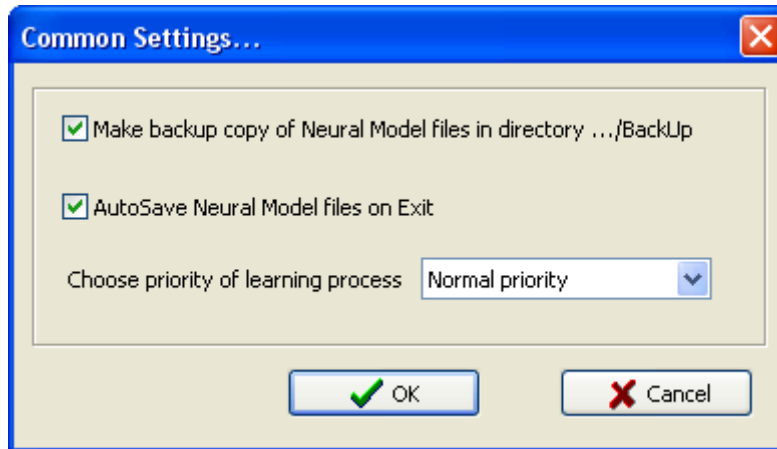
For estimating the performance of the model, we usually learn it and then get forecasting results on out-of-sample period. It is the period which was extracted from learning process, so the data for that period is unknown for the system. If the accuracy of signals is acceptable and profitability is high, we reset the out of sample period, randomize the model and relearn it from the beginning. Then, the model is ready for using.

Once the model is learned, it may be used for getting forecasting signals. Press UPDATE button daily and a new price bar will appear on the stock chart. Then, click MAKE PREDICTION ON ENTIRE DATASET menu and you will have a forecasting signal at the last added bar and at CURRENT POSITION INFO tab. There the recommendation will appear for the next day open. It may show HOLD for being out of trade if you are in opened position already, NONE for keeping track in the next several days if you're not in opened position yet, SELL for closing previously opened LONG position or for opening SHORT position tomorrow, BUY for opening LONG position at the next market open or for closing previously opened SHORT position.

One last point - you don't have to relearn your model after each update, as the structure of NN changes every time when learning process starts. So, we would recommend to use optimized model for the period of 15-20% of initial training period, then, estimate model's performance again, and if it is still up-to-date and can give adequate forecasting signals, continue using it. If the accuracy goes lower after some period of updating, just relearn it, so that the new data passed to the system by updating procedure, has reflection in the model. For example, if we create the model with 1 year historical period stock quotes, learn it, then, we may use it for 1-2 months, for getting forecasting and being in high accuracy level and total net profit. If the accuracy decreases after 2 months of using, we may click AFTERLEARN MODEL, so new data will be used in afterlearning. The model is not randomized in this case. Or, I may select RANDOMIZE MODEL, RESET OUT OF SAMPLE PERIOD and start total learning.

Common Settings

In settings menu, there're several options for changing program settings.



Backup copy of NN model

You can select this option and the program will save a backup copy of all model's files in the BACKUP directory. At any time, you can restore your model from this directory by copying the model into a working database directory.

Autosave NN model on exit

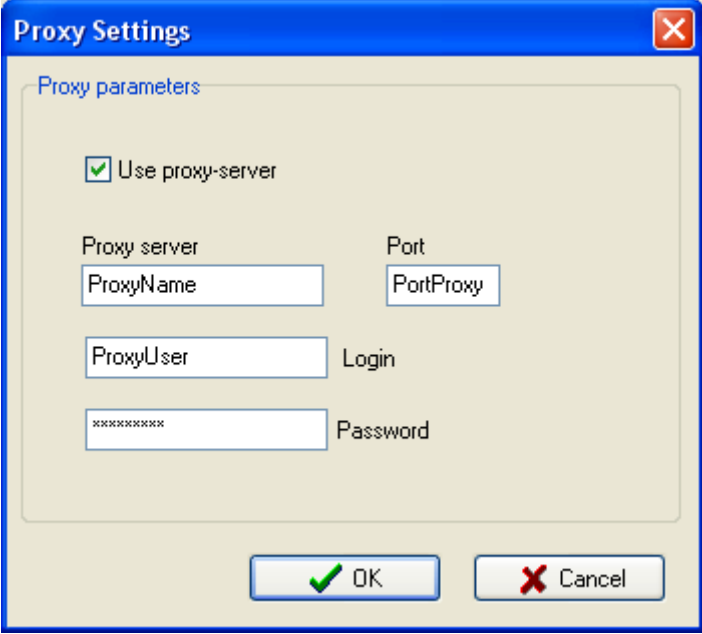
This option will save your opened model automatically on exit from the program.

Priority of learning process

Here you can select the system priority for the process during learning. This means that in high priority settings, your PC will give more CPU and memory resources to Stock Assault 2.0 SideKick Edition, for its calculations. Be careful of setting it to the highest. It may freeze your computer until it is finished processing. Other programs may become unresponsive. However, when it completes, your computer will return to normal.

Internet Connection Settings

If you use some kind of proxy server for connecting to the Internet, you may have to adjust these parameters to properly enable downloads of stock quote histories. If you have troubles connecting to the Internet from the program, please contact your system administrator to get the proxy name, proxy port, username and password, which should be used for accessing to the Internet.



The image shows a 'Proxy Settings' dialog box with a blue title bar and a close button in the top right corner. The main area is titled 'Proxy parameters' and contains a checked checkbox for 'Use proxy-server'. Below this are four input fields: 'ProxyName' (labeled 'Proxy server'), 'PortProxy' (labeled 'Port'), 'ProxyUser' (labeled 'Login'), and a masked password field (labeled 'Password'). At the bottom, there are two buttons: 'OK' with a green checkmark and 'Cancel' with a red X.

Field Label	Value
Use proxy-server	<input checked="" type="checkbox"/>
Proxy server	ProxyName
Port	PortProxy
Login	ProxyUser
Password	*****

Trading Strategy Settings

This option will give you the ability to select which strategy of trading is used for forecasting. If you wish to have only LONG positions or only SHORT positions, select the appropriate option. If it does not matter for you, select the LONG AND SHORT option. Then, all generated signals will be used for building predictions.

The screenshot shows a dialog box titled "Trading strategy". It has three sections:

- Entries available on...**: Three radio buttons are present. The first is "Only LONG positions", the second is "Only SHORT positions", and the third is "LONG and SHORT positions", which is selected.
- Shares Number Based On...**: Two radio buttons are present. The first is "Fixed Account Size", which is selected, and the second is "Fixed Number Of Shares".
- Shares Number Settings...**: A text input field contains the value "100000". To its right is a currency dropdown menu currently set to "USD". Below these fields is an "OK" button with a green checkmark icon.

There you're able to select the account properties, how the trades will be calculated and performance is reflected.

If you wish to use fixed account size, in US Dollars, select that option. Then, when the order is processed, the number of shares is calculated as $(\text{Account size}/\text{Order price})$.

For Fixed Number Of Shares account settings, you set up the number of shares for every order, so that the account size is not fixed.

Stop Loss Settings

This option will give you great ability to decrease the risk of losses. You can select between 3 different modes of stop loss alarm.

Plain loss - when the position is opened and the price falls (or rises, depending on LONG or SHORT position), when the difference between current price and opening position price is more than the selected percentage, the program gives you a STOP signal which closes an opened position.

Top price break - when the current is price less than (or more than, depending on SHORT or LONG current position which is opened) all low prices for the selected number of last bars, the position is closed by the stop loss module. This means, if you are in long position, but the price falls down as the minimum of last n bars (it is lowest price of the last N bars starting from the opening position bar), the position is closed by the stop loss module.

Position duration limit - if you wish to limit the maximum period of opened positions, use this stop rule. If you select 3 days, this means that if the position is opened, and you are 3 trading days beyond the date of opening the position, it is closed automatically. This allows you to avoid long-term opened positions. As the default, use 5 or 7 days.

Stop Loss Parameters...

Long Stops

Plain Loss

Top Price Break

Position Duration Limit

5 %

Short Stops

Plain Loss

Top Price Break

Position Duration Limit

17 %

Use STOP-LOSS Rule

OK Cancel