ANALYSIS SYSTEM FOR PREDICTING PERFORMANCE

Claim of Priority

[01] This patent application claims the priority and benefit of the following provisional patent applications fully incorporated herein by reference thereto as if repeated verbatim immediately hereinafter. Benefit of these filing dates is claimed with respect to all common subject matter:

Application No. 60/513,938, filed October 23, 2003, entitled "SYSTEM AND METHOD FOR PREDICTING STOCK PRICES"; and

Application No. 60/514,033 and filed October 23, 2003, entitled "INTELLIGENT AGENTS FOR PREDICTIVE MODELING".

Background Of The Invention

[02] This invention is related in general to digital processing and more specifically to predicting the behavior of a system by using the results of other analysis processes.

[03] Various analytical and predictive techniques have been devised for purposes of predicting. Some techniques may operate on simple concepts but may use variables or parameters that must be characterized or selected by a human user or operator in order to arrive at an analysis or prediction. For example, the common measure of a "moving average" of a stock's price is a simple calculation but the start and end of the time period used to calculate the moving average may vary.

[04] Although traditional techniques have proven to be useful for prediction and analysis of stock prices, as the number and complexity of techniques grows it is often difficult for a human user of the techniques to effectively use the techniques and to combine or correlate the various results provided by the techniques.

[05] An apparatus and method for a stock investment method with intelligent agents is described and illustrated. In one embodiment, the invention comprises a system-of-

systems for nonparametric, multifactor financial time-series modeling. The base system is not itself a model, but rather an environment for creating and dynamically managing a user's or other proprietary predictive model(s), which could be comprised of any number of user specified factors, indicators and trading systems (proprietary models) or other systems.

[06] Often different techniques may work well in some situations, or for a certain period of time, but will fail to provide desirable results in other situations or at other periods of time. For example, one technique may be to combine the performance of a group of stocks of companies in a common market, such as production of natural gas and oil. The general movement of the group of companies (i.e., up or down in price) can be used to predict the probable movement of another individual company in the same market. While this technique might work well where general market trends are the dominating factor in the market's price, the technique might not work well or even at all when a singular factor, such as an excessively high price to earnings ratio, can cause a company's current stock price to decline even when the general market trend for the that company's industry is increasing.

[07] Some techniques may operate on simple concepts but may use variables or parameters that must be characterized or selected by a human user or operator in order to arrive at an analysis or prediction. For example, the common measure of a "moving average" of a stock's price is a simple calculation but the start and end of the time period used to calculate the moving average can be infinitely variable. Other parameters can include the resolution of samples (e.g., per day, hour, minute, etc.), the manner of display of the results (e.g., table, plot, bar graph), and the type of data to which the technique is applied.

[08] Although traditional techniques have proven to be useful for prediction and analysis of systems such as financial markets, as the number and complexity of techniques grows it is often difficult for a human user of the techniques to effectively use the techniques and to combine or correlate the various results provided by the techniques.

Summary of Embodiments of the Invention

[09] Embodiments of the present invention are described in conjunction with systems, clients, servers, methods, and machine-readable media of varying scope. In addition to the aspects of the present invention described in this summary, further aspects of the invention will become apparent by reference to the drawings and by reading the detailed description that follows.

[10] In one embodiment, the invention is a stock predicting system that through experience learns to make money based on short-term stock predictions and due to inherent flexibility continues to be profitable in virtually all market environments.

[11] One embodiment of the invention provides a method for predicting a stock price comprising: Pre-processing stock data from a large set of mathematical indicators to produce indicator output signals; Entering the indicator output signals into a database; rocessing with advisors the indicator output signals to produce advisor output signals; Enter the advisor output signals into a database; and Inputting the advisor output signals into a neural network to produce a prediction of a stock price; Entering the neural network prediction into the database; and Iteratively updating the neural network weights for all stocks and system components upon receipt of new data.

[12] Another embodiment provides a method for predicting a stock price comprising: processing stock data from a set of mathematical indicators to produce indicator output signals; entering the indicator output signals into a database; processing with advisors the indicator output signals to produce advisor output signals; entering the advisor output signals into a database; and entering the advisor output signals into a neural network to produce a prediction of a stock price.

[13] In another embodiment the invention provides a method for predicting securities prices and other data types comprising: pre-processing securities price data and other data types using mathematical indicators to produce indicator output signals; entering the indicator output signals into a database; processing with advisors the indicator output signals to produce advisor output signals; entering the advisor output signals into a database; inputting the advisor output signals into a neural network to produce a prediction of a securities price or other data types; entering the neural network

prediction into the database; processing output signal data with overlay advisors to produce overlay advisor output signals; entering the overlay advisor output signals into a database; inputting the overlay advisor output signals and lower-level neural network output signals into a second high-level neural network to produce a final prediction of securities price or other data types; and iteratively updating the neural network weights for all securities and other data types and system components upon receipt of new data.

[14] In another embodiment the invention provides an apparatus and method for a stock investment method with intelligent agents is described and illustrated. In one embodiment, the invention comprises a system-of-systems for nonparametric, multifactor financial time-series modeling. The base system is not itself a model, but rather an environment for creating and dynamically managing a user's or other proprietary predictive model(s), which could be comprised of any number of user specified factors, indicators and trading systems (proprietary models) or other systems.

[15] In another embodiment the invention provides a method for predicting securities prices and other data types comprising: pre-processing securities price data and other data types using mathematical indicators to produce indicator output signals; entering the indicator output signals into a database; processing with advisors the indicator output signals to produce advisor output signals; entering the advisor output signals into a database; inputting the advisor output signals into a neural network to produce a prediction of a securities price or other data types; entering the neural network prediction into the database; processing output signal data with overlay advisors to produce overlay advisor output signals; entering the overlay advisor output signals into a database; inputting the overlay advisor output signals and lower-level neural network output signals into a second high-level neural network to produce a final prediction of securities price or other data types; and iteratively updating the neural network weights for all securities and other data types and system components upon receipt of new data.

[16] In another embodiment the invention provides a machine-readable medium having instructions for: pre-processing securities price data and other data types using mathematical indicators to produce indicator output signals; processing with advisors the indicator output signals to produce advisor output signals; inputting the advisor output signals into a neural network to produce a prediction of a securities price

or other data types; processing output signal data with overlay advisors to produce overlay advisor output signals; inputting the overlay advisor output signals and lower-level neural network output signals into a second high-level neural network to produce a final prediction of securities price or other data types; and iteratively updating the neural network weights for all securities and other data types and system components upon receipt of new data.

[17] In another embodiment the invention provides an apparatus for predicting securities prices and other data types comprising: means for processing securities price data and other data types using mathematical indicators to produce indicator output signals; means for processing with advisors the indicator output signals to produce advisor output signals; means for processing output signal data with overlay advisors to produce overlay advisor output signals; and means for iteratively updating neural network weights for all securities and other data types and system components upon receipt of new data.

[18] Another embodiment of the invention provides Graphical user interface (GUI) controls and displays to allow a user to configure different aspects of an analysis system. A financial analysis application allows a user to designate instruments whose data points, factors or other relevant data can be pre-processed with indicators. The outputs of the indicators are provided to advisors for calculation and grouping. A user is able to select the , types of raw data to use, the instruments, the factors, the data feeds and other sources for data, the indicators that perform processing and the advisors that perform analysis, grouping and outputting of results for combining using machine learning mechanisms. A user can set or adjust other parameters such as modeling frequency, scheduled processing, whether the system is running a simulation or is designed for higher performance execution. These and other characteristics of the system can be saved as a portfolio for later use and adjustment by the user. Other features are described.

[19] In one embodiment the invention provides a method for predicting the behavior of a system, the method using a processor coupled to a user input device and a display, the method comprising displaying a plurality of possible instruments; accepting a signal from a user input device to select one or more instruments as predictor

instruments, wherein the predictor instruments include factors; accepting a signal from a user input device to designate one or more possible indicators as active indicators; processing the output of one or more active indicators; and displaying a result of the processing.

[20] In another embodiment the invention provides A method for predicting the performance of a system, the method executing on a computer system, the computer system including a processor coupled to a display and coupled to a user input device, the method comprising: displaying a plurality of indicators, wherein an indicator accepts one or more data points and provides a result; accepting a signal from the user input device to select a plurality indicators; associating a plurality of the data points with the plurality of indicators; storing a state of a process including an indication of the selected plurality of indicators and the associated plurality of the data points; and restoring the state of the process including the selected plurality of indicators and the associated plurality of the data points.

[21] In another embodiment the invention provides a method for analyzing a financial instrument, the method comprising: identifying a plurality of instruments having data points; allowing user selection of a plurality of processes to process the data points and provide outputs; allowing user selection of a combining of the outputs; and adjusting performance of an analysis based on a comparison of an output with a past event.

[22] In another embodiment the invention provides a method for applying analysis techniques to a collection of data, the method executing on a computer system, the computer system including a processor coupled to a display screen, the method comprising: displaying a plurality of data sources on the display screen, wherein the data sources include data points; displaying a plurality of data types on the display screen; accepting a signal from a user input device to select a data type; identifying data points of the data sources having the selected data type; displaying a plurality of advisor processes on the display screen, wherein the advisor processes accept inputs and provide outputs; accepting a signal from a user input device to select one or more advisor processes; and applying the data points having the selected data type to the selected advisor processes to provide an input.

Brief Description of the Drawings

Figure 1 illustrates a screen display that allows a user to create and configure a portfolio;

Figure 2 illustrates a management interface screen;

Figure 3 shows an instrument selection and management screen;

Figure 4 shows how a user selects machine learning components;

Figure 5 illustrates an Indicator Library Creation and Selection screen;

Figure 6 illustrates an Indicator Association / Management screen;

Figure 7 illustrates an Advanced User Preferences interface screen;

Figure 8 illustrates a MACD Spectrum Indicator;

Figure 9 illustrates a One-to-One Configuration – Indicators screen;

Figure 10 illustrates a Strategy Description Language Scripting interface screen;

Figure 11 illustrates an Advisor Creation and Selection screen;

Figure 12 illustrates an Advisor Association / Management screen display;

Figure 13 illustrates a One-to-One Configuration - Advisors interface screen;

Figure 14 illustrates an Overlay Advisor Creation and Selection screen

Figure 15 illustrates an Overlay Advisor Association / Management screen display;

Figure 16 illustrates a One-to-One Configuration – Overlay Advisors interface screen;

Figure 17 shows combiner selection and management;

Figure 18 illustrates a Combiner Association / Management screen display;

Figure 19 illustrates a Portfolio Management interface screen;

Figure 20 illustrates a Back Test / Forward Test Creation and Management interface screen;

Figure 21A is a first part of a Portfolio View Output Screen;

Figure 21B is a second part of a Portfolio View Output Screen;

Figure 22A is a first part of an Instrument View/Output Screen;

Figure 22B is a second part of an Instrument View/Output Screen;

Figure 23 illustrates a basic data model;

Figure 24 is an illustration of a computer system;

Figure 25 illustrates subsystems that might typically be found in a computer system;

Figure 26 is a generalized diagram of a typical network;

Figure 27 illustrates relationships between an embodiment of an application and various other modules;

Figure 28 illustrates an embodiment of an application utilizing intelligent agents;

Figure 29 illustrates relationships between an embodiment of an application and various other modules; and

Figure 30 illustrates an embodiment of an application utilizing intelligent agents.

Detailed Description of Embodiments of the Invention

[23] Various embodiments of the invention are next discussed with reference to Figures 27 and 28.

[24] Embodiments of the present invention are described in conjunction with systems, clients, servers, methods, and machine-readable media of varying scope. In addition to the aspects of the present invention described in this summary, further aspects of the invention will become apparent by reference to the drawings and by reading the detailed description that follows.

[25] An apparatus and method for a stock investment method with intelligent agents is described and illustrated. In one embodiment, the invention is a stock predicting system that through experience learns to make money based on short-term stock predictions and due to inherent flexibility continues to be profitable in virtually all market environments.

[26] In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the invention.

[27] The reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in

connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment nor are separate alternative embodiments mutually exclusive of other embodiments.

[28] In the following detailed description of embodiments of the invention, reference is made to the accompanying drawings in which like references indicate similar elements, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical, functional, and other changes may be made without departing from the scope of the present invention. The flowing detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

[29] In one embodiment, the system is the implementation of a Technical Analysis approach to the stock market that is based on and exploits the following assumptions. Some of these assumptions are rather non-traditional and may even turn out to be false, but due to the flexibility of our overall architecture and interactions even bad choices can turn out to be good.

[30] Stock prices are not a "random-walk" and past price-volume trading behavior provides enough information (if processed carefully) for future price behavior to be predicted at a level of statistical and profitable significance.

[31] Given proper normalization and canonization of past data, all securities in all time frames exhibit behavior that is useful in helping to predict a future price move at a given time. For example, IBM's trading day tomorrow may resemble the MEX (Mexican) index 255 days ago, especially if a strong analogy can be established between their current and underlying technical environments. Despite these similarities, after normalization, each security or index may also exhibit characteristics and rhythms that are essentially their own "signature."

[32] A market predicting system must be complex enough to model a large gamut of technical trading strategies at varying time frames in order to simulate the habits of populations of traders that follow (or appear to) follow these strategies. Given a security,

certain predicting strategies will have proved to be more useful than others at predicting recent stock behavior. A stock predicting strategy can never be "very bad" since its very badness can be exploited by trading contrary to it. The only useless features and predictions are those that are essentially random. However, perversely, some "mal-features" may manage to change their success as soon as one tries to exploit them. Clearly, it is these mal-features that must be ignored or avoided or exploited when properly recognized.

[33] Combining these assumptions, a useful stock prediction can be developed as a function of:

[34] The past price behavior of the stock,

[35] Its past price behaviors, and the relationship to other securities in similar scenarios,

[36] The relative successes of various features (trading strategies) at predicting correctly or incorrectly recent price behavior (weighing these successes or failures by the amount of win or loss as described in detail later). These features may come from traditional technical analysis books, general and chaos theory time-series analysis, and other human or computer designed features and "expertise modules". As long as mal-features and over fitting can be avoided, adding new features to the system should improve performance in the long run once the system becomes adept at using these features,

[37] The rhythm of the successes and failures of individual features. Features themselves may be viewed as securities for which predictions (at a meta-level) become relevant,

[38] The Metropolis simulated annealing strategy of "heating up" (to encourage innovation) a system that is doing poorly and "cooling" a system that doing well is also used. Specifically, the system adjusts the learning rate to be higher (hotter) or lower (cooler) by decreasing or increasing the historical time period covered by output signals used by the system to make a final prediction. In one embodiment these adjustments are made in the final combining neural network so if the system is doing well it effectively considers larger advisor histories than it does when it is doing poorly. This added randomness should keep systems out of ruts created by any mal-feature behavior.

[39] In one embodiment, the system is 5000+ lines of Lisp coupled with a large historical database. It takes about 5-minutes on a Sun SPARC II to predict tomorrow's stock price for a given security.

[40] In one embodiment, the current securities and indices followed include: VOXX INSP TLAB MERQ CNXT NVDA AMCC VTSS CMVT NTPA MU ALTR PMCS ADI JNPR QLGC OSX DCLK ADCT WIND BKS ADBE EFII SEBL EMC SLR TJZ BBY SPLS SUNW WCOM QCOM APC LXK ALA CSCO GOX BBH MDT SGP VOD AMGN SWY HMA XOI MSFT AOL BGEN WLP BSYS CTL ONT TXCC REMD DIGL NTAP AMZN BVSN XLNX RNWK DELL PWER JDSU IDTI ATML NANO TLGD YHOO MOT COF ORCL IDPH BRCM NOK TXN XAU CHBS WMT XLE QQQ PAYX GE IBM TYC IXIC MEX OEX PFE DJI Indices followed include: OEX, COMPX, XOI, XAU, OSX AND MEX.

[41] In one embodiment, the system employs the following major advisors. The addition of each advisor contributed successfully to the system, so adding more in the future may be of additional benefit. Moreover, each advisor has an "anti" version which always bets contrary to it. For a given stock at a given time these advisors are deemed more or less relevant to future predictions. Details of rhythmic timing and advisor weighting mechanisms are not presently described, and these algorithms may affect success.

[42] Nearest Neighbor Advisor: Finds the historical precedent which best matches the current situation and reasons by analogy with the situation to make the prediction.

[43] Decision Tree Advisor: Develops a decision tree which explains 90 percent of past price movement as a function of the indicators listed below. Thus, the decision tree represents a "pattern that predicts the past." Given a security, the Decision Tree advisor uses the current decision tree to make its prediction for that security.

[44] Bob Advisor: a method of combining the indicators used in the system based on human intuition.

[45] Retracement Advisor: a day trading system based upon the system published by Joe DiNapoli in the book "Trading with Dinapoli levels."

[46] Complex Retracement Advisor: a system that combines a neural net with traditional Fibonacci retracement anaylsis

[47] For each security the system updates and stores the following features on a daily basis.

"Name of security:"
"Positive NN weight": Current weight of Nearest Neighbor advisor.
"Negative NN weight": Current weight of Anti Nearest Neighbor advisor.
"Positive DT weight": Current weight of Decision Tree Advisor.
"Negative DT weight": Current weight of anti-Decision Tree Advisor.
"Positive BOB weight: Current weight of Bob advisor.
"Negative BOB weight": Current weight of Anti-Bob advisor.
"Positive JOE weight": Current weight of Joe advisor.
"Negative JOE weight": Current weight of Anti-Joe advisor.
"Positive FIBO weight": Current weight of Fibonacci advisor
"Negative Fibo weight": Current weight of Anti-Fibonacci advisor.
"Alpha": A timing parameter
"Positive Trendpred Weight": Weight of midterm trend continuing.
"Negative Trendpred Weight": Weight of midterm trend discontinuing.
"Positive Shortpred Weight": Weight of short term trend continuing.
"Negative Shortpred Weight": Weight of short term trend discontinuing.
"Beta": A timing parameter
"Facilitation in Up trends": Ease of movement in up trends.
"Facilitation in Down trends": Ease of movement in down trends.
"Average Up Retracement": Average percent retracement after an uptrend.
"Average Down Retracement": Average percent retracement after a downtrend
"Beginning of current trend": how many days since current trend began.

[48]

"Total change in previous trend": how much has trend covered.
"RSI 8": 8 period stochastic
"3 fast rsi": 3 period fast stochastic
"3 slo rsi": 3 period slo stochastic.
"mavg8 ": 8 day average
"mavg17 " 17 day average
"mavgdiff9": 9 day moving average of 17-8.
"Previous stochastic reading ".
"Previous MACD reading:"
"Average advance 15 ema" average advance in last 15 days.
"Average decline 15 ema" average decline in last 15 days.
"Projected Drummond High" Drummond Geometry trend projection.
"Projected Drummond Low" " " " "
"Price Pulse High": Price pulse trend projection
"Price Pulse Low": Price pulse trend projection
"Fuel": measure of stock power
"Positive Reactivity": how performs after up day
"Negative Reactivity": how performs after down day
"3 day Pivot:" trading pivot 3 day avg.
"Pivot sum:" how far from pivot
"Pivot trend average: How far do we usually go on average.
"5 day avg. facilitation:"
"34 day avg. facilitation:"
"5 day avg. force"
"34 day avg force"

```
"34 day avg force"
"winning/losing streak:"  # of wining or losing days in a row.
"range streak:"  # days of increased range
"facilitation streak:" # of days of increased ease
"trend avg. prediction:" average trend length
"real prediction:"  our prediction
"last nearest neighbor prediction"
"last decision tree prediction"
"last bob-based prediction"
"last joe-based prediction"
"last fred-based prediction"
"last composite prediction:"  previous prediction
"13 day moving average:"
"public power:"   how stock performs during night time
"pro power:"      how stock performs during day time
"bear power:"     downward tendency
"bull power:"      upward tendency
"trend up/down:" direction of current short term trend
"key high:"  last important high
"key low:"   last important low
"current trend length:"  # of days
```

```
"current avg. trend length:"  # of days that is usual.
"trendrangetotal"; how much range in current trend.
"avg. pivas support" measure of support levels
"prev pivas support"  measure of support levels.
"current pivas support"
" 3 day avg. range"
" 34 day moving average"
" 5 day moving average"
" 34 day diff from avg"
"5 day diff from average"
" obos avg"    on balance stochastic
" 10day range average"
" facilitation"
" adaptive fair price:"
" adaptive momentum: "
"Fibonacci support levels:"
"Fibonacci resistance levels:"
" previous closes(most recent to least):"
"previous open:"
"previous high:"
"previous low:"
"previous close:"
" open:"
" high:"
" low:"
" close:"
" lohi:"
" change:"
```

Indicators: The following "low-level" indicators are used by the advisors in making daily predictions. They are composites of the features described above:

"dayofweek" we have discovered that the day of the week is an important trading feature!!
"breakdirection OEX" stocks are compared to the performance of the OEX
"Joe signal OEX"
"Uptrend facilitation OEX"
"Downtrend facilitation OEX"
"Uptrend retracement ratio "
"Downtrend retracement ratio"
"advance/decline ratio OEX"
"within Drummond Range OEX"

```
"within PricePulse Range OEX"
"fuel OEX"
"Positive Reactivity OEX"
"Negative Reactivity OEX"
"versus pivotpoint  OEX"
" Pivot trend clock OEX"
" Pivot trend clock - XAU"  compurison with to gold index
" Pivot trendclock  - XOI"   comparison with  oil index
"5dayfacilitation versus 34 day facilitation OEX"
"5day force versus 34 day force OEX"
"winning streak OEX"
"range increase streak OEX"
"facilitation increase streak OEX"
"average of trend length predictions OEX"
"aboveorbelow13dayavg OEX"
"public OEX"
"professionals OEX"
"bull ratio OEX"
"uporddown trend OEX"
"currenttrendversus avg OEX"
"vs averagepivas OEX"
" vs prepivas  OEX"
"withincurrentkeyrange OEX"
"relative rangesize  OEX"
"5versus34momentum   OEX"
"obos OEX"
"fairprice OEX"
"lohi  OEX"
"breakdirection"
"Joe signal"
"Uptrend facilitation"
"Downtrend facilitation"
"Uptrend retracement"
"Downtrend retracement"
"ax/axdx ratio"
"within Drummond Range"
"within PricePulse Range"
"fuel"
"Positive Reactivity"
"Negative Reactivity"
"versus pivotpoint"
" Pivot trend clock"
" 13 day relative strength in ranges vs. oex"
" 5dayfacilitation versus 34 day facilitation"
" 5day force versus 34 day force"
```

MODELING PROCESSES

[49] Figure 27 illustrates the primary components of the system in summary form. At 100, 102, 104, 106, and 108, the system's 5 advisors are shown, which are comprised of both machine learning components, common trading models in the field that are enhanced with embedded machine learning components, and non-learning proprietary (to the applicant) and common trading models. At 110, a group of proprietary and common indicators and compound indicators are shown. All of these components, described in more detail later, produce outputs which are then combined by a neural net combiner as shown at 112, producing a final prediction as shown at 114.

[50] In a preferred embodiment as illustrated by Figure 27, raw time-series stock data is entered into the process at 2, where all raw data is stored in a database as shown at 4. At 6, the first process step uses mathematical indicators to pre-process the raw time-series data. Each of the stocks for which the system is producing a prediction has a minimum indicator value which is equal to the change over the prior closing price for each respective stock. Additionally, each stock has its own value for each indicator it is pre-processed with.

[51] At 8 and 10, the raw time-series data values and the indicator output values are shown as being entered into the Data Base 1, at 12. Data Base 1 stores all raw time-series data and indicator output histories for further use in subsequent processes by more complex components called Advisors, as described in more detail later.

[52] Advisors comprise static or non-static mathematically based routines with embedded logic, which are generally more complex than the mathematical indicators used in the pre-processing of the raw time series database. In the context of embodiments of the present invention, static advisors do not have any learning function that causes changes in how the outputs are derived (i.e., they have fixed parameters), where non-static advisors have a degree of freedom generally governed by a learning mechanism and parameter ranges (e.g., as in a neural network). Different Advisors and combinations of advisors can have profound impact on the accuracy of predictions. Embodiments of the present invention employs specific implementations of machine learning components with unique proprietary enhancements described in more detail later.

[53] As shown at 14, the Nearest Neighbor Advisor comprises a component that creates a vector of the input values, and using table lookup finds the vector of values in previous periods of time that is most similar (based on a selected distance metric) and "assumes" what happened then will happen again; thus, its prediction can be said to be reasoned by analogy with past data or "case based" reasoning. Usually the more periods the nearest neighbor has to consider the more reliable it will be. Embodiments of the present invention uses normalized indicator values (e.g., using percentage moves rather than raw values, and standard deviations to normalize the size of moves) to allow case data on different stocks to be relevant candidates for the current query. For example, what IBM did on May 22, 1998 may be viewed as a relevant case for predicting the MEX (Mexican Stock Index) on June 11, 2004, if their normalized indicator value vectors are similar

[54] As shown at 16, a Decision Tree Advisor, is informally a conditional series of "tests" that are applied to the input, where depending on the outcome of the tests (a path through the tree), a prediction is made. Given n samples of prior instances of the classification path of the data as seen in the input history, the system uses a traditional "minimum entropy" heuristic that attempts to approximate the smallest "explanation" of the data over that period. For example, a small decision tree may, by way of example only, look like the following:

[55]

[56] If 13mvag is > close

[57] if 23ema is < high

[58] then expect 2.2% gain next period (5 samples)

[59] else expect 0.1% loss next period (2 samples)

[60] else

[61] if up 3 days in a row expect 4.5% drop next period (1 sample)

[62] else expect 0.5% gain (7 samples).

[63]

[64] Embodiments of the present invention comprises the use of decision trees in a manner to identify and then possibly "mimic" or "fade" what it expects other trading systems may have discovered about the current period. To mimic means take prediction

as is explained by decision tree as a final output, and to fade means to multiply its prediction by negative 1.

[65] Additionally, which tests are asked of the data depends on the outcome of their parent tests, thus producing a tree structure. Unlike conventional use of decision trees used in the art, which utilize just one back-tested static tree forward in time, embodiments of the present invention continually creates new decision trees for each new period (e.g., each day). Further, the decision trees operate on normalized data, such as like the data produced from the implementation of nearest neighbor, in order to allow rules to be learned across differing types of data, e.g., individual stocks and stock indices.

[66] As shown at 18, the Bob Advisor combines all of the indicator outputs to create an intermediate prediction for each respective stock. The Bob Advisor is an example of a static advisor because is not adaptive, treating every stock the same given a historical record of indicator values. Note it takes each stock's data and computes a "score" which is based on the applicant's personal heuristics. The score is initialized to 0 and then adjusted for each rule. For example, if the 5 Day Average Facilitation is greater than the 34 Day Average Facilitation the score is increased by 1. If the 18 Day Average is greater than the 40 Day Average the score is increased by 5.0 else decreased by 5.0 etc. Finally, the total score is normalized into a range of -3.0% and 3.0% which represents expected change in each respective stock's price as of the next day's Close.

[67] As shown at 20, an embodiment of the Retracement Advisor is based upon Joe DiNapoli's published trading methods which makes use of specific settings for MACD (moving average convergence divergence using several different moving averages each with different period parameters) stochastic indicators and delayed moving averages to generate a buy and sell signal of varying strength. The Retracement Advisor is designed to mimic the behavior of day traders who are following traditional stochastics and moving averages on their trading screen, thus exploiting any impact on stock price formation that results directly or indirectly from large populations of market participants using the same common trading signals. The unique implementation comprises the use of only a subset of the published method, excluding Fibonacci support and resistance levels. Rather, only the formulas (i.e., not chart patterns) are used, to produce a magnitude prediction rather than a trading signal, which is done by normalizing the

trading signals the formulas produce and rescaling the outputs to be within the range of typical market movements as measured by standard deviations.

[68] As shown at 22, the Complex Retracement Advisor comprises a machine learning mechanism embedded into a traditional Fibonacci retracement analysis system. The Complex Retracement Advisor foregoes the assumptions of stock price retracements (i.e., rebounds) based on Fibonacci ratios (e.g., 0.382 and 0.500 and 0.618 ratios) of the most recent trends, and learns non-linear effect of a stock's price reaching a support level (the price a stock trades at or near, but does not go lower than, over a certain period of time, e.g., the floor) and resistance level (the price a stock trades at or near, but does not go higher than, over a certain period of time, e.g., the ceiling). That is, rather than assuming the traditional ratios hold true, it learns what actually happens. Resistance and support price levels are defined as prices at which short-term trends changed. The Complex Retracement Advisor is a non-linear neural network (specifically, a multi-layer gradient descent with 100 non-linear interior nodes representing products of proprietary variables). It learns a non-linear combination of the 3 most recently identified support levels and the 3 most recently identified resistance levels and attempts to predict the next daily change in a stock's Closing price.

[69] At 14, 16, 18, 20, and 22, the Advisors that are shown further process the indicator output data stored in Data Base 1, producing output values that are representative of each Advisor's respective prediction for the next day's closing price. At 24, the outputs of all Advisors are entered into the second database called UPD, shown at 26. UPD Neural Net Combiner, shown at 34, is responsible for the next step in the prediction process. This Combiner is a neural net which reviews all of the new Advisor predictions for each stock's closing price, and then compares them to the actual closing prices stored in Data Base 1, updating the weights for each Advisor (each stock has negative and positive weights for each advisor), which weights are stored in a table in UPD as shown at 28. The weights represent what the Combiner has learned (i.e., its memory) about the accuracy of the Advisor predictions, where the final prediction for each respective stock is a learned linear combination of all advisor outputs for that stock. The Combiner comprises a traditional gradient descent neural network that attempts to learn a linear combination of its input weights to produce predictions that minimize their

error. In the context of embodiments of the present invention, the Combiner creates an output which is a linear combination of all Advisor predictions for each respective stock.

[70] Unlike most "weighted-expert" learning schemes, embodiments of the present invention are actually able and willing to assign negative weights to Advisors that are often wrong, thus, using their information as a contrarian would (i.e., learning how to exploit wrong predictions by doing the opposite). Other advances over the prior art include the fact that each instrument has its own neural net Combiner, which is itself evolving over time. In other words, the same exact predictions from the group of Advisors may not be interpreted the same way as an identical previous instance, even for the same stock. In general the system views Advisors as having cyclical tendencies not unlike stocks themselves, so that as an Advisor gets "hot" or "cold" or "bottoms" or "tops" this can be learned and exploited using a unique implementation of simulated annealing, which is incorporated into the mathematical underpinnings of the weighting mechanism in the neural net Combiner.

[71] At 34, each new final prediction delivered to the user, with this new prediction being stored in UPD Prediction Output Histories table as shown at 30. This final prediction then forms a part of a historical record of final outputs and their accuracies that are also reviewed by the Combiner prior to each new prediction task, and given it's own weighting used in the Combiner process. At 36, the new predictions are fed back for use by particular advisors in the next iteration (this conceptual, in practice, the new predictions are simply stored in the appropriate database tables where they are accessed at during the next prediction task). For example, the Complex Retracement Advisor updates its multi-layer neural net weights using the new prediction values, and the Nearest Neighbor Advisor and Decision Tree advisors use prior predictions as part of the set of indicator values they review with the next prediction task.

Further Description of An Embodiment

[72] Thus practice of embodiments of the present invention performs one or more of the following features:

[73] i) The use of a large pool of indicators for pre-processing raw data, including those that may not be relevant or that don't work well on their own rather than a small subset of definitely relevant indicators selected by an expert;

[74] ii) The use of the change in a security's price over the prior period as a default minimum indicator output for use as an input to higher level advisors;

[75] iii) The use of higher level signal generating components called advisors or agents that then further process the indicator pre-processed data producing a second order signals that are then combined by a neural network which iteratively learns to use the output signals to make more accurate predictions;

[76] iv) The use of machine learning algorithms together with static algorithms to produce output signals that are inputs to a neural network;

[77] v) The use of non-neural network machine learning algorithms to produce output signals that become inputs to a neural network;

[78] vi) The use of nearest neighbor, decision tree and neural network algorithms together in a single automated system for predictive modeling;

[79] vii) The embedding of neural networks into common analysis systems used in the field (such as Fibonacci) to learn the non-linear effects of price behavior meeting the conditions the original unimproved analysis system is intended to identify.

[80] vii) The use of normalization of securities and market indices time-series data with disparate quotation bases for purposes of comparing price activity and drawing analogies.

[81] ix) The use of historical time-series data for a group of securities, neural network learned weighting of the predictive accuracy of technical indicators used to pre-process time-series data, and the identification of behavioral analogies within the group by nearest neighbor and decision tree algorithms as a method for predicting future behavior of securities in the group.

[82] x) The modeling and prediction of system features themselves (such as particular indicator outputs) that have been determined to be relevant for the current prediction task, and the use of these meta-level predicted outputs to giving deeper meaning to the features role in the current prediction task.

[83] xi) The use of Metropolis simulated annealing to "heat up" (encourage innovation) the system when it is performing poorly and "cooling" the system when it is doing well through implementation within the neural network algorithm weighting mechanism that produces the systems final predictive output.

[84] xii) No pre-biased conception of relationship of indicator to price data (e.g. traditional use of indicator may actually be opposite of the case). The ability to trade counter to the advice of an advisor, indicator, or system.

[85] xiii) Use of machine learning advisors dynamically changing over relatively short time frames as opposed to trading one static, learned, backtested system.

[86] xv) Ability to determine average trend length dynamically over time and usethis to adjust indicators that require one to specify a period of time. (e.g. system may have a moving average bases on "3 trend lengths."

[87] xvi) Specific groups of traders following certain indicators or systems are identified (recognized by abnormal short term results of such indicators or systems). Advisors recent histories are observed and their outputs are normalized based on recent periods (e.g., 50) based on the number of standard deviations from the mean. So that an advisor that is consistently predicting a stock price will move "up 3%" or "up 2.5%" switches to "up 2%" the system will actually treat this as a negative signal since the number of deviations from the 50-period mean is now negative. An assumption is that trading populations are becoming less correlated with bullish signals from this advisor.

[88] xvii) The ability to trade counter to the advice of an advisor, indicator, or system is a function of the learning mechanism and the allowance of negative weights plus the normalization procedure above.

[89]

[90] xviii) A mutual fund and stock "scoring" system based on human assessment of the individual value of a large set of indicators is used as an advisor. The human assessment might be "intuitive" but wrong, however the system adjusts for this (as with any advisor or indicator) when producing the final prediction.

[91] xix) The ability to determine average trend length dynamically over time and use this to adjust indicators that require one to specify a period of time. (e.g. system may have a moving average bases on "3 trend lengths". The system keeps track of the number

of periods between changes in short term trend indicator and then takes a 3-period EMA of these.xx) The use of Anti-Advisors in the neural network weighting mechanism, for example:with 4 advisors we have 10 weights:

A1+ A1-

A2+ A2-

A3+ A3-

A4+ A4-

[92] Bulls Bears

[93] If, for example, A1 predicts up 2 percent and A3 predicts up 1.3 percent and A2 predicts down 1.5 percent and A4 predicts down 0.7 percent.

[94] If the market actually goes up 1 percent:

[95] Then A3 would be weighted 1/0.3 (actually an ema6 of these over time)

[96] since its error is 0.3

[97] A2 would be weighted 1/2.5

[98] A1 would be weighted 1/1

[99] A4 would be weighted 1/1.7

[100]

[101] A1- would be viewed as having said down 2 percent (being the anti of A1)

[102] and hence would be weighted 1/3.0

[103] A2- would be weighted 1/0.5

[104] A3- would be weighted 1/2.3

[105] A4- would be weighted 1/0.5

[106] Note that A3+ and A4- get the strongest weights: A3 was accurate and the opposite of A2 was also accurate.

[107] Bulls would get any positive movement not explained by the advisors and their weights, and bears would get any negative movement not explained by the advisors and their weights.

[108] If, for example, the consensus prediction was 0.8 percent, then bears would get 0.2 (1-0.8) and bears 0.0.

[109] This is just one way of doing this (assigning reward and punishment) - there are

[110] The primary improvement over prior art is the conception of the A1- A2- A3- and A4- anti-advisors.

[111] The actual code weights are here:

```
(w1      (initial-range-10) :type float)   ; "positive NN weight"
(w2      (initial-range-10) :type float)   ; "negative NN weight"
(w3      (initial-range-10) :type float)   ; "positive DT weight"
(w4      (initial-range-10) :type float)   ; "negative DT weight"
(w5      (initial-range-10) :type float)   ; "positive BOB weight"
(w6      (initial-range-10) :type float)   ; "negative BOB weight"
(w7      (initial-range-10) :type float)   ; "positive JOE weight"
(w8      (initial-range-10) :type float)   ; "negative JOE weight"
(w9      (initial-range-10) :type float)   ; "positive FIBO weight"
(w10     (initial-range-10) :type float)   ; "negative Fibo weight"
```

[112] Some portions of the detailed description are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of operations leading to a desired result. The operations are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms numbers, or the like.

[113] It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussion, it is appreciated that throughout the description, discussions utilizing terms such as "processing" or "computing" or "calculating" or "determining" or "displaying" or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as

physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

Intelligent Agents

[114] Various embodiments of the invention are next discussed with reference to Figures 29 and 30.

An apparatus and method for intelligent agents for predictive modeling are described and illustrated. In one embodiment, the invention comprises a system-of-system for nonparametric, multifactor financial time-series modeling. The base system is not itself a model, but rather an environment for creating and dynamically managing a user's or other proprietary predictive model(s), which could be comprised of any number of user specified factors, indicators and trading systems (proprietary models) or other systems.

In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the invention.

The reference in the specification to "one embodiment" or "An embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment nor are separate or alternative embodiments mutually exclusive of other embodiments.

In the following detailed description of embodiments of the invention, reference is made to the accompanying drawings in which like references indicate similar elements, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical, functional, and

other changes may be made without departing from the scope of the present invention. The flowing detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

MODULAR FRAMEWORK:

[115] The applicant's system has four base insertion points for user specified theory and strategy as illustrated by Figure 1. This represents a substantial improvement over both prior art, and the applicants prior patent application, introducing the following benefits: (a) the entire program has been modularized so that instead of being a closed system where the applicant has hard coded all strategy elements into an executable piece of software, the user can now select or create and insert any strategy elements including factors, indicators, advisors, and new overlay advisors, eliminating any strategy and market bias of the applicant (b) a third, "higher level" of signal generating strategy elements is introduced into an additional processing layer with a second neural net combiner which has contributed to prediction accuracy, (c) a method for extracting the decision path for each prediction task and ranking all strategy elements by their relative influence on the current prediction has been devised.

[116] At 100 of Figure 1, Insertion Point 1 is shown, where the instruments the user wishes to model and the factors the user wishes to include in the modeling processes can be inserted by the user. The factors selected for inclusion in a particular application would consist of financial instruments or other data types that the user has determined have a relationship directly or indirectly to the price action of the instruments the user wishes to model, examples of which are shown in Table I below. These relationships may be measured as either negative or positive correlations. The objective is to use the applicant's system to process the time-series data for any instrument that may itself serve as a leading or lagging indicator for the price movements of other instruments being modeled. Other valid relationships or dependencies, including those that are non-linear, will be detected and used by the system's learning mechanisms, contributing to the accuracy of each prediction task. Fundamental economic data, using both

estimated/anticipated statistics and actual releases may also be used as a factor in the system.

========== BEGIN TABLE I ==========

Potential Factors:

Sector and Key Mover Factors

IXCO NASDAQ Computer Index
IXF NASDAQ Financial 100 Index
MOX Morgan Stanley Internet Index
MSH Morgan Stanley Technology Index
PSE DJ PSE High Technology Index
NDX NASDAQ 100 Index
OEX S&P 100 Index

Volatility and Trend Factors

VIX CBOE Volatility Index
TRIN NYSE Short Term Trading Index
TRIT NASDAQ Stocks Short Term Trend Index

Interest Rates, Exchange Rates and Commodities Factors

TNX Ten Year Treasury Note Index
TYX Thirty Year Treasury Bond Index
VIX CBOE Market Volatility Index
OIX CBOE Oil Index
GOX CBOE Gold Index
OSX Oil Service Sector Index
XAU Phil American Gold & Silver Index
XEU Euro Index

Macroeconomic Announcement Factors

Change in non-farm payrolls
Unemployment rate
Employment cost index
Durable goods orders
NAPM manufacturing
NAPM non-manufacturing
Advanced retail sales
Industrial production
Consumer price index

========== ENDTABLE I ==========

[117] At 102, Insertion Point 2 is shown, where a user can insert the mathematical indicators of their choice, examples of which are shown in Table II below. The system uses these indicators to pre-process the raw time series data, producing output signals that can be used in subsequent modeling processes. For a typical user-specific application, the user's proprietary indicators and any other user specified indicators commonly used in the art would be inserted into the system. For a retail application, the licensee might select every known common indicator applicable to the asset class they wish to permit their users to trade, and seed the system with all of them. The guiding principle is that there is no limit to the number of indicators the licensee may choose to include. Irrelevant indicators will be "ignored" by the system, and consistently bad indicators can even be used by the applicant system to formulate contrarian views. Any proprietary indicator that can be mathematically defined can be coded and added to the Indicator Module, and new indicators can be added on the fly without disrupting the system.

========== BEGIN TABLE II ==========

Optional Indicators:

Facilitation in Uptrends: total change/ total range (EMA: 4 trends)

Facilitation in Downtrends: total change/total range (EMA: 4 trends)

Average Up Retracement: total change/total range divided by same for previous trend (EMA: 4 trends)

Average Down Retracement: total change/total range divided by same for previous trend (EMA: 4 trends)

8 Day Day Fast RSI: 3 day fast stochastic (DiNapoli) (EMA: 3 periods)

3 Day Slow RSI: 3 day slow stochastic (DiNapoli) (EMA: 3 periods)

MA 8: average close (EMA: 8 periods)

MA 17: average close (EMA: 17 periods)

MA Difference 9: average of MA17 - MA8 (EMA: 9 periods)

Average Advance 15 EMA: average up move (EMA: 15 previous up moves)

Average Decline 15 EMA: average down moves (EMA: 15 previous down moves)

Positive Reactivity: change/range after up move (EMA: 3 up moves)

Negative Reactivity: change/range after down move (EMA: 3 down moves)

3 Day Pivot: hi+lo+close/3 (EMA: 3)

5 Day Average Facilitation: change/range (EMA: 5)

34 Day Average Facilitation: change/range (EMA: 34)

5 Day Average Force: change * range (EMA: 5)

34 day Average Force: change * range (EMA: 34)

Winning/Losing Streak: consecutive up periods, if positive; consecutive down periods, if negative

Positive Range Streak: consecutive periods range beats 3 period average range

Negative Range Streak: consecutive periods range smaller than 3 period average range

Positive Facilitation Streak: # of periods of increasing change/range

Negative Facilitation Streak: # of periods of decreasing change/range

13 day Moving Average: average close (EMA: 13)

Public Power: open - previous close (EMA: ATL)

Pro Power: close - open (EMA: ATL)

Bear Power: close - low (EMA: ATL)

Bull Power: high - close (EMA: ATL)

Trend Up/Down: proprietary short term trend indicator (1 or -1)

Current Average TrendMagnitude: current trend length total change (EMA: 4 trends)

3 period Average Range: average range (EMA: 3 periods)

34 Day Moving Average: average close (EMA: 34 periods)

5 Day Moving Average: average close (EMA: 5 periods)

34 Day Difference from Average: 5 day moving average -34 day moving average (EMA: 34 periods)

5 Day Difference from Average: 5 day moving average - 34 day moving average (EMA: 5 periods)

10 Day Range Average: average range (EMA: 10 periods)

Adaptive Fair Price: proprietary estimate of "correct" close based on rating high range periods higher

Pivot Trend Clock: total distance price has stayed on one side of 3 day pivot (EMA: 3)

Within Drummond Range: (close - Drummond Low) / (Drummond Hi - Drummond Low) (EMA: ATL)

Within Current Key Range: proprietary version of Drummond Range using Key High and Key Low from proprietary short-term trend indicator

Within Price Pulse Range: same as Current Key Range but using Price Pulse High [+2.618 * (-Key High * close)] and Price Pulse Low [+2.618 * (-Key Low * close)] (EMA: 3)

Pivot Tension: distance from 3 day pivot / average distance from 3 day pivot

Momentum Speed: indicator from Bill Williams "New Trading Dimensions" (close - close 5 periods ago) / (close - close 34 periods ago)

Relative Range Size: 3 day range average / 10 day range average

Break Direction Indicator: scoring system that gives next day's trade direction [Let Prevpiv = previous pivot; Let Buybreak = 2Prevpiv - previous low; Let Sellbreak = 2Prevpi - previous high] If close above buybreak or if low above sellbreak get 2 points or else get 1 point; If close below sellbreak if high above buybreak lose 2 points or else lose 1 point; If high above buybreak lose 0.5 points if low above sellbreak lose -0.5 points

Fuel: proprietary ATL day EMA of sum of number of elements in {previous open, previous close previous high, previous low} that open, high, low, and close beat; high fuel (above 8) is usually considered bullish.

Note: References to "ATL" mean average trend length, which is a dynamically adjusted by noting the number of periods between sign changes in selected Short term Trends ("STT" as described later is a system default) and taking a 3 period exponentially weighted moving average (EMA) of these trend lengths.

========== END TABLE II ==========

[118] A second important feature of the applicant's system, and improvement over prior work in the art, is the use of Spectrum Indicators, which take the guesswork out of identifying the most relevant indicator parameters. For example, if a user believes the moving average is a valuable indicator for their strategy, but they have determined that the optimum number of days used in the calculation varies depending on the current market environment (e.g., high volatility or low volatility), they can use a Spectrum Indicator which permits the user to specify the range of days they would like to consider (e.g., 10 day - 50 day). The system will then process the time-series data using each of these variations on an ongoing, dynamic basis, and the learning mechanisms of the system will utilize the optimum output based upon what is working best in the current market environment.

[119] At 104, Insertion Point 3 is shown, where a user can insert their own trading or other models as Advisors, examples of which are shown in Table III below. Both the applicant's Base Advisors (recommended default set), described in more detail later, and

user specified Advisors are placed mid-level in the system where they produce output signals that are used in subsequent modeling processes. Any type of trading or other model a user can mathematically defined for purposes of coding can be inserted into the system's Advisor Module. Generally, Advisors are more complex than the mathematical indicators. These models can include "static" advisors, which contain fixed parameters, as well as "non-static" advisors such as a user's proprietary neural network or genetic algorithm based systems.

[120] At 106, Insertion Point 4 is shown, where a user can insert their most complex or possibly their most important trading and other models, as Overlay Advisors, where their output signals will be used in the final step of the modeling processes, discussed in more detail later. The user can insert any proprietary or non-proprietary, static or non-static trading or other models, which must be coded in order to be added to the Overlay Advisor Module. The applicant has coded five proprietary overlay advisors, which the user may also chose from, examples of which are shown in Table III below.

========== BEGIN TABLE III ==========

OptionalAdvisors

Joe DiNapoli Advisor: Fibonacci based day trading system as outlined by Joe DiNapoli

Equity Trading Advisor: equity day trading system using all current coded indicators with proprietary scoring system

Mutual Fund Trading Advisor: proprietary mutual fund day trading system

Optional Overlay Advisors

Surprise Overlay Advisor: evaluates difference between actual close and predicted close; close - predicted close (EMA: 1)

Momentum Overlay Advisor: reviews total change in last ATL periods; close - close ATL periods previous

Pattern Analysis Prediction Overlay Advisor: reviews signals from mid-level pattern analysis advisors to approximate the populations of traders correlated with following them or "fading" them (leaving off following them)

Buying Pressure Overlay Advisor: proprietary spectrum indicator that adjusts for trending versus chopping market movements

Pivot Point Overlay Advisor: proprietary day trading system related to distance from 3

period pivot points

Balance Overlay Advisor: estimated bulls - estimated bears as determined from review of pattern analysis routines

========== END TABLE III ==========

At 108, the complete set of machine learning and database components are represented for purposes of illustration, and at 110, the final prediction is represented

MODELING PROCESSES:

[121] In a preferred embodiment as illustrated by Figure 2, raw time-series stock data is entered into the process at 2, where all raw data is stored in a database as shown at 4. At 6, the first process step uses mathematical indicators that are commonly used in the art to pre-process the raw time-series data. Each of the stocks for which the system is producing a prediction has a minimum indicator value which is equal to the change over the prior Closing price for each respective stock. Additionally, each stock has its own value for each indicator it is pre-processed with.

[122] At 8 and 10, the raw time-series data values and the indicator output values are shown as being entered into the Data Base 1, at 12. Data Base 1 stores all raw time-series data and indicator output histories for further use in subsequent processes by more complex components called Advisors, as described in more detail later.

[123] Advisors are static or non-static mathematically based routines with embedded logic, which are generally more complex than the mathematical indicators used in the pre-processing of the raw time series database. In the context of the applicant's system, static advisors do not have any learning function that causes changes in how the outputs are derived (i.e., they have fixed parameters), where non-static advisors have a degree of freedom generally governed by a learning mechanism and parameter ranges (e.g., as in a neural network). Different Advisors and combinations of advisors can have profound impact on the accuracy of predictions. The applicant uses specific implementations of machine learning components as well as specific common trading systems used in the art, with unique proprietary enhancements described in more detail later.

[124] As shown at 14, the Nearest Neighbor Advisor is, informally, a component that creates a vector of the input values, and using table lookup finds the vector of values in previous periods of time that is most similar (based on a selected distance metric) and

"assumes" what happened then will happen again, thus, its prediction can be said to be reasoned by analogy with past data or "case based" reasoning. Usually the more periods the nearest neighbor has to consider the more reliable it will be. Unlike prior work in the art using nearest neighbor techniques, the applicant's system uses normalized indicator values (e.g., using percentage moves rather than raw values, and standard deviations to normalize the size of moves) to allow case data on different stocks to be relevant candidates for the current query. For example, what IBM did on May 22, 1998 may be viewed as a relevant case for predicting the MEX (Mexican Stock Index) on June 11, 2004, if their normalized indicator value vectors are similar

[125] As shown at 16, a Decision Tree Advisor, is informally a conditional series of "tests" that are applied to the input, where depending on the outcome of the tests (a path through the tree), a prediction is made. Given *n* samples of prior instances of the classification path of the data as seen in the input history, the system uses a traditional "minimum entropy" heuristic that attempts to approximate the smallest "explanation" of the data over that period. For example, a small decision tree might look like the following:

```
if 13mvag is > close
    if 23ema is < high
        then expect 2.2% gain next period (5 samples)
          else expect 0.1% loss next period (2 samples)
        else
          if up 3 days in a row expect 4.5% drop next period (1 sample)
        else expect 0.5% gain (7 samples).
```

[126] The applicant's system also uses decision trees in a unique way to identify and then possibly "mimic" or "fade" what it expects other trading systems may have discovered about the current period. To mimic means to accept use prediction as is explained by the decision tree and to fade means to multiply the prediction by negative 1. This is accomplished implicitly by the weighting mechanism used in the neural network, which uses Anti-Advisors as described in more detail later. Additionally, which tests are asked of the data depends on the outcome of their parent tests, thus producing a tree structure. Unlike conventional use of decision trees used in the art, which trade just one back-tested static tree forward in time, the applicant's system continually creates new

decision trees for each new period (e.g., each day). Further, the decision trees operate on normalized data like the applicant's implementation of nearest neighbor, in order to allow rules to be learned across differing types of data, e.g., individual stocks and stock indices.

[127] As shown at 18, the Short Term Trend Advisor which is a very simple indicator that the applicant recommends, and specifies as a default advisor. This simple advisor does contain strategy bias, and for this reason its specifications are published and made completely transparent to the user. The user may modify this advisor's parameters or, use its basic concept to create a their own proprietary replacement. It is recommended not because it is a powerful component, but because it is useful when combined with other signals. Its specific formula/process is as follows:

Short Term Trend Indicator/Advisor (STT):

1. initialize STT to direction of last period change.
2. initialize hi to last period High
3. initialize lo to last period Low

Repeat for each period:

4. If the period Close > hi then STT = up
   hi replaced with period High
   lo replaced with period Low

   If the period Close < lo STT = down
   hi replaced with period High
   lo replaced with period Low

Using the above formula, the STT base advisor determines if the trend is up or down based on the current OHLC bar for a given frequency when its direction was changed (when its previous OHLC bar was violated). Average trend length can be calculated by counting the number of consecutive bars in a given direction.

[128] As shown at 20, the client can specify or insert any signal generating strategy element as an Advisor, which will then operate in parallel to the other advisors. These client specified or inserted advisors can be either static mathematical models, non-static machine learning based models or a hybrid. At 14, 16, 18, and 20, the Advisors that are shown further process the indicator output data stored in Data Base 1, producing output values that are representative of each Advisor's respective prediction for the next day's closing price. At 22, the outputs of all Advisors are entered into the second database called UPD, shown at 24. UPD Neural Net Combiner, shown at 30, is responsible for

the next step in the prediction process. This Combiner is a proprietary implementation of a neural net which reviews all of the new Advisor predictions for each stock's closing price, and then compares them to the actual closing prices stored in Data Base 1, updating the weights for each Advisor (each stock has negative and positive weights for each advisor), which weights are stored in a table in UPD as shown at 26. The weights represent what the Combiner has learned (i.e., its memory) about the accuracy of the Advisor predictions, where the final prediction for each respective stock is a learned linear combination of all advisor outputs for that stock. The type of neural network used is a form of Perceptron, which is, informally, a type of neural network that attempts to learn a linear combination of its input weights to produce predictions that minimize their error. In the context of the applicant's system, the Combiner creates an output which is a linear combination of all Advisor predictions for each respective stock. Perceptrons have some unique characteristics that make them suitable for determining market temperament: (a) since it iteratively adjusts itself is most influenced by the most recent data, and (b) it tends to over compensate if the recent data point is misevaluated, and these characteristics are not unlike human emotion - a major factor in financial markets. The Perceptrons also permit the use of anti-advisors and negative weights in the learning mechanism that permits the system to make predictions in a contrarian way.
Unlike most "weighted-expert" learning schemes, the applicant's system is actually able and willing to assign negative weights to Advisors that are often wrong, thus, using their information as a contrarian would (i.e., learning how to exploit wrong predictions by doing the opposite). Advisors recent histories are observed and their outputs are normalized based on recent periods (e.g., 50) based on the number of standard deviations from the mean. So that an advisor that is consistently predicting a security will go "up 3%" or "up 2.5%" switches to "up 2 percent," the system will actually treat this as a negative signal since the number of deviations from the 50-period mean is now negative. This identifies when trading populations are becoming less correlated with bullish signals from this advisor.

[129] The use of Anti-Advisors in the neural network weighting mechanism comprising: with 4 advisors we have 10 weights:

A1+ A1-

A2+ A2-
A3+ A3-
A4+ A4-
Bulls Bears

If, for example, A1 predicts up 2 percent and A3 predicts up 1.3 percent and A2 predicts down 1.5 percent and A4 predicts down 0.7 percent.

If the market actually goes up 1 percent, then:

A3 would be weighted 1/0.3 (actually an ema6 of these over time) since its error is 0.3

A2 would be weighted 1/2.5
A1 would be weighted 1/1
A4 would be weighted 1/1.7

A1- would be viewed as having said down 2 percent (being the anti of A1)
and hence would be weighted 1/3.0

A2- would be weighted 1/0.5
A3- would be weighted 1/2.3
A4- would be weighted 1/0.3

Note that A3+ and A4- get the strongest weights because A3 was accurate and the opposite of A2 was also accurate.

Bulls (positive change) would get any positive movement not explained by the advisors and their weights, and bears (negative change) would get any negative movement not explained by the advisors and their weights. If, for example, the consensus prediction was 0.8 percent, then bears would get 0.2 (1-0.8) and bears 0.0.

Example code weights are as follows:

```
(w1        (initial-range-10) :type float)   ; "positive NN weight"
(w2        (initial-range-10) :type float)   ; "negative NN weight"
(w3        (initial-range-10) :type float)   ; "positive DT weight"
(w4        (initial-range-10) :type float)   ; "negative DT weight"
Etc...
```

[130] Other advances over the prior art include the fact that each instrument has its own neural net Combiner, which is itself evolving over time. In other words, the same exact predictions from the group of Advisors may not be interpreted the same way as an identical previous instance, even for the same stock. In general the system views

Advisors as having cyclical tendencies not unlike stocks themselves, so that as an Advisor gets "hot" or "cold" or "bottoms" or "tops" this can be learned and exploited using a unique implementation of simulated annealing, which is incorporated into the mathematical underpinnings of the weighting mechanism in the Perceptron Combiner. Specifically, the system adjusts the learning rate to be higher (hotter) or lower (cooler) by decreasing or increasing the historical time period covered by output signals used by the system to make a final prediction.

[131] At 32, each new final prediction is delivered to the user, with this new prediction being stored in UPD Prediction Output Histories table as shown at 28. This final prediction then forms a part of a historical record of final outputs and their accuracies that are also reviewed by the Combiner prior to each new prediction task, and given it's own weighting used in the Combiner process. At 34, the new predictions are fed back for use by particular advisors in the next iteration (this is conceptual, in practice, the new predictions are simply stored in the appropriate database tables where they are accessed during the next prediction task). For example, the Fibonacci Advisor updates its multi-layer perceptron weights using the new prediction values, and the Nearest Neighbor Advisor and Decision Tree advisors use prior predictions as part of the set of indicator values they review with the next prediction task.

[132] At 36, there can be any number of optional, user selected Overlay Advisors, which can be proprietary or non-proprietary, static or non-static trading or other models that produce an output signal. The applicant has developed his own proprietary models that are suitable for use as Overlay Advisors, which the user can select from. Overlay Advisors operate on the signal outputs from the Advisors. The use of a third level of signal generating components operating on the signals from the prior layer is an improvement over prior art in the field, where the number of processing layers has been limited due to inadequate combining methods. In addition, previous systems usually generate their rules from look-back-time-frames (i.e., number of periods used in the modeling process), which are very large (and hence represent statistical averages), as opposed to adaptation to various learned, shorter time frames that are often exploited by the applicants' system. Shorter time frames can produce more obvious signals and can often reflect actual perspectives taken by large populations of traders and systems. The

applicants' use of multiple-layers allow for more precise processing of these short-term effects.

[133] At 38, a Neural Net Combiner that is the same as the Combiner used at 30, is used to combine the output from the Overlay Advisors to produce a final prediction for each instrument, as shown at 40.

[134] The following Example is an actual example of the applicants' system running on 10 days of data for the stock IBM. Some of the highlights have been annotated in italics.

EXAMPLE

BEGIN EXAMPLE:

10 days of IBM: predictions are made of days 2 through 10.

IBM 12182003 93 93.38 92.5 92.73 6873600
IBM 12192003 93.23 93.25 92.67 93.14 7090700
IBM 12222003 92.83 93.5 92.78 93.39 4332300
IBM 12232003 93.07 93.44 92.34 92.79 3677200
IBM 12242003 92.8 92.8 92.18 92.27 1760500
IBM 12262003 92.37 93 92.3 92.9 1408500
IBM 12292003 93.1 93.73 93.03 93.52 4034200
IBM 12302003 93.27 93.5 92.36 92.63 4003400
IBM 12312003 92.66 92.85 92.16 92.68 4726900
IBM 01022004 92.86 93.05 91.2 91.55 5331200

(SETF PROFITS '0.021433705) *total profits (9 days)*
(SETF XPROFITS '0.017256744) *total profits confid >4*
(SETF XXPROFITS '0) *total profits confid > 7*
(SETF BHPROFITS '-0.012590914) *Buy and Hold p/l*

(SETF WINS '5)
(SETF LOSSES '4)
(SETF PERCENTAGE '0.5555556) *winning pct*

(SETF XWINS '2)
(SETF XLOSSES '1)
(SETF XPERCENTAGE '0.6666667) *winning pct confide > 4*

(SETF XXWINS '0)
(SETF XXLOSSES '0)
(SETF XXPERCENTAGE '0)

(SETF BHWINS '5)
(SETF BHLOSSES '4)
(SETF BHPERCENTAGE '0.5555556) *Buy Hold win pct*

(SETF ANNUALRETURN '53.584267) *annualized p/l*
(SETF XANNUALRETURN '143.8062) *annualized p/l confid > 4*
(SETF XXANNUALRETURN '0) *confid > 7*
(SETF BHANNUALRETURN '-31.477285) *annualized pl/ buy and hold*
(SETF AVERAGERROR '0.65668494) *average error in magnitude.*

Factors were not actually used in this example but here they are:

(SETF FACTORSLIST
'($NDX $SPX $IXCO $SOX $RUT $BKX $MOX $TRIN $MSH $OFIN $OSX $PSE $TRIT
DNA MRK C JPM HPQ SCH AA CAT XOM HD DD CY KLAC AMD))

record of trades: (sym date period predicted p/l)

(SETF PROFIT-VALUES
'((IBM 1022004 0 91.55 -0.003668537 0.012192461)
(IBM 12312003 0 92.68 -0.006153514 -5.398149e-4)
(IBM 12302003 0 92.63 -0.003742515 0.009516675)
(IBM 12292003 0 93.52 -0.0037674918 -0.00667379)
(IBM 12262003 0 92.9 -0.0033597054 -0.006827841)
(IBM 12242003 0 92.27 -0.005927363 0.005604098)
(IBM 12232003 0 92.79 -0.0038548023 0.0064246543)
(IBM 12222003 0 93.39 -0.0031135923 -0.0026841315)
(IBM 12192003 0 93.14 0.004744958 0.0044213957)
(IBM 12182003 0 92.73 10.0e-9 0.0)))

the theree trades with confid > 4

(SETF XPROFIT-VALUES
'((IBM 1022004 0 91.55 -0.003668537 0.012192461)
(IBM 12312003 0 92.68 -0.006153514 -5.398149e-4)
(IBM 12242003 0 92.27 -0.005927363 0.005604098)))

Advisor predictions over time:
date symbol prediction

advisors are 0) nearest neighbor
1) decision tree
2) bob advisor
3) joe advisor
4) short term trend
5) balance of power

for example: on 12262003, bob advisor predicted -0.06033659

```
(SETF ADVISOR-VALUES
     '((1022004 IBM -0.0012015013 -0.005604098 -0.078255095 -0.010922993
       0.9155 -0.27133426 0.0)
      (12312003 IBM 0.0041214316 -0.0053863567 -0.08539278 0.010789814
       0.92679995 -0.09445794 -0.012192461)
      (12302003 IBM 5.60451e-4 0.0051517896 -0.08582047 -0.010795639
       0.9262999 -0.048839614 5.398149e-4)
      (12292003 IBM 0.0024637857 -0.004009584 -0.077993184 0.0106929
       0.9351999 0.1367071 -0.009516675)
      (12262003 IBM -1.5590312e-4 0.0035527637 -0.06033659 -4.001082e-5
       0.929 -0.09166712 0.00667379)
      (12242003 IBM -9.846449e-4 -0.0032123271 -0.07132701 -0.010837759
       0.9226999 -0.22492443 0.006827841)
      (12232003 IBM 1.7021824e-4 0.002368509 -0.26854187 -0.010777024
       0.9279 -0.043795254 -0.005604098)
      (12222003 IBM 0.002368509 0.0022106979 0.09213007 -0.037477247
0.9339
       0.096638545 -0.0064246543)
      (12192003 IBM 0.0022106979 0.0 -0.07449984 0.010736526 0.9314
       -0.09569428 0.0026841315)
      (12182003 IBM 0.0 0.0 -0.078 0.010783996 0.92730004 -0.2424254
       0.0044213957)))
```

The 88 indicators:

```
(setf indicators (list
"surprise-oriented-adaptive-average" ;0
"breakdirection" ;1
"last change"   ;2
 "upward facilitation" ;3
 "downward facilitation" ;4
 "upward retracement"  ;5
 "downward retracement" ;6
 "ADX" ;7
```

```
"Drummund range size"  ;8
"Price Pulse range size"; 9
 "Volume increase streak"; 10
 " positive reactivity"; 11
 " negative reactivity"; 12
 " pivot momemtum"; 13
 " pivot clock"; 14
 " relative strength"; 15
 " 5facilitation/34facilitation" ;16
 " 5force / 34 force"; 17
 " winning streak"  ; 18
 " range streak"    ; 19
 " facilitation streak" ; 20
 " change improvement streak"; 21
 " distance from 13 mavg"; 22
  " public power" ; 23
  " pro power"    ; 24
  " bullspower"   ; 25
  " updown switch"; 26
   " trend clock" ; 27
   " pivot agony" ; 28
   " immed. pivot agony" ; 29
   " key range"; 30
   " range growth"; 31
   " 5/34  power"; 32
   " oscillator"; 33
   " adaptive average"; 34
   " surprise"       ; 35
   " low power"      ; 36
   " high power"     ; 37
    " uptrend progress" ; 38
    " downtrend progress" ; 39
    " 5/20 MACD" ; 40
    " 20/50 MACD"; 41
    " trendsize/float turnover" ; 42
    " Doji-ness"; 43
     " 3-day climax"        ; 44
     " 40-day climax"       ; 45
     " 3mavg 3 displaced"   ; 46
     " Fidelity indicator" ; 47
     " decision tree advisor" ; 48
     " nearest neighbor"; 49
     " Bob predictor"; 50
     " Joe predictor"; 51
     " short term trend pred" ; 52
     " weird trader's formula"       ; 53
```

```
      " buying pressure"    ; 54
      " balance of power"    ; 55
      " Momemtum"    ; 56
      " Price Pulse"    ; 57
      " Real Prediction"    ; 58
      " percent change"    ; 59
      " Range adx"    ; 60
      " Volume Adx"    ; 61
      " Distance from 50"    ; 62
       " Distance from 20"    ; 63
        " Payoff"          ; 64
        " Standard Deviation of change" ; 65
        " Range narrowing streak" ; 66
         " Inside bar";   67
         " Cobra Tail";  68
         " Rectangle"; 69
         " Low acceleration"           ; 70
         " High acceleration"          ; 71
         " Pivot acceleration"         ; 72
         " Close acceleration"         ; 73
         " As projection"          ; 74
         " Excursion ratio"            ; 75
         " Surfing Index"          ; 76
         " Bias Line"          ; 77
         " Punctuation"            ; 78
         " C streak"             ; 79
         " C average"              ; 80
         " PCI Index"                ; 81
         " Volume Acceleration"         ; 82
         " Volume meetsPrice Accelaration" ; 83
         " Zindex"                   ; 84
         " Smash Index"                ; 85
         " Support/Resistance"           ; 86
         " Cobweb Projection"            ; 87
))
```

The decision tree which explains the 10 day data:

If indicator 41 (20/50 MACD) < 0.04324388
we have seen 5 examples in which stock goes up 0.0041214316 on average
Else
if indicator 21 (change improvement streak) is < -0.0066396147
if indicator 48 (decision tree advisor in previous period is < -0.0025321415
we have 1 example in which stock drops 0.005604098

else we have 1 example in which stock goes up 5.398149e-4
else
we have 3 examples in which stock drops -0.00937793 on average

it is derived from this Lisp:

```
(SETF DECISIONTREE
      '(41 0.04324388 10 -0.0012590913 0.96169746
        (-1 -1 5 0.0041214316 0.0 NIL NIL)
        (21 -1.0 5 -0.0066396147 0.11685127
         (48 -0.23491637 2 -0.0025321415 0.42929205
          (-1 -1 1 -0.005604098 0.0 NIL NIL)
          (-1 -1 1 5.398149e-4 0.0 NIL NIL))
         (-1 -1 3 -0.00937793 0.0 NIL NIL))))
```

history of the 88 indicators over time + actual change (not known yet for 10th day)

```
(SETF HISTORY
      '((1022004 IBM 1.5402374 -2.0 -1.1299973 0.0 0.0 1.0 1.0 0.42555004
        0.48029256 0.47499847 2.0 -4.2142816 0.046412587 -1.4441355
        -2.5040603 0.0 -0.7242715 -1.1789486 -1.0 1.0 -1.0 -1.0 -1.5249519
        0.080538906 -0.69493145 0.119220614 1.0 3.6666667 -1.1800003
        -1.1800003 0.20499992 1.3296895 -0.09926091 0.2557323 0.476799
        0.7899972 -1.3157959 0.5342102 -0.80999756 -1.949997 -0.21078491
        -0.014411926 270.0 -1.1100006 -0.61709106 -0.026564877 -1.1188812
        -1.0 0.11789696 -0.58711064 0.961463 0.24751252 -3.5 -0.06862005
        0.5773874 0.66306424 -1.9699936 0.41153717 0.40477878 -
0.012192461
        0.47174639 0.4945299 -1.1909866 -1.1765747 0.011139918 1.5425261
0.0
        -1.0 -1.0 0.782274 0.2730046 0.14701822 0.20198569 0.18596357
        0.34878856 0.48894945 0.0039902995 -1.0868225 114328.64 -2.0
        -0.5503906 -0.28733295 189910.25 35316.387 -2.3713033 1.2289059
        1908.0 25.536316 0.0)
       (12312003 IBM -0.7891027 0.0 0.05000305 0.0 0.0 1.0 1.0 0.4342156
        0.39367324 0.44999695 1.0 -0.52757263 0.055558484 -0.31056893
        -0.3963599 0.0 -0.57706654 -0.6732979 1.0 -1.0 1.0 1.0 -0.21592371
        -0.02265129 -0.095603295 0.27953827 1.0 3.3333333 -0.059856474
        -0.059856474 0.48749924 0.9688649 -0.022430427 0.47530115
        -0.001388188 -0.8884297 -0.77453583 0.051426884 -0.4189679
        -1.2569128 -0.0036439935 0.04324388 270.0 -0.7062589 -0.5676202
        0.013224227 -0.1822088 -1.0 -0.03898557 -0.7351944 0.9120605
```

0.20172381 -3.5 -0.0053984225 0.018731017 0.5604592 -0.22000122
-0.19307709 0.47746786 5.398149e-4 0.4606784 0.5810322 -
0.09395599
-0.13008118 0.009013622 1.5465809 0.0 -1.0 1.0 0.48544145 0.4369995
0.50000244 0.47063476 0.4749935 -0.12556405 0.47222775 6.225499e-
4
0.05998993 545246.94 -1.0 -0.10078126 -0.10133317 506419.28
240545.86 0.8371064 1.1576875 75.0 -0.0028093336 -0.012192461)
(12302003 IBM 1.0669658 -1.0 -0.8899994 0.0 0.0 1.0 1.0 0.46720168
0.7269323 0.89499664 -1.0 -0.50792855 0.6183572 -0.54381716
-0.29888135 0.0 -0.62561727 -0.7821642 -1.0 1.0 -2.0 -2.0
-0.30177197 -0.07819765 -0.20713285 0.29050827 1.0 3.0 -0.115255445
-0.115255445 0.47499847 1.0703707 0.006631601 0.39692217 0.352273
0.7172378 -0.65564066 0.65819055 0.38031307 -0.4264223 0.10699905
0.061654806 270.0 -0.54166895 -0.11336013 0.06362113 -0.46482906
1.0
-0.23491637 -0.66100365 0.8511502 0.29656625 -3.5 0.35609102
0.0838598 0.5072968 0.3600006 0.2389679 0.47721025 -0.009516675
0.4932337 0.6052604 -0.15255737 -0.20605469 0.009367278 1.5542324
0.0 -1.0 -1.0 0.4792585 0.05898519 -0.14500976 0.047333986
0.22801757 0.4337726 0.48066798 8.56254e-4 0.046661377 -7294.6504
-1.0 -0.10078126 -0.32600313 -1004840.1 -229121.2 -1.5233814
1.2840769 30.0 1.277589 5.398149e-4)
(12292003 IBM -2.0253232 1.0 0.6199951 0.0 0.0 1.0 1.0 0.46720994
0.5781801 0.5849991 1.0 0.60959345 0.6647145 0.9732274 1.0271989
0.0
-0.30902106 -0.4706904 2.0 -3.0 -1.0 -1.0 0.88880765 0.14160101
0.34756252 0.86863655 1.0 2.6666667 0.9787102 0.9787102 0.6974983
0.88890225 -0.0026930633 0.7521695 -0.3409597 -1.4887816
0.06398961
0.9312142 1.6601002 0.89198864 0.1754941 0.06947173 270.0
1.4247092
0.87128055 0.1290773 0.5974191 1.0 -0.065080605 -0.6340133
0.7860845
0.2415947 -3.5 0.5066133 -0.11439761 0.5062835 0.7299957 -
0.3577118
-0.5624644 0.00667379 0.4879508 0.73818916 0.7393799 0.68330383
0.008631074 1.5654395 0.0 -1.0 -1.0 0.3207843 -0.3789917 -0.4630249
-0.47666505 -0.5880078 0.03471925 0.45588234 -0.0019653025
0.9216385
1837969.9 3.0 0.7984375 0.35799342 -1423340.2 836935.2 0.19969112
1.1083513 -1045.0 0.6101444 -0.009516675)
(12262003 IBM -1.5732361 1.0 0.6300049 0.0 0.0 1.0 1.0 0.4749614
0.32942113 0.26999664 -4.0 -0.94519246 0.6456638 0.24513099
0.061278637 0.0 -0.52818847 -0.5974057 1.0 -2.0 1.0 2.0 0.12857513
0.034413565 0.16978633 0.7431958 1.0 2.3333333 0.20457202

0.20457202
0.54249954 0.8844827 -0.044554457 0.65651417 -0.26896787 -1.3612381
-0.5465659 0.29579487 0.6738978 -0.6498251 -0.02454093 0.024476662
270.0 0.12033594 -0.06431106 0.0417086 0.32487124 -1.0 -0.35310873
-0.48319253 0.7278631 0.37983668 -3.5 0.07200176 -2.9999065e-4
0.8722299 -0.48999786 0.022087097 -0.55873555 0.006827841 0.4878065
0.6634023 0.14575195 0.12541199 0.00894504 1.5699563 0.0 -1.0 -1.0
0.27318344 -0.11398519 0.10002197 -0.0763322 -0.21503255 -0.277137
0.48801786 -9.533934e-5 0.2240448 -301714.84 1.0 0.19375002
0.33932364 -259269.87 55751.46 1.399586 1.1104565 782.0 -0.7632748
0.00667379)
(12242003 IBM 0.4276021 -1.0 -0.5200043 0.0 0.0 1.0 1.0 0.48316786
0.4797833 0.5299988 -3.0 -0.91320217 -0.8938751 -0.7572009 -0.594435
0.0 -0.86714566 -0.7839716 -2.0 -1.0 -3.0 1.0 -0.6366371
-0.049774352 -0.28812286 0.116448596 1.0 2.0 -0.5348264 -0.5348264
0.38499832 0.8952434 -0.012833595 0.21239714 0.29856572 -0.04054683
-0.6797229 0.04112275 -0.5929535 -1.4300623 -0.016622959 0.027701974
270.0 -0.93012285 -0.49631622 0.023284119 -0.38003063 -1.0
-0.15241869 -0.58021283 0.6937592 0.41513538 -3.5 0.095496535
0.24291898 0.7989214 -0.87000275 0.81749725 0.3278135 -0.005604098
0.48701507 0.64528245 -0.48510742 -0.508934 0.009321608 1.5733668
0.0 -1.0 -1.0 0.27318344 0.2029956 0.25301433 0.29632896 0.43301514
-0.24870817 0.5206815 0.0019077662 -0.4384842 -278218.1 -2.0
-0.61249995 -0.114690244 -78069.875 -33805.437 -0.79469347 1.1331756
-184.0 -0.45067525 0.006827841)
(12232003 IBM 0.4795408 -1.0 -0.5999985 0.0 0.0 1.0 1.0 0.46771285
0.7090316 0.829998 -2.0 -0.85986125 1.0947378 -0.31727383
-0.08877746 0.0 -0.6161339 -0.7215885 -1.0 1.0 -2.0 -2.0
-0.069591485 -0.1046837 0.0376313 0.40787378 1.0 1.6666666 0.0656816
0.0656816 0.5149994 1.0071626 0.029635778 0.4920366 0.19294387
0.28762642 -0.655346 0.5488723 0.13137154 -0.50357836 0.13308375
0.052702297 270.0 -0.043790516 -0.05886673 0.06796045 0.38737443 0.0
-0.97654396 -1.0725812 0.6214325 0.3311095 -3.5 0.02992436
-0.038333844 0.58414423 0.05999756 0.21000671 0.32966095
-0.0064246543 0.48341992 0.74462324 0.021080017 -0.027061462
0.009844388 1.5749923 0.0 -1.0 -1.0 0.21864754 0.513711 -0.09972738

0.28830567 0.45098388 0.1725573 0.48534194 4.1273546e-4 -
0.05797577
-267996.7 -1.0 -0.22499996 -0.3096835 471989.37 212859.6 -0.9221704
0.98381597 353.0 1.4399927 -0.005604098)
(12222003 IBM -1.053479 2.0 0.25 0.0 0.0 1.0 1.0 0.45409536
0.64377683 0.704998 -1.0 0.28164816 1.1467808 0.52992237 1.4016297
0.0 -0.32698435 -0.452138 2.0 -3.0 -1.0 -1.0 0.6639822 0.14764896
0.39993778 0.7845559 1.0 1.3333334 0.7568708 0.7568708 0.664999
0.8486187 0.023900943 0.7306524 -0.19746168 -0.7101574 -
0.22891521
0.59676844 1.0206343 0.011470258 0.19118847 0.054534037 270.0
0.18348038 0.45006 0.104246706 1.0041945 0.0 -0.68439204 -1.388245
0.26590642 0.21625958 -3.5 0.10623959 -1.5246059 0.58721626
1.6499557 -0.15666199 0.3317926 0.0026841315 0.4678281 0.9999987
0.6246567 0.57710266 0.009337255 1.5891227 0.0 -1.0 -1.0 0.4980857
0.037348628 -0.18153076 -0.014108077 0.101848155 0.44845134
0.4634683 -0.0016141571 0.61932373 -2336976.0 2.0 0.55 -0.20842636
0.0 0.0 0.061675146 0.8591997 -1138.0 0.1524405 -0.0064246543)
(12192003 IBM -0.21044776 -0.5 0.40999603 0.0 0.0 1.0 1.0 0.47952795
0.49845764 0.5 2.0 0.0 1.1040148 0.34758008 1.1208119 0.0
-0.24929953 -0.3788575 1.0 -2.0 1.0 1.0 0.39869848 0.6265274
0.15180077 0.17055592 1.0 1.0 0.45264167 0.45264167 0.602499
0.8390514 -0.010554886 0.41418332 -0.15869707 0.03657029 -
0.20730598
0.4330246 2.65336 -0.42364982 0.097243115 0.025437351 270.0
0.15456139 0.46910644 0.08512388 0.32033375 0.0 0.0 -0.97852695 3.5
-3.5 -3.5 -0.31489915 1.3676678 2.2903454 1.2092819 0.10999298
0.4118088 0.0044213957 0.49003485 0.9999985 0.3933487 0.37030792
0.009724984 1.5916578 0.0 1.0 1.0 0.57798 -1.0568603 0.1403955
-0.3885213 -0.2491805 -0.40821746 0.48489028 -0.0010968554
0.2943802
175925.77 1.0 0.1 -0.116867475 0.0 -0.0 0.622046 0.9359996 -226.0
0.16980161 0.0026841315)
(12182003 IBM 0.05366885 0.0 0.0 0.0 0.0 1.0 1.0 0.5 0.5 0.5 1.0 0.0
1.0223054 0.0 1.0 0.0 -0.27619046 -0.27619046 -1.0 -1.0 -1.0 -1.0
0.0 0.6491622 0.37314317 0.45652065 1.0 0.6666667 0.0 0.0 0.5
0.96096563 -0.017251715 0.39772832 0.080507174 1.0451083e-8
-0.28283584 0.61679006 -0.1327955 -1.2684677 0.034068525
0.008017959
270.0 2.0446107 0.8278816 0.08076893 0.033623952 0.0 0.0 0.0
0.07715885 -0.010667702 -0.91730005 0.0 0.031265415 0.23981108
1.444252 -0.069999695 0.0 0.0 0.5 0.5 -0.005493164 -0.013336182
0.010548703 10.0e-9 0.0 1.0 -1.0 0.57798 0.25423992 0.501377
0.11353923 -0.41499025 0.13525248 0.5 9.2963254e-5 -0.12550354
1796541.0 1.0 0.1 -0.24063702 0.0 -0.0 -4.9036497e-4 1.0 18.0
-5.1397734 0.0044213957)

```
        (0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
         0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
         0 0 0 0 0 0 0 0 0 0 0 0 0)))
(SETF DB1
      '(#S(STK
          :NAME IBM
          :DATE 1022004
          :TIME 0
          :SFLOAT 1.0
          :90VOL 90.0
          :SHORTS 0.0
          :INSIDERS 0.0
          :INSTITUTIONS 0.0
          :IDS NIL
          :FAX NIL
          :Z-TABLE NIL
          :MOOD-VECTOR NIL
```

weightings of various decision trees importance over various time frames and other neural net internals

```
          :MM-VECTOR (0.1507634 -0.123149544 0.17906503 0.02537639
                  0.13510539 0.029712915 0.027669989 -0.06681216
                  0.03384554 0.061458457)
          :MP-VECTOR (1.4149358 2.2107365 1.5595443 1.7347609 1.6156236
                  1.8831747 1.6176236 1.862967 0.6966936 1.3848939)
          :PM-VECTOR (0.06483989 0.15998861 0.27528226 0.06166677
                  0.27808005 0.1671105 0.055806223 0.10185026
                  0.12067121 0.30529913)
          :PP-VECTOR (1.4089518 -0.38906765 -0.32473803 1.8556639
1.091323
                  2.3230762 2.3117182 1.0310829 2.1580915 1.0966296)
          :MMX (-0.012192461 5.398149e-4 -0.010854568 0.003683828
              -0.012192461 -0.012192461 -0.012192461 -0.012192461
              0.00667379 -0.012192461)
          :MPX (-0.1 0.1 -0.1 0.1 -0.1 0.1 -0.1 -0.1 -0.1 -0.1)
          :PMX (-0.012192461 -0.005826323 -0.010854568 -0.00448843
              5.398149e-4 5.398149e-4 5.398149e-4 5.398149e-4
              5.398149e-4 5.398149e-4)
          :PPX (-0.1 0.1 -0.1 -0.1 0.1 0.1 0.1 0.1 0.1 0.1)
          :NEURAL-VECTOR (14.557797)
          :NEURAL-AVG (0.64492416)
          :SW (6.006428 3.0073466 -6.9988804 3.005674 3.0049248
                  2.981453 22.008978 3.0030477 12.006138 3.0479155
                  50.008926 2.9519193 10.007889 3.037959 11.002487
                  2.9765859 106.00992 2.9151645 93.000786 3.1678839)
```

```
:NEURAL-BLOCK (-0.123601556)
:Ws ((3 -0.023518085 2 0.028136253 1 -0.017460346 1
      0.012187481 1 -0.01915741 1 0.021342278 1 -0.0144844055
      1 0.0126338005 2 -0.032789707 2 0.015382767)
     (4 0.02615261 8 -0.028936386 2 0.01712513 2
      -0.0074543953 3 0.019226551 6 -0.016995907 10
      0.017064095 2 -0.0052728653 2 0.0021848679 6
      -0.019257545)
     (0 0 0 0 0 0 0 0 53 0.007168293 4 -0.0035033226 11
      0.0029010773 10 -0.011811256 21 0.0118403435 10
      -0.022345543)
     (0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 57 0.0036649704 52
      -0.019415379)
     (0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 109 -0.015750408))
:W1 4.0514975
:W2 1.2300371
:W3 1.8065244
:W4 0.57437927
:W5 3.3796163
:W6 2.8148468
:W7 0.19105533
:W8 3.669929
:W9 15.472191
:W10 7.342058
:W11 3.7999926
:W12 6.6300097
:ALPHA 0.19373706
:EFFECTIVEAGE 5.5
:WINNINGSTREAK 1.0
:BETA 1.5317254
:GAMMA 0.028744213
:DELTA -0.2673816
:ETA (2.9088676 0.015106827 -5.9619484 -9.636207 -3.6395347e-4)
:FETA (3.0935512 0.035255685 -4.402421 -10.647896 3.7868834e-4)
:GRETA (2.8522272 0.14311253 -3.6339023 -9.251171 9.841259e-4)
:HETA (0.64307696 0.11693882 11.243773 30.95407 4.4609094e-4)
:IOTA 0.05
:JOTA 0.05
:KOTA 0.05
:MRMO -1.9699936
:CB 1908.0
:BALANCE 0.66306424
```

a spectrum indicator with 3 possible settings

```
:B-TABLE (((5 1.0 1) . 4.515751) ((5 3.0 -1) . 2.8301158)
```

((10 2.0 -1) . 2.031228) ((10 3.0 -1) . 1.9519618)
((5 2.0 -1) . 1.0448267) ((10 1.0 1) . 0.82095575)
((10 1.0 -1) . -0.7591944) ((5 2.0 1) . -0.9012246)
((10 3.0 1) . -1.8422732) ((10 2.0 1) . -1.9449581)
((5 3.0 1) . -2.789735) ((5 1.0 -1) . -4.438587))

a decision tree explanation of the data in terms of just advisors 0-5 as specified above:
this decision tree produces "Real Prediction" (indicator 58)

:TOP-TREE (0 5.60451e-4 9 -0.0013989904 0.96169746
(2 -0.07132701 4 0.0030797324 0.7920296
(1 0.002368509 2 -5.913512e-4 0.9899378
(-1 -1 1 0.0044213957 0.0 NIL NIL)
(-1 -1 1 -0.005604098 0.0 NIL NIL))
(-1 -1 2 0.0067508155 0.0 NIL NIL))
(0 0.002368509 5 -0.0049819686 0.47784603
(-1 -1 2 0.0016119732 0.0 NIL NIL)
(-1 -1 3 -0.00937793 0.0 NIL NIL)))
:HORIZON 10
:ASP 1.0226322
:VREALPRED -0.2616897
:ASP2 0.02

how well we are doing on this stock (1.00 is neutral):

:THRUST 1.0214462

sequence of correct/wrong (some predictor may be able to operate over this...)

:SWITCHO (1.0 -1.0 1.0 -1.0 -1.0 1.0 1.0 -1.0 1.0)

sequence of raw predictions
; one could predict the prediction from this data

:PLIST (-0.8890016 -0.6746359 -0.69925356 -0.5112092 -0.5607992
-1.3187201 -3.0725813 -1.388245 -0.97852695
0.0010783997)
:BETPRED -0.8890016

these 5 predictions come from the PROFESSIONAL PROPRIETARY BOXES

:NEURALPRED -0.12360155
:TIMEPRED -0.6784082
:APRED 0.27325237
:PROJECTION 7.8674235

:SURFINDEX 0.003653119

data that simply needs to be maintained to calculate the 88 base indicators

:ZINDEX -2.3713033
:PCIINDEX -0.28733295
:UPFAC 0.0
:DOWNFAC 0.0
:UPRETR 1.0
:DOWNRETR 1.0
:TRENDBEGIN 1.0
:PREVTRENDL 1.0
:RSI8 0.47942698
:3FAST 0.513501
:3SLO 0.528049
:OBV10 -102596.5
:FIDELITY -1.0
:VOLUME 5331200
:CLX3 -0.61709106
:CLX40 -0.026564877
:MAVG8 92.5656
:MAVG17 92.68631
:MAVGDIFF9 0.0029696915
:PREVSTOC 0.014548004
:PREVMACD -0.12368197
:BUYPRESSURE 0.5773874
:AX 0.6828899
:AXX 0.64408255
:ASMASH 0.7576605
:DSMASH 0.61653256
:DX -0.921833
:DXX 0.67319584
:AXR 0.77440107
:DXR -0.8671612
:AXV 1093122.0
:DXV -1117304.6
:DHI 93.957214
:DLO 91.10654
:PPHI 93.73
:PPLO 91.73
:FUEL 0.1632978
:POSREACT -4.2142816
:NEGREACT 0.046412587
:3PV 92.344864
:3PVDISTSUM -2.4777176
:3PVAVGSUM 0.98948

:|5F| -0.17588955
:|34F| 0.5483819
:|5S| -0.77709675
:|34S| 0.4018519
:STREAK -1.0
:RSTREAK 1.0
:CSTREAK -2.0
:CAVG -0.5503906
:NRSTREAK 0.0
:FSTREAK -1.0
:CSTREAK -1.0
:VSTREAK 2.0
:PP 0.41153717
:TRPRED 0.3386259
:REALPRED 0.40477878
:MMPRED 0.0026064336
:MPPRED -0.014835936
:PMPRED -0.0062555214
:PPPRED -0.027746698
:NNPRED -0.58711064
:DTPRED 0.11789696
:BPRED 0.961463
:JOEPRED 0.24751252
:STTPRED -3.5
:13AVG 92.69669
:PUBLIC 0.080538906
:PROF -0.69493145
:BEARPOWER 1.4378585
:BULLPOWER 0.19462577
:UPDOWN 1.0
:KEYHI 94.73
:KEYLO 90.73
:BIGHIGH 94.73
:BIGLOW 90.73
:TRENDLEN 11.0
:TRENDAVG 3.0
:TRENDRANGE 11.980011
:AVGPIVAS 92.73
:PREVPIVAS 92.73
:CURPIVAS 92.72942
:RANGE3 1.3296895
:34AVG 92.73904
:5AVG 92.51579
:20AVG 92.72658
:50AVG 92.74099
:34DIFF 0.07515683

```
:5DIFF -0.024104085
:OBOSAVG 0.2557323
:RANGAVG 1.0198596
:CDIVR -0.6108073
:ADPAVG 92.30024
:ADPMOM -0.27343714
:FIBS NIL
:FIBR NIL
:BUYERS ((2823909.2 93.42667) (3670418.2 93.22333)
      (5745909.0 93.02) (1796541.0 92.87)
      (1504316.1 92.85667) (948162.44 92.829994)
      (1207287.9 92.73334) (1.0 92.73) (3562305.0 92.56333)
      (255544.92 92.416664) (1008619.7 91.93333))
:SELLERS ((4322580.5 91.93333) (1164594.7 92.56333)
       (3055237.5 92.829994) (1210290.9 93.42667)
       (201212.1 92.73334) (1504955.1 92.416664)
       (2172884.0 92.85667) (661881.75 93.22333)
       (1344791.2 93.02) (5077059.0 92.87) (1.0 92.73))
```

some of these lists are extra long because they were padded with random data in order for us to get started...

```
:CLOSES (91.55 92.68 92.63 93.52 92.9 92.27 92.79 93.39 93.14
      92.73 91.74004 91.93072 91.28575 93.033905 93.98393
      91.26687 93.269615 93.85047 91.445984 92.236885
      92.88247 92.9717 92.81057 94.39887 94.15404 93.6077
      94.42417 93.833824 93.37601 92.84735 93.994446
      94.067894 93.96624 91.49905 92.23618 93.24622 91.11579
      90.98263 93.67065 94.11397 91.88977 93.938 92.44306
      94.53937 94.21963 91.21531 94.00554 91.79722 90.90467
      94.0968 92.921165 92.01977 94.35154 90.89607 93.906044
      91.309 92.25376 91.21509 93.183975 92.15749 91.47192
      93.69706 93.08604 94.20935 93.36084 94.336586 92.649155
      94.37689 90.95084 92.13251 91.537315 91.641975 91.74822
      91.57626 94.3805 94.58095 93.621086 94.54616 92.03146
      94.22132 92.720055 93.927536 91.221695 90.954414
      92.996346 93.60541 91.51035 92.05999 93.556244 92.51345
      93.12651 91.57633 93.601135 91.93201 91.36921 91.59955
      93.46512 93.54348 92.93557 90.94821 92.550156 91.42902
      93.19747 91.2295 92.56051 91.39846 92.32506 94.44509
      93.46803 93.00335)
:OPENS (92.86 92.66 93.27 93.1 92.37 92.8 93.07 92.83 93.23 93.0
      91.98492 91.044716 93.41333 94.04303 93.800644 92.06545
      91.71454 92.0751 90.99212 93.570175 92.039925 92.92551
      94.56569 94.42902 91.81554 91.41762 93.84971 91.283585
      93.163246 93.06751 90.974556 93.78293 92.00946 93.29114
```

91.33265 90.97176 91.42287 91.8197 93.559975 93.60619
94.560036 93.673096 94.3784 91.41294 92.34884 92.878296
92.61052 92.39955 93.4069 93.01851 93.639336 93.796776
92.03935 93.35834 91.89545 94.00211 91.009964 92.40333
92.49808 92.43249 94.13866 93.15493 94.06985 91.2678
91.50223 92.586525 93.293945 91.81063 93.74721 91.529625
92.63409 93.96139 94.216774 93.40093 92.96018 92.04832
93.89096 92.83607 91.96518 93.29297 91.22922 90.94771
91.71848 93.38427 94.0663 91.25845 91.96286 92.1117
93.10101 93.34224 92.944595 94.50834 94.51433 94.54356
93.752266 90.92416 92.72726 94.557434 93.17417 93.90183
93.920586 92.890495 91.91574 92.470665 91.91894 92.35035
90.942726 93.51733 92.126236 92.93425)

:VOLUMES (5331200 4726900 4003400 4034200 1408500 1760500
3677200 4332300 7090700 6873600)

:HIGHS (93.05 92.85 93.5 93.73 93.0 92.8 93.44 93.5 93.25 93.38
93.523735 93.970795 92.93684 92.64361 95.56854 92.26984
95.262436 93.35379 93.91076 92.23941 94.323746 95.48162
94.00925 92.19689 94.727 92.9738 94.88199 93.242874
91.98391 94.79027 93.67591 94.88214 93.798386 95.41642
93.05751 95.47188 93.883224 93.07867 94.07318 93.70278
94.89553 95.122154 94.2971 94.4734 95.514984 93.11012
94.15307 93.91022 93.319336 92.93893 95.33938 92.55473
94.15447 95.54676 92.50223 93.35712 95.118774 94.408775
92.219635 95.47531 94.34505 95.14002 92.79374 94.292496
92.97647 93.22187 93.03264 91.98377 92.89614 94.90171
92.43391 92.01859 94.81219 92.69529 93.19884 92.16148
94.42064 93.143196 92.96498 94.34496 92.843735 95.35876
94.56713 91.95538 92.572845 94.036194 92.833725 93.31918
91.90966 94.83341 92.916695 93.57593 94.82009 94.19634
95.54373 92.22193 91.95599 92.44003 94.197334 91.96026
93.991455 93.87787 93.702705 94.96929 94.31321 91.91799
93.94887 95.04628 94.12736 93.862816)

:LOWS (91.2 92.16 92.36 93.03 92.3 92.18 92.34 92.78 92.67 92.5
90.736626 92.8599 92.94094 91.02654 92.228546 92.25229
91.71631 90.26318 89.9729 90.9911 93.247604 89.95115
90.359726 89.990486 92.90693 91.85003 92.838455 91.68948
93.35178 92.99663 91.61903 91.65791 92.847275 90.40377
93.13453 91.87915 92.79389 92.8706 92.855316 92.44184
93.2468 91.34777 90.05708 91.70407 91.66604 91.035
90.439285 90.87655 92.45764 91.36684 90.72083 93.03445
91.17679 90.71797 90.742935 92.60732 91.9584 93.179184
92.206055 91.19661 90.93113 91.05153 92.45668 93.18138
90.76875 92.44293 92.60666 90.0069 91.97063 93.35686
91.29316 91.23389 91.59776 91.8878 91.909904 91.89249
93.376434 93.38769 90.71916 91.79743 93.46338 91.76918

91.76349 92.68131 92.01528 91.23595 91.270485 92.7896
91.78704 90.17036 90.580536 92.63508 92.3129 90.12361
92.88669 91.84872 92.40741 91.450874 93.51858 90.347374
93.36154 89.930115 89.95011 91.97708 91.970695 93.037605
93.301796 90.072205 91.06223 90.32682)
:PIVOTS (91.93333 92.56333 92.829994 93.42667 92.73334
92.416664
92.85667 93.22333 93.02 92.869995 92.00013 92.92047
92.38785 92.23469 93.927 91.929665 93.416115 92.48914
91.77655 91.82246 93.4846 92.80149 92.39319 92.19541
93.92932 92.81051 94.048195 92.92206 92.9039 93.54475
93.09646 93.53599 93.53729 92.43975 92.80941 93.53242
92.59763 92.31063 93.53305 93.419525 93.34403 93.46931
92.26575 93.57227 93.800224 91.78681 92.86597 92.194664
92.22721 92.800865 92.9938 92.536316 93.2276 92.38694
92.38373 92.42448 93.11031 92.93435 92.53656 92.94314
92.24936 93.2962 92.77882 93.89441 92.36869 93.3338
92.76282 92.12252 91.93921 93.463684 91.75479 91.63149
92.71939 92.05312 93.163086 92.8783 93.80605 93.69235
91.905205 93.45457 93.009056 93.68516 92.51743 91.8637
92.52816 92.95919 91.87152 92.72292 92.417656 92.50574
92.20792 92.59577 93.57804 92.083984 93.26654 91.89007
92.609505 92.47813 93.5505 91.08529 93.30105 91.74567
92.283424 92.72529 92.94814 92.11802 93.1919 93.18786
92.88587 92.39767)

:PO 92.86
:PH 93.05
:PL 91.2
:PC 91.55
:OPEN 92.86
:HIGH 93.05
:LOW 91.2
:CLOSE 91.55
:SURPRISE 0.7899972

predicted magnitude

:MAGNITUDE -0.93

our current confidence

:CONFIDENCE 6.1286774
:MAGN-SDEV 1.5382873

current prediction in pct.

:CHANGE -0.010158383)))

Further Description of An Embodiment

Thus practice of embodiments of the present invention performs one or more of the following features:

i) The modularization of a machine learning based system and provision of four insertion points where a non-technical user of the system can specify or add their own (1) securities instruments and factors, (2) mathematical indicators for pre-processing raw instrument and factor data and producing output signals (3) complex, higher level "advisors" or models producing output signals, and (4) ultra-complex, top-level "overlay advisors" producing output signals, through a user interface that does not require programming skill, instead of a hard coded closed system.

ii) The ability to utilize any number of user specified signal generating strategy elements at each insertion point.

iii) The use of a base set of "advisors" that consist of machine learning components with no strategy or market bias and a short-term trend advisor which can be

modified or removed by the user, together with user inserted signal generating strategy elements which outputs are collectively used as inputs to a neural network combiner which learns their optimum weighting for the current prediction task.

iv) The use of nearest neighbor and decision tree algorithms to pre-process technical indicator output signals for use as input to a neural network.

v) The use of nearest neighbor and decision tree algorithms in combination with user inserted signal generating advisor models to produce outputs that are used as inputs to a neural network.

vi) The use of "spectrum" variable sets for a single type of signal generating strategy elements where an array of variables is used when processing data, each set of variables in the spectrum producing a different output which are then collectively considered and independently evaluated for each prediction task with the best version (one variable set) selected, and the ability to employ spectrum processing at each processing layer. As in moving averages.

vii) The use of a second processing layer, where user specified top-level advisors can be inserted as signal generating overlay advisors, which output signals are then together with the output signal from the preceding neural network used as inputs to a second neural network that learns the optimum weighting of all high level output signals to produce a final single prediction for each new prediction task.

viii) The modeling of all instruments and factors and signal generating strategy elements independently with a single iterative process.

ix) An iterative method of predicting securities instrument prices using user specified or inserted factor instruments, user specified or inserted mathematical indicators, base machine learning algorithms and user specified or inserted mid-level advisors and user specified or inserted high-level advisors at three distinct processing layers, with the first processing layer consisting of processing raw time series data with mathematical indicators which outputs are used as inputs to the mid-level advisors at the second processing layer which consists of decision tree and nearest neighbor algorithms and other user specified or inserted advisors and which outputs are used as inputs to a

neural network, which produces an output signal that is then considered in combination with the output signals of high-level overlay advisors at the third processing layer by a second neural network that produces the final output signal for each instrument modeled by the system for the current prediction task.

[135] The identification and categorization of market forces exerted by all market participants as "bulls" versus "bears" with specific indicators and learning mechanisms designed to indentify and exploit these groups. The market activity that is not explained by the combinations of advisor weights in the neural networks are assigned to the "bulls" on an up day and to the "bears" on a down day (as if there was a bull advisor (always up) and a bear advisor (always down).

[136] Some portions of the detailed description are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of operations leading to a desired result. The operations are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms numbers, or the like.

[137] Various embodiments of the invention are next discussed with respect to Figures 1-27.

DATA MODEL OVERVIEW:

[138] In general, the user interface described herein allows a human user to select, manage and manipulate data, and processes that use the data, to provide a prediction, recommendation or other analysis of a system being modeled. In a securities

analysis application, the price movement of instruments such as stocks, bonds, futures, etc., is predicted by modeling data points of instruments and "factors." In the context of one embodiment of a securities analysis application, instruments may also be designated as factors in the price movement of other instruments. Additionally, data types such as Price/Earnings Ratio, dividends, etc., and other data such as macroeconomic data (e.g., an unemployment rate, domestic product, etc.), can also be designated as "instruments," so that they can be the target of the modeling processes.

[139] Both instrument and factor data are referred to as "raw data." The instrument and factor raw data are used to derive other data called "indicators" which include results of pre-processing the raw data. For example, an indicator can be used to calculate a change over prior data in a series The resulting indicator outputs and other possible data (e.g., the raw data itself, or data from external data "feeds") are used as inputs to processes called "advisors," which can include one or more machine learning based components (e.g., "combiners" discussed in more detail, below) to collect, or group, the outputs from lower-level advisors to arrive at an ultimate recommendation or prediction of the price movement of one or more instruments. In one embodiment, a user may add one or more additional processing layers, where the outputs of the combiners at the preceding processing layer are used as inputs for another group of advisors called "overlay advisors," whose outputs are then used as inputs for another group of combiners to arrive at an ultimate recommendation or prediction of the price movement of one or more instruments.

[140] Figure 23 illustrates basic data items, processes and data flow in a data model for a preferred embodiment of the invention. In Figure 23, the analysis system, or "portfolio," 901 includes predictor instruments 907 such as stock AA 909 and BB 911. Naturally, any number of instruments or other modeling targets can be designated as predictor instruments. In practice the number of predictor instruments can be in the hundreds or thousands or more, as desired. For ease of illustration, the number and complexity of the data items discussed in Figure 23 is greatly reduced.

[141] Predictor instruments 907 are chosen from a large pool of possible instruments 903. Although the example of Figure 23 uses stocks, any type of financial instrument, or other data, can be used. In other embodiments, other systems or processes

can be applied with features of the present invention. For example, the invention can be used to determine the outcome of sports events, determine an individual most likely to be successful in business or scholastic aptitude, identify social trends such as growth of cities, predict productivity or output of different markets (e.g., agricultural, industrial, technology, etc.), predict the outcome of an election, etc. In general, techniques or functionality described herein may be applied to any type of analysis system.

[142] A user can choose to use less than the entire possible set of instruments by performing instrument selection 905. Such selection is useful, and often necessary, to reduce the amount of processing power and other resources (e.g., memory, bandwidth, etc.) required for achieving desired results. Also, allowing user selection of data items, parameters, processes, inter-process communication, and other characteristics of the system is important to allow the user to customize and direct the system to achieve desired results.

[143] The extent of a user's ability to select, set, or otherwise affect characteristics of the system can vary with different embodiments. Normally this is a design tradeoff between complexity of use and degree of user control over the system. A preferred embodiment of the system attempts to strike a balance between complexity and control. Automatic selecting or a combination of manual and automated selecting can be used. Default or stored selections can similarly be used along with other options or ways to perform selection of current optimum configuration. Other selection of characteristics or settings of the system, discussed below, can also be set in various ways. The selections can be stored as part of the overall "portfolio" that a user designs, which configuration can be reused in subsequent portfolios created by the user. Or portions of the settings (e.g., just the instrument selections) can be stored as a recommended or available predetermined selection set, e.g., default and user defined libraries.

[144] Each instrument includes data points that are associated with it. For ease of illustration, each of the predictor instruments in Figure 23 are shown to have five data points represented as open ("O"), high ("H"), low ("L"), close ("C") and price/earning ratio ("P/E"). In actual implementation many more data points can be included, some of which are shown in Table I, below.

========== BEGIN TABLE I ==========

Sector and Key Mover Factors

IXCO NASDAQ Computer Index
IXF NASDAQ Financial 100 Index
MOX Morgan Stanley Internet Index
MSH Morgan Stanley Technology Index
PSE DJ PSE High Technology Index
NDX NASDAQ 100 Index
OEX S&P 100 Index

Volatility and Trend Factors

VIX CBOE Volatility Index
TRIN NYSE Short Term Trading Index
TRIT NASDAQ Stocks Short Term Trend Index

Interest Rates, Exchange Rates and Commodities Factors

TNX Ten Year Treasury Note Index
TYX Thirty Year Treasury Bond Index
VIX CBOE Market Volatility Index
OIX CBOE Oil Index
GOX CBOE Gold Index
OSX Oil Service Sector Index
XAU Phil American Gold & Silver Index
XEU Euro Index

Macroeconomic Announcement Factors

Change in non-farm payrolls
Unemployment rate
Employment cost index
Durable goods orders
NAPM manufacturing
NAPM non-manufacturing
Advanced retail sales
Industrial production
Consumer price index

========== ENDTABLE I ==========

[145] Instrument data points can be filtered by selection of data types, or factors. A default condition is to use all available data points from all predictor instruments as factors. However, if a user chooses to select specific factors then the selected factors will be used in analysis. Other ways of selecting factors are possible such as by using all

types of data points except for the selected factors, allowing Boolean or relational expressions to determine sets of selected factors, etc. Figure 23 shows active factors 917 including open 919, close 923 and P/E ratio 921 data point types. The active factors have been selected from a larger set of possible factors 913 by factor selection 915. Typically, selection of factors and other data types is by user selection. However, factor selection can be achieved in different ways as described above in connection with characteristic settings for the portfolio.

[146] Active factors can include data types other than the data point types of the target instruments (i.e., OHLC). For example, macroeconomic data such as unemployment rates and consumer price index could be used. The data point representation that will be used in the modeling processes may itself be specified by the user, and can include integer, floating point, ordinal, etc. specifications.

[147] Indicators are used to operate on or transform active instruments, factors and other data into indicator outputs for processing by advisors. Many indicators can include computations that are standard in financial analysis. For example, a 25-day moving average of the closing price of an instrument can be computed. The computation can use the data point frequency of the close of a stock at the end of each day in the 25 day average. Other indicators can use custom, non-standard, yet-to-be-created or any other type of computation or processing. As discussed below, indicators can use parameters that are set by a user, obtained as defaults, obtained from a network source, set automatically, etc. For the moving average indicator, typical parameters include the time-series sampling frequency (e.g., per day, per week, per hour, etc.) and the interval over which to compute the average. Examples of indicators are shown in Table II, below.

========== BEGIN TABLE II ==========

Facilitation in Uptrends: total change/ total range (EMA: 4 trends)

Facilitation in Downtrends: total change/total range (EMA: 4 trends)

Average Up Retracement: total change/total range divided by same for previous trend (EMA: 4 trends)

Average Down Retracement: total change/total range divided by same for previous trend (EMA: 4 trends)

8 Day Day Fast RSI: 3 day fast stochastic (DiNapoli) (EMA: 3 periods)

3 Day Slow RSI: 3 day slow stochastic (DiNapoli) (EMA: 3 periods)

MA 8: average close (EMA: 8 periods)

MA 17: average close (EMA: 17 periods)

MA Difference 9: average of MA17 - MA8 (EMA: 9 periods)

Average Advance 15 EMA: average up move (EMA: 15 previous up moves)

Average Decline 15 EMA: average down moves (EMA: 15 previous down moves)

Positive Reactivity: change/range after up move (EMA: 3 up moves)

Negative Reactivity: change/range after down move (EMA: 3 down moves)

3 Day Pivot: hi+lo+close/3 (EMA: 3)

5 Day Average Facilitation: change/range (EMA: 5)

34 Day Average Facilitation: change/range (EMA: 34)

5 Day Average Force: change * range (EMA: 5)

34 day Average Force: change * range (EMA: 34)

Winning/Losing Streak: consecutive up periods, if positive; consecutive down periods, if negative

Positive Range Streak: consecutive periods range beats 3 period average range

Negative Range Streak: consecutive periods range smaller than 3 period average range

Positive Facilitation Streak: # of periods of increasing change/range

Negative Facilitation Streak: # of periods of decreasing change/range

13 day Moving Average: average close (EMA: 13)

Public Power: open - previous close (EMA: ATL)

Pro Power: close - open (EMA: ATL)

Bear Power: close - low (EMA: ATL)

Bull Power: high - close (EMA: ATL)

Trend Up/Down: proprietary short term trend indicator (1 or -1)

Current Average TrendMagnitude: current trend length total change (EMA: 4 trends)

3 period Average Range: average range (EMA: 3 periods)

34 Day Moving Average: average close (EMA: 34 periods)

5 Day Moving Average: average close (EMA: 5 periods)

34 Day Difference from Average: 5 day moving average -34 day moving average (EMA: 34 periods)

5 Day Difference from Average: 5 day moving average - 34 day moving average (EMA: 5 periods)

10 Day Range Average: average range (EMA: 10 periods)

Adaptive Fair Price: proprietary estimate of "correct" close based on rating high range periods higher

Pivot Trend Clock: total distance price has stayed on one side of 3 day pivot (EMA: 3)

Within Drummond Range: (close - Drummond Low) / (Drummond Hi - Drummond Low) (EMA: ATL)

Within Current Key Range: proprietary version of Drummond Range using Key High and Key Low from proprietary short-term trend indicator

Within Price Pulse Range: same as Current Key Range but using Price Pulse High [+2.618 * (-Key High * close)] and Price Pulse Low [+2.618 * (-Key Low * close)] (EMA: 3)

Pivot Tension: distance from 3 day pivot / average distance from 3 day pivot

Momentum Speed: indicator from Bill Williams "New Trading Dimensions" (close - close 5 periods ago) / (close - close 34 periods ago)

Relative Range Size: 3 day range average / 10 day range average

Break Direction Indicator: scoring system that gives next day's trade direction [Let Prevpiv = previous pivot; Let Buybreak = 2Prevpiv - previous low; Let Sellbreak = 2Prevpi - previous high] If close above buybreak or if low above sellbreak get 2 points or else get 1 point; If close below sellbreak if high above buybreak lose 2 points or else lose 1 point; If high above buybreak lose 0.5 points if low above sellbreak lose -0.5 points

Fuel: proprietary ATL day EMA of sum of number of elements in {previous open, previous close previous high, previous low} that open, high, low, and close beat; high fuel (above 8) is usually considered bullish.

Note: References to "ATL" mean average trend length, which is a dynamically adjusted by calculating a 3 period exponentially weighted moving average ("EMA"; default) of a selected short term trend indicator ("STT" as described later is a system default).

---------- END TABLE II ----------

[148] In Figure 23, P/E ratio 921 is an input to indicator 937. Indicator 939 uses indicators 919 and 923 as inputs which represent the open and close of different instruments. For example, if 10 instruments are selected as predictor instruments, then indicator 931 would use the open and close values for each of the 10 instruments. Further, if indicator 939 has sampling frequency as a parameter and the sampling frequency is set to "daily" and assuming a 25 day interval then indicator 939's computation will use 10 x 2 x 25 = 500 data points. Other indicators such as 941 can use data that is obtained from sources other than the active factors. For example, data feed

951 is supplying indicator 941 with data for computation. Other data sources can include user-input data, data obtained from data stored on a user's local computer or generated by a user's company, etc. In general, any type of data or computation can be used at one or more points of the system shown in Figure 23. For example, data feeds can be used to obtain predictor instruments, active factors, etc.

[149] Active indicators can be selected from a pool of possible indicators 931 via indicator selection 933 similarly to selection of other items of the portfolio.

[150] Advisors use the output of indicators as inputs. Advisors can be configured to use any type, number or combination/sequence of indicator outputs, data and parameters. For example, data feed 951 is shown as an input directly to active advisor 957. Advisors can similarly be selected from a pool of possible advisors 961 via advisor selection 963 to result in active advisors 965.

[151] Advisors can couple to other advisors such that the output of one advisor is used as the input to a second advisor. For example, the outputs of active advisors 953 and 957 are provided to active advisor 955. Examples of advisors are provided in Table III, below.

[152] Advisor outputs can be used as inputs to other advisors called overlay advisors, and or machine learning based components called combiners. Ultimately, an advisor is used to provide an output signal that represents the advisor's predicted change in the target instrument price upon which a recommendation or prediction of a target instruments 971 is based, in part (generally, factors will also influence advisor outputs, and these advisor output signals are subjected to additional processing as described in detail later). For example, the advisors output signal can be an increase or decrease in price which could correspond to a "buy" or "sell" recommendation for target instrument. User's can select the target instruments from possible instruments 903 via instrument selection 981, but more typically the target instruments will include all instruments in the portfolio and the top-level advisors will check each instrument and provide a recommendation. Advisors can use any type of data, including data from instruments, to arrive at a result.

[153] In Figure 23, advisors 953 and 957 output their results, or signals, to overlay advisor 955. An overlay advisor is an advisor that does not output its result to

another advisor. The overlay advisor's output is used (in whole or in part) to generate a prediction of performance or recommendation of an action. An overlay advisor's output can be used by a combiner such as combiner 987 to produce the final outcome or recommendation for a prediction of a target instrument's behavior.

========== BEGIN TABLE III ==========

OptionalAdvisors

Joe DiNapoli Advisor: Fibonacci based day trading system as outlined by Joe DiNapoli

Equity Trading Advisor: equity day trading system using all current coded indicators with proprietary scoring system

Mutual Fund Trading Advisor: proprietary mutual fund day trading system

Optional Overlay Advisors

Surprise Overlay Advisor: evaluates difference between actual close and predicted close; close - predicted close (EMA: 1)

Momentum Overlay Advisor: reviews total change in last ATL periods; close - close ATL periods previous

Pattern Analysis Prediction Overlay Advisor: reviews signals from mid-level pattern analysis advisors to approximate the populations of traders correlated with following them or "fading" them (leaving off following them)

Buying Pressure Overlay Advisor: proprietary spectrum indicator that adjusts for trending versus chopping market movements

Pivot Point Overlay Advisor: proprietary day trading system related to distance from 3 period pivot points

Balance Overlay Advisor: estimated bulls - estimated bears as determined from review of pattern analysis routines

---------- END TABLE III ----------

[154] Combiners such as 987, 989, 991 and 993 can be used at different points in the system. In Figure 23 combiners are assigned to process information from factors, indicators, advisors and overlay advisors. A combiner is used to combine, aggregate or compute two or more inputs. Combiners can include machine learning algorithms to detect relationships between or among input data. In a preferred embodiment a user is able to select and assign combiners, as shown at 985, from a library of possible

combiners 983. In general, combiners can be assigned at any point in the system with any type and number of data items or outputs.

BASIC HARDWARE:

[155] Figure 24 is an illustration of computer system 1 including display 3 having display screen 5. Cabinet 7 houses standard computer components (not shown) such as a disk drive, CDROM drive, display adapter, network card, random access memory (RAM), central processing unit (CPU), and other components, subsystems and devices. User input devices such as mouse 11 having buttons 13, and keyboard 9 are shown. Other user input devices such as a trackball, touch-screen, digitizing tablet, etc. can be used. In general, the computer system is illustrative of but one type of computer system, such as a desktop computer, suitable for use with the present invention. Computers can be configured with many different hardware components and can be made in many dimensions and styles (e.g., laptop, palmtop, pentop, server, workstation, mainframe). Any hardware platform suitable for performing the processing described herein is suitable for use with the present invention.

[156] Figure 25 illustrates subsystems that might typically be found in a computer such as computer 100 of Figure 24.

[157] In Figure 25, subsystems within box 20 are directly interfaced to internal bus 22. Such subsystems typically are contained within the computer system such as within cabinet 7 of Figure 24. Subsystems include input/output (I/O) controller 24, System Random Access Memory (RAM) 26, Central Processing Unit (CPU) 28, Display Adapter 30, Serial Port 40, Fixed Disk 42 and Network Interface Adapter 44. The use of bus 22 allows each of the subsystems to transfer data among the subsystems and, most importantly, with the CPU. External devices can communicate with the CPU or other subsystems via bus 22 by interfacing with a subsystem on the bus. Monitor 46 connects to the bus through Display Adapter 30. A relative pointing device (RPD) such as a mouse connects through Serial Port 40. Some devices such as Keyboard 50 can communicate with the CPU by direct means without using the main data bus as, for example, via an interrupt controller and associated registers (not shown).

[158] As with the external physical configuration shown in Figure 24, many subsystem configurations are possible. Figure 25 is illustrative of but one suitable configuration. Subsystems, components or devices other than those shown in Figure 25 can be added. A suitable computer system can be achieved without using all of the subsystems shown in Figure 25. For example, a standalone computer need not be coupled to a network so Network Interface 44 would not be required. Other subsystems such as a CDROM drive, graphics accelerator, etc. can be included in the configuration without affecting the performance of the system of the present invention.

[159] Figure 26 is a generalized diagram of a typical network.

[160] In Figure 26, a network system includes several local networks coupled to the Internet. Although specific network protocols, physical layers, topologies, and other network properties are presented herein, the present invention is suitable for use with any network.

[161] In Figure 26, computer USER1 is connected to Server1. This connection can be by a network such as Ethernet, Asynchronous Transfer Mode, IEEE standard 1553 bus, modem connection, Universal Serial Bus, etc. The communication link need not be a wire but can be infrared, radio wave transmission, etc. Server1 is coupled to the Internet. The Internet is shown symbolically as a collection 80 of servers. Other common devices and structures such as routers, switches, backbones, etc., are not shown in this simplified diagram. Note that the use of the Internet for distribution or communication of information is not strictly necessary to practice the present invention but is merely used to illustrate a preferred embodiment, below. Further, the use of server computers and the designation of server and client machines is not crucial to an implementation of the present invention. USER1 Computer can be connected directly to the Internet. Server1's connection to the Internet is typically by a relatively high bandwidth transmission medium such as a T1 or T3 line. Similarly, other computers at 164 are shown utilizing a local network at a different location from USER1 computer. The computers at 164 are coupled to the Internet via Server2. USER3 and Server3 represent yet a third installation.

[162] Note that the concepts of "client" and "server," as used in this application and the industry, are very loosely defined and, in fact, are not fixed with respect to

machines or software processes executing on the machines. Typically, a server is a machine or process that is providing information to another machine or process, i.e., the "client," that requests the information. In this respect, a computer or process can be acting as a client at one point in time (because it is requesting information) and can be acting as a server at another point in time (because it is providing information). Some computers are consistently referred to as "servers" because they usually act as a repository for a large amount of information that is often requested. For example, a World Wide Web (WWW, or simply, "Web") site is often hosted by a server computer with a large storage capacity, high-speed processor and Internet link having the ability to handle many high-bandwidth communication lines.

[163] A server machine will most likely not be manually operated by a human user on a continual basis, but, instead, has software for constantly, and automatically, responding to information requests. On the other hand, some machines, such as desktop computers, are typically thought of as client machines because they are primarily used to obtain information from the Internet for a user operating the machine.

[164] Depending on the specific software executing at any point in time on these machines, the machine may actually be performing the role of a client or server, as the need may be. For example, a user's desktop computer can provide information to another desktop computer. Or a server may directly communicate with another server computer. Sometimes this is characterized as "peer-to-peer," communication. Although processes of the present invention, and the hardware executing the processes, may be characterized by language common to a discussion of the Internet (e.g., "client," "server," "peer") it should be apparent that software of the present invention can execute on any type of suitable hardware including networks other than the Internet.

[165] Although software of the present invention may be presented as a single entity, such software is readily able to be executed on multiple machines. That is, there may be multiple instances of a given software program, a single program may be executing on two or more processors in a distributed processing environment, parts of a single program may be executing on different physical machines, etc. Further, two different programs, such as a client and server program, can be executing in a single

machine, or in different machines. A single program can be operating as a client for one information transaction and as a server for a different information transaction.

USER INTERFACE:

[166] Figure 1 illustrates a screen display that allows a user to create a new portfolio and to set basic portfolio specifications. At 102 the user can enter a name for the portfolio. Defining a name can be left completely to the user's discretion or a name or partial name (i.e., default identifier) can be automatically provided. For example, if the user is working at a large analysis firm a naming convention may be used for different groups or clients. A manager can be designated for the portfolio at 104 and a description can be supplied at 106. Other basic information about the portfolio can be automatically generated. For example, the date and time of the portfolio can be stored along with the creating user, company, etc.

[167] At 108 the user can select the estimated number of instruments to be initially included in the portfolio. This information is used to estimate the required computing resources at each stage of the portfolio setup process to help the user avoid overloading existing resources. Each subsequent configuration choice has implications on processing time, so certain computationally expensive features, such as spectrum processing, may not be recommended for higher frequency prediction tasks without first ensuring that adequate computing power is available (i.e., a user wishing to predict a 5 minute timeframe would need to ensure that the processing time is within 5-30 seconds in order for the prediction to be actionable). At 110 a "Research" or "Executing" type of portfolio can be designated. If the portfolio is a Research Portfolio then it is intended for simulation to see how well the portfolio's modeling and predictions are performing. In other words, limited, or no, trading of securities will be performed using the Research Portfolio. If the portfolio is designated as an Executing Portfolio then actual trades may be automatically or manually placed as a result of the portfolio's processing. In a preferred embodiment, a Research Portfolio includes recordkeeping and diagnostic functions and data that are not necessarily used for an Executing Portfolio.

[168] A Research Portfolio can be optimized into an Executing Portfolio for faster execution. For example, a preferred embodiment of the invention is written in a computer language called "List Processing" (LISP). The LISP code or script for a Research Portfolio can be compiled into an executable version, or Executing Portfolio. A Research Portfolio will generally not be considered as important as an Executing Portfolio and the system can schedule running of research portfolio's at non-peak periods such as after markets are closed. A default scheduler will automatically process Research Portfolios at a time when Execution Portfolios are not being processed so that maximum resources are available for time sensitive processing tasks. Because the processing can be scheduled by the user for any time of day, such as in the evenings, they may maintain a large number of Research Portfolios without necessitating the procurement of additional computing resources.

[169] A business day and calendar type can be set at 112 and 114, respectively. These settings are used in time-of-day and market open and closed timings and are also important when using agents based on the local time of a market. For example, a user may reside in New York, but trade markets in London, England. In this example the portfolio's prediction tasks must be mapped to the open and closed timings of the market in London, but the user may only transact business on business days in New York, observing U.S. Holidays rather than U.K. Holidays. A currency basis is set at 116 as the units for outputting references to instrument price and portfolio holdings. A default condition is to make all references to instrument prices in the original currency denomination (i.e., if the portfolio has U.S. stocks, and European Union stocks, then the magnitude component of the predictions would be referenced in USD and Euros respectively). However, a user may select their preferred currency basis, and all references to instrument price will be in the designated currency basis. Additional settings include a time period for modeling frequencies at 118. In a preferred embodiment, a user can select one or more time intervals for modeling within any single portfolio, resulting in an output for each instrument in the portfolio, which represents a prediction of the next price in the series, at each selected frequency.

[170] For example, a user could select hourly and daily frequencies, to obtain from the system, immediately after the close of each hour and also each day, the

predictions of the instrument price at the close of the next hour and day in the series. The time-series data of an instrument, generally consisting of Open, High, Low and Close price data points, and sometimes other time-series data, is used in the modeling for each respective frequency. A user can also specify custom frequencies using Greenwich Mean Time notation. For example, a user could specify a "custom day" frequency that consists of time intervals 09:00 (9:00 am EST) through 15:30 (3:30 pm EST; 30 minutes before the market closes) in a case where they wish to obtain a prediction for the closing price of stocks listed on the NYSE for the next business day, but they wish to receive these predictions immediately after 15:30 so that they have 30 minutes before the market closes to execute trades should an adjustment be deemed desirable.

[171] When a user specifies custom frequencies, the system automatically constructs time-series data for the custom period, including, but not limited to, Open, High, Low and Close ("OHLC") price points using lower frequency time-series data. In the above example, the OHLC price points could be obtained by taking the Open of the first 30 minute interval beginning at 09:00 and the High and Low being captured by monitoring all Highs and Lows for each subsequent 30 minute interval, and finally, the Close would be obtained from the Close price of the 30 minute interval beginning at 15:00 and concluding at 15:30. Generally, most commercial data feeds will provide various frequencies of data ranging from tick data or one minute OHLC to daily OHLC. The user can save their entries and proceed to the next screen in the portfolio sert up process by activating Finish button 120.

[172] Other embodiments can use different numbers or types of settings. For example, additional settings can include selection of input data formats (e.g., designating a network data source, data format), output data file formats (e.g., Excel, plain text, etc.), custom data sources such as user-supplied data from a local file, manually typed or entered, etc.

[173] Figure 2 illustrates the Modeling Frequencies - Combiner Association / Management interface screen, where the user can specify that various modeling frequencies be combined using machine learning components provided with the system, called "Base Combiners," or other combiners provided by the user. The user can specify the Active Portfolio (i.e., operative portfolio for the current screen) at drop down menu

130, and the combiner library they wish to view at 132. Modeling frequencies that are to be combined to produce an overall prediction for the coarsest frequency specified, can be selected by checking the appropriate boxes in Available Modeling Frequencies window 144. Available Combiners window 138 lists all combiners in the specified library. The combiner symbols, icons or other identifiers are drag-and-drop enabled and can be transferred to Active Combiners – Frequency Modeling window 148.

[174] At 146, the user can enable the selected combiner for use as a Spectrum Combiner, which will cause the combiner to process data over a spectrum of time periods, with the current optimum being used for processing. The user can specify the Look Back Time Frame, or the method of determining the LBTF, for combining modeling frequencies, by selecting one of the options from the drop down menu. The user can specify that the LBTF be "Dynamically Learned" as in the example at 150. This specifies that the LBTF will be dynamically adjusted to reflect the current optimum number of periods for each individual instrument or data type being modeled (i.e., each individual modeling subject has its own evolving optimum which could potentially be different with each new predictive task). The user can establish parameters for determination of the optimal LBTF and specify other advanced features by clicking button Preferences at 136, which will open a separate interface screen that is the same or similar to the screen illustrated by Figure 7, which is described in detail later. When the preferences screen is accessed through Modeling Frequencies - Combiner Association / Management screen all preferences will be associated with this set of combiners only, and only with respects to portfolio currently being created or modified.

[175] The applicant will specify a default LBTF for combining modeling frequencies, and can make available any possible parameter through the drop down menu, or other selection means. The user can also specify that the frequency combining process be tuned towards a particular performance objective, such as "lowest volatility" prediction performance, by enabling and selecting the desired tuning objective from Learned Optimization drop down menu at 152. Dynamic Network Structure Optimization (described in detail later) can be enabled or disabled at 140, and the advanced user preferences screen can be accessed at 142. The user can conclude the actions on this screen by activating Finish button 154.

[176] Figure 3 shows an instrument selection and management screen. The user can specify whether actions performed on this screen are for the new portfolio (default) or existing portfolios, which can be selected at 174. The selection of instruments and factors creates respective groupings that can later be duplicated for tasks such as creating a test portfolio with slight modifications. The user can also import instrument and factor groups created previously for another portfolio, e.g., a research portfolio, as well as export information for any portfolio that details the contents of each grouping by using buttons 184 and 186, respectively.

[177] Predefined groups of instruments and factors can be imported from other files such as Excel (.xls) that are available on the user's hard drive or network. The user can enable select instruments to be copied to the factor group by selecting the instruments in the instrument grouping and clicking copy button 178 between the groupings. Instruments and factors can be removed from a grouping by dragging them outside of a grouping, and/or they can be transferred between groupings. A default condition is to use all instruments as factors.

[178] In Figure 3, types of data for modeling are grouped into 3 broad classes as "Equities," "Futures Contracts," and "Data" (i.e., non-instrument data), for purposes of illustration. The user is presented with sub-classes in the form of drop-down menus for selecting equities in, e.g., small, medium or large capitalization equities in U.S., European or Asian markets. Naturally, any other desired asset classes, sub-classes or categories within sub-classes, e.g., sector and industry, can be listed. A default condition would group the data types into conventional groupings in the particular field of use. Once a suitable sub-class or category is selected, the instruments in the sub-class or category are listed as, for example, at 170 of Figure 3. A user can drag-and-drop the instrument symbol, icon or other data type identifier into Active Instruments or Active Factors windows at 176 and 180, respectively. Figure 3 shows that "AA" for "AAcorp Inc." has been placed into each window. Any number of instruments or data types can be placed in the windows.

[179] Placement of instruments or other data types into the Active Instruments window specifies these instruments and data types (collectively refered to in this context as "instruments") as the targets, or subject of the modeling processes for the designated

portfolio. These target instruments have unique time-series data points, e.g., generally Open, High, Low and Close price data, Volume and Open Interest, which are then put through a user specified number of individually configured pre-processing routines (i.e., data transformations using some mathematical calculation, which is at a minimum (default) as simple as calculating the change in the value over the prior value), followed by machine learning based processing routines, resulting in a final prediction as described later. Placement of instruments or other data types into the Active Factors window allows the instruments and data to be used in subsequent modeling processes. The factors are themselves modeled, with their predictions used for learning linear and nonlinear relationships between the factors and instruments and data being modeled so that any detected relationships can be exploited by using the factors as indicators of instrument movements in further machine learning based processes. Button at 178 allows instruments selected for the Active Instruments window to be more easily copied to the Active Factors window as, for example, by selecting the items and then activating the button.

[180] A Price to Earnings Ratio ("P/E Ratio") data type is shown at 130. This data type has been placed in the Active Factors window 180 and can be used in subsequent data preprocessing routines and machine learning based processes discussed, below. As discussed above, a user could also place the P/E Ratio data type into the Active Instruments window 176 if they wish to model the P/E Ratio itself, for purposes of predicting the change in the P/E Ratio over a specified period. Doing so would cause the P/E Ratio to be one of the instrument outputs on the user interface in Figures 21 and 22.

[181] The user can also enable the system to perform an automatic selection of factors by selecting from the options available under the drop down menu at 182. These options may include applicant-provided utilities such as a correlation matrix based process with user specified selection parameters (with or without a learning mechanism),and/or third party factor selection engines, configuration of which may be provided through a separate interface screen. Any such utilities could be scheduled to re-run automatically, to mechanically re-select and thus optimize factor selection before further dynamic optimization provided by the system's machine learning processes, described in more detail below. Finish button can be activated at 188.

[182] Figure 4 shows how a user selects the machine learning components called Combiners, which are described in detail later. In a preferred embodiment, the combiners at this level represent the first occurrence of machine learning components used in the system. Combiners can be used for review of raw instrument and factor time-series data and to identify the most relevant linear and non-linear relationships between instruments and factors, based especially upon the most recent data. These observations and rankings by relationship strength are subsequently considered by the system as indicators (strategy elements formally designated as Indicators will be detailed later), which accuracies will be measured and subsequently ranked together with the outputs of the strategy elements called Indicators. In a preferred embodiment, this process is iterative, and occurs with each new predictive task. Therefore the factor selection process can be unique to each individual instrument or data type being modeled, and also potentially unique with each new predictive task.

[183] In a preferred embodiment, the 6 combiners in the Available Combiners window 208 are "base" combiners that are provided with the system. A default condition is to employ all available base combiners and use a simple average of all as a final result. Alternatively, a user can select any combination of combiners, or create and employ within the system their own combiners (machine learning based or mechanical). New base combiners can be added by the applicant as variants are developed or altogether new methods discovered. The current base combiners, and very likely any future base combiners, will remain proprietary trade secret information, with the specific details of their implementations not published unless specifically patented at some future date.

[184] Figure 5 illustrates an Indicator Library Creation and Selection screen. Indicators are mathematical functions that operate on raw instrument or other data types. Indicators represent one form of pre-processing of raw data (described later), where their output signal is used as input in subsequent modeling processes. At 240, a user can designate the indicator library, selecting from new (default) or other indicator libraries that have already been created, including the base indicator libraries included with the system, or other libraries created previously, consisting of any combination of base indicators and/or proprietary indicators supplied by the user. The grouping of both base indicators and user supplied or created indicators into library is important, because, for example, some

indicators are specifically designed for particular asset classes or investment strategies, and it may be desirable to organize them this way (e.g., stock indicators, or sell-short strategy indicators). At 242 the user can enter a name for the new library. Defining a name can be left completely to the user's discretion or a name or partial name can be automatically provided. For example, if this screen is accessed during a portfolio setup process, the automatically provided name might reference the name of the portfolio currently being created, in whole or in part.

[185] At 246, 252, 254, and 256 a user can select the format of the indicator(s) they wish to add, by checking the applicable boxes. Browse button 248 appears after an indicator format is selected. A user can activate the button to open window that will show the contents of the portion of the system directory where the indicator files of the respective file format type are located. The user can then select the desired indicator, which will close the browse window, displaying the file location/name as shown at 250. The applicant may provide, through a separate interface screen, some type of folder based organizational structure for the user to file indicators in their respective formats. For example, the user might be directed to transfer any Excel files containing indicator specifications into the folder marked "Excel," possibly through a similar browse function as at 248, except that the browse function would permit access to the user's entire hard drive or network. Other file formats can automatically be filed by the system, e.g., in cases where an indicator coded in the applicant's specified format is coded using a utility provided by the applicant. At 258 a grouping of all indicators that comprise the library being created or modified is shown. Add/Delete (+/-) buttons at 244 can be used to add any number of additional files of the respective types without exiting the current screen. Finish button 260 is used to finalize the operation and proceed to the next setup screen, or the main application screen, as applicable.

[186] Figure 6 illustrates the Indicator Association / Management screen display where a user can associate particular indicators with the instrument and/or factor groupings specified for the portfolio being created or modified as indicated, for example, at 270. At 278, all available indicators contained in the indicator library specified by the user at 272 are shown. A user can drag-and-drop the indicator name, icon or other identifier into Active Indicators-Instruments or Active Indicators-Factors windows at 284 and 290,

respectively. The example shows that the indicator called "MACD" has been placed into the instruments window, and the indicator called "EMA" has been placed into the factors window. As shown in 172 and 178, the source library name is displayed, and indicators that have been added can be designated by the user as a Spectrum Indicator by checking Enable Spectrum box, as in the examples at 282 and 288. A default condition of an enabled Spectrum Indicator (described in more detail later) would be a spectrum of parameters designated by the applicant. The user can modify any indicator parameters through a separate interface provided by the applicant, that is the same or similar to the screen illustrated by Figure 7, by clicking on the indicator name, icon or other identifier.

[187] Placement of indicators into the Active Indicators-Instruments window associates the specified indicators with the instruments previously specified by the user in the operative portfolio, causing the raw instrument data, exclusively, to be processed using the specified indicators. Similarly, placement of the indicators into the Active Indicators-Factors window associates the specified indicators with the factors previously specified by the user in the operative portfolio, causing the raw factor data, exclusively, to be processed using the specified indicators. In a preferred embodiment, the system uses indicators, which are mathematical functions that operate on raw instrument or other data types, to produce an output that represents the new value in a series, which is used as input in subsequent modeling processes. An indicator may be as simple as the calculation of the change in the raw data value from the prior raw data value, which could be expressed as $c\text{-}p=i$ where c is current raw data value, p is previous raw data value and i is new indicator value. Each asset class will have a default set of indicators specified for use in Quick Start portfolios. The use of indicators in a financial time-series application of the system represents one form of pre-processing, which is designed to help filter out the noise that is present in most data, and especially in financial time-series data, producing a signal. In evolutionary dynamical systems like financial markets, the level of noise fluctuates widely over time, and is present at different levels in differing asset classes and even in different instruments within an asset class or sub-class (i.e., sector or industry). Therefore, indicator produced signals fluctuate widely in their accuracy and usefulness over time. The applicant's system uses these signals in further machine learning based processing routines like that are able to make the best possible

use of the indicator signals by reviewing each individual indicator signal's historic and current values, detecting performance patterns and learning when each signal type is performing well, or likely to perform well, in the current market environment. The signals showing or expected to show the most strength presently are then most influential in the producing the system's final predictive output for each respective instrument or data type being modeled.

[188] At 292 and 296 the user can specify the Look Back Time Frame, or the method of determining the LBTF, for Instruments and Factors, respectively, by selecting one of the options from the drop down menu. In the example at 292, "Last n Periods" has been selected, causing the period specification field "where *n*=" to be displayed, as at 294. The user can then enter for *n*, the number of periods, or individual historical indicator values, that they want the system to use in the machine learning based portions of the modeling process. If, for example, the user specifies 500 days, the system will use the last 500 values produced by indicators associated with instruments, including the current period. If the user specifies a number of periods that exceeds the historical data available in the database for a given instrument or factor, (e.g., a newly listed stock of a newly created index included in the portfolio), the system will automatically use the maximum number of periods available, and will expand this to include new data until the specified number of periods exists (the applicant may provide an error notification pop-up window that requires user acknowledgement). The user can also specify that the LBTF for instruments and/or factors be "Dynamically Learned" as shown in the example at 296. This specifies that the LBTF will be dynamically adjusted to reflect the current optimum number of periods for each individual instrument or data type being modeled (i.e., each individual modeling subject has its own optimum which could potentially be different with each new predictive task).

[189] The user can establish parameters for determination of the optimal LBTF and specify other advanced features by activating button Preferences at 276, which will open a separate interface screen as illustrated by Figure 7, which is described in detail later. When the preferences screen is accessed through Indicator Association / Management screen all preferences will be associated with indicators only, and only with respects to portfolio currently being created or modified. The applicant will specify a default LBTF

for both instruments and factors, and can make available any possible parameter through the drop down menu, or other selection means.

[190] At 274 of Figure 6, the button Configure 1:1 provides the user with access to a separate interface screen, illustrated by Figure 8, where all strategy elements preceding the Advisor Layer (described later) can be linked on a one-to-one basis to form a strategy chain. SDL scripting interface screen can be accessed at 298. At 280, the user can export a complete list of indicators and their associations for any specified portfolio. At Finish button 300 is used to finalize the operation and proceed to the next setup screen, or the main application screen, as applicable.

[191] Figure 7 illustrates the Advanced User Preferences interface screen, which may be accessed from various other interface screens. The particular preference settings shown in Figure 7, or otherwise described, are an example only, and the applicant may make any possible system variable accessible to the user for specification. The default settings described are preliminary and are specified as examples, and the applicant may change the default settings from time to time based upon future research. The advanced preferences selected by the user are applicable to the operative portfolio only, with each portfolio having its own advanced preferences screen available at each processing layer which may contain additional or different settings, and selection mechanisms. For example, the one-to-one configurations pages provide access to an advanced user preferences page that is the same or similar to the screen illustrated in Figure 7, with a method of selecting any strategy element being worked with at that particular processing layer. The user can then specify their preferences, as applicable, for each strategy element individually.

[192] At 312, the Spectrum Parameters grouping allows the user to specify which variable set in a spectrum processed strategy element the system will use in subsequent processes, e.g., if a 5-20 day moving average is specified, the variable set is 5-20 days, or 16 possible variations generating 16 different signals which will be used as specified by the user. The default setting uses the variable set that the machine learning processes have determined to be most effective overall (i.e., most useful in contributing to final prediction accuracy) in the most recent prediction tasks, which could potentially be any of the variable sets.

[193] At 314, the Period Parameter grouping allows the user to specify various forms of period sampling parameters. The system default uses all periods within the default range rather than sampling based upon any criterion. At 318, the Look Back Time Frame grouping allows the user to specify various historical data range size parameters and sampling criterion. The default setting is to use all periods including the current period for the entire default range, e.g., 1000 periods. At 320, the Normalization – Input Data grouping allows the user to specify data transformations that raw instrument and other data types will undergo prior to any further processing by the system. At 322, the Normalization – Output Data grouping allows the user to specify data transformation operations that are to be performed on post-processing output data at each processing layer, prior to use of the data in subsequent processing. At 324, the Learning Criterion – General grouping allows the user to select various criterion for overall system performance. For example, the user may prefer to tune the system for greater accuracy in the direction of the predictions, where they may prefer the magnitude of the price move to wrong rather than the direction of the price move. The user can select the option to assign weightings to all possible tuning possibilities to achieve the desired balance. At 326, the user can activate the Restore Defaults button, and at 328, the user can activate the Save/Continue button to return to the previous screen.

[194] Figure 8 illustrates a MACD Spectrum Indicator – Configuration screen, which can be accessed by the user by double clicking the MACD indicator symbol, icon, or other identifier on any interface screen. A default condition is that when an indicator is designated by the user as a spectrum indicator, e.g., by checking the enable spectrum box when selecting the indicator for inclusion in a portfolio as in Figure 56 the default spectrum variables are used. A user can then expand any indicator into a screen that is the same or similar to Figure 8, where at 352, the user can modify all variables, specifying the Start Range, which is the beginning spectrum variable, and the End Range, which is the ending spectrum variable. The primary consideration when selecting the number of variables in a spectrum, is computing resources. Wide spectrums, especially when used with a number of indicators, can be computationally expensive. The applicant will provide through a separate interface screen a utility for monitoring and managing computing resources, so that users can avoid overreaching their current computing

capacity. This utility could provide a complete map of all computing resource consumption, so that users can eliminate bottlenecks. The applicant may also provide key resource requirements and allocation information through pop-up windows or other conspicuous means at critical points in the portfolio configuration processes. The applicant will provide basic indicator details and guidelines for each particular indicator's use via pop-up windows or other means, which can be accessed by the user by activating the selector marked Details, as at 350. In a case where the user specifies a proprietary indicator, they user can write and save within the system similar such information, if they choose.

[195] Figure 9 illustrates the One-to-One Configuration – Indicators screen, where a user can link strategy elements together on a one-to-one basis to form the first part of a strategy chain. Figure 9 is an example of the screen that can be accessed from the Indicator Association / Management screen, illustrated by Figure 6, which covers strategy elements up to the Advisor Layer. The ability to string together strategy elements on a one-to-one basis allows the user to have the maximum possible control over how strategy elements are utilized within the system. At 390, the user can access the advanced user preferences screen, which is the same or similar to the advanced user preferences screen illustrated by Figure 7, except that the applicant will provide a drop down menu where the user can select, individually, any strategy element contained in the tree currently being created or modified. The applicable preferences for each strategy element will be available to the user on this preferences screen, while inapplicable preferences will be grayed out, e.g., the data source preferences will be limited to operations specifically impacting the raw data, such as system input data transformations and period parameters. Any specifications by the user on the advanced preferences screen accessed through the One-to-One Configuration – Indicators screen (i.e., "local" settings) will override any "global" specifications from a prior preferences screen. These overrides will be applicable only to the particular strategy elements specified in the drop down menu that will be provided. For example, a change to one particular indicator will not effect other indicators in the same tree.

[196] At 360, the user can select the operative portfolio. The default is, in the order of priority, the portfolio currently being created or modified, or the most recently created or

modified portfolio. Drop down menu 362 allows the user to select the indicator library they wish to view at Available Indicators window 374. At 364, the user can designate which tree they wish to view for modification. The default tree display is New Tree which shows a basic tree structure with four placeholder strategy elements, representing data sources, instruments, factors, and indicators, which are automatically replaced when a user drags-and-drops strategy elements into their place.

[197] There are four different type of strategy elements that can be specified at the Indicator Level. At 368, the Available Data Sources window displays all data sources previously specified by the user as available on their local workstation or network. Any data source can be dragged-and-dropped into the designated positions as shown with eData at 376. Both Instruments and factors can have one or more distinctly different data sources. The factor data placeholder is shown at 378. If a data source placeholder is not filled, the system default is to use the data source specified, for example for instruments, as the data source for factors.

[198] At 370 and 372, the Available Instruments and Available Factors windows, respectively, display all of the symbols, icons or other identifiers for all data types previously specified by the user as instruments and factors in the operative portfolio. These instruments and/or factors can be dragged-and-dropped onto the instrument and factor placeholder areas, as shown at 380 with the example of the instrument AAcorp, and at 382 and 384 with the example of the factors P/E and AUX, respectively. Additional instruments and factors can be added in the same way by dropping them over the first instrument or factor added, where they will automatically be appended to the tree in the order in which they have been added.

[199] At 374, the Available Indicators window displays all of the symbols, icons or other identifiers for indicators specified in Indicator Library drop down menu 362. The default is to list all of the indicators currently included in the operative portfolio. The indicators are, by default, in the same form they are used in the portfolio, i.e., if they have been specified as spectrum indicators, they remain as such, unless the user re-specifies them. These indicators can be associated with, i.e., linked to, either instruments or factors, by being dragged-and-dropped onto the target instrument or factor, as shown at 386, where the indicator EMA was dropped onto AAcorp at 380. Additional indicators

can be added in the same way by dropping them over the same target instrument or factor, where they will automatically be appended to the tree in the order in which they have been added. At 388, the indicator MACD was associated with factor AUX at 384.

[200] The tree specified as an example in Figure 9 becomes the first part of the strategy chain and can be written in the following statements: (1) eData 376 is the data source for both instrument data AAcorp 380 and, factor data P/E 382 and AUX 384; (2) indicator EMA 386 will be used to pre-process instrument data AAcorp 380, while indicator MACD 388 will be used to pre-process factor data AUX 384; and, (3) factor data P/E 382, which has no indicator associated with it will be limited being pre-processed by Change, which is the default minimum indicator that exists for all data types (Change is the change in the data value between the prior period and the current period). Unless the user disables the use of Change as an indicator (and supplies at least one replacement indicator), a system default is to use Change in addition to any other indicators specified in by the user. Finally, to save the specifications and return to the preceding screen, the user activates Finish button 400.

[201] Figure 10 illustrates the Strategy Description Language ("SDL") Scripting interface screen, where a user can create or re-specify any type of signal generating strategy element from a simple indicator to a complex trading system with nested strategy elements and operations including if/then logic. In a preferred embodiment, SDL is an intermediate language layer between the application graphical user interface and the core Lisp application. In a preferred embodiment of the system, signal generating strategy elements can take two forms. A strategy element formula "type" refers to the generic form and function of a strategy element, i.e., a moving average constitutes a strategy element type, where the specifications are not present. In object oriented programming, these "types" might be referred to as "macros." An "instantiation" of a strategy element type, is a particular instance of a strategy element where the specifications are present, i.e., a 10 day moving average calculated on the Close, is an instantiation of a moving average. In object oriented programming, these "instantiations" might be referred to as "functions," which can include the use of macros in their processes. The important difference between SDL and basic Lisp is that the resulting objects can read, modify and coordinate the use of instrument/factor symbol values across various timeframes,

encompassing various histories, and perform various operations. For example, the user can insert into any formula type "shell," a simple syntax that refers to a specified set of values such as period options (e.g., start *n* periods back, and use *k* total periods for the calculation) for any database variables (OHL and/or C, etc.), and then the calculation/output of this newly specified object/function creates new values for the current prediction task that are stored in the database (whether these are used for further processing or accepted as the final modeling output, in the case of combiners). The applicant may create definitions that are compatible with other third-party strategy scripting products, so that users can employ strategies already created in another scripting language in the system.

[202] The user can specify, using SDL, all details including the source data feed, data points, frequencies, number of periods, period sampling starting points, list functions, mathematical operations. The applicant will provide a library of default strategy element definitions, e.g., a simple moving average and OHLC data points. An example of syntax used to specify period data are as follows:

(IBM.high); means IBM's most recent High.
(IBM.high -1); means IBM's High for one default time frame back.
(IBM.high -1 daily); means IBM's High one period back in the time frame, "Daily."
(list IBM.high -3 11); lists all values beginning -3 back for a total of 11 periods as a list of values.

[203] At 420, the user can select from a list of base formula types, e.g., moving average, which consists of default formula types supplied with they system, or new formula types defined by the user. The user can select Free Form if they prefer not to use a base formula type as a starting point. At 424, the user can specify whether the SDL formula being defined is a new type, or a new instantiation, for various broad strategy element categories, e.g., an indicator instantiation. The user can specify the library where they want the new strategy element to be saved at 426. A default condition is to permit only specific instantiations (i.e., fully specified strategy elements) to be stored in a library of already specified and operable strategy elements. Strategy element types, would be stored in broad strategy element category libraries, e.g., Advisor Types. The user can access full details for all SDL object definitions and examples for their use, through a

separate interface screen, by activating View Definitions button at 422. At 428, the SDL Formula bar displays the formula type specified at 420. The user can either place their cursor into the formula bar and make direct entries into the indicated data area within the formula shell, or, alternatively, the user can make entries into Required Fields table at 430, where the field headers for each respective formula type's required input is displayed. If the user selects Free Form as the formula type, the formula bar will be empty, and the user can enter any SDL syntax they choose. At 432, the user can enter a detailed description of the strategy element in ordinary notation (i.e., not SDL), which can later be accessed for review and/or modification. The user can save the new SDL formula by activating finish button 434.

[204] Figure 11 illustrates an Advisor Creation and Selection screen. Advisors are components within the system that serve to further pre-process signal data generated or made available at the preceding processing layer, e.g., the indicator layer, where all advisor output signal is then used as input in subsequent modeling processes, as described in more detail later. One primary difference between indicators and advisors includes their respective hierarchies within the system, i.e., the processing layer in which they operate. In theory, especially outside of the context of the applicant's system, they serve the same general purpose as indicators, i.e., they are advanced indicators. In practice, and in the context a preferred embodiment of the applicant's system, the more complex a signal generating strategy element is, the less additional processing a user would subject its output signal to. Therefore, advisors are generally more complex than indicators and would typically consist of the recommended Base Advisors (described in more detail later), and the user's proprietary or selected third-party provided models or complete trading systems (the distinction between models and trading systems by the applicant is intentional, e.g., if the user is specifying a portfolio within the system for risk management purposes vs. active trading, the term models is more appropriate than trading systems).

[205] These advisor models or trading systems can be purely mathematical functions on input data, i.e., static signal generators, or, they may have any number of desired rules or logic embedded in them, i.e., non-static signal generators, such as neural networks or genetic algorithm based systems. In a case where the user has identified very powerful,

i.e., accurate, yet simple signal generating strategy elements, the strategy element could be designated as an advisor rather than an indicator, or, the user could specify its use in both the indicator and advisor processing layers. A default condition is to utilize all base advisors, included with the system, which are machine learning based components described in more detail later. Other base advisors may be added by the applicant in the future, and the recommended default set may be re-specified.

[206] The user can specify whether the advisors will operate on raw or pre-processed data, e.g., some advisors may be designed to produce output signal from raw instrument or other raw data types, where some advisors may be designed specifically to produce output from indicator output signal data (i.e., raw data pre-processed with one or more indicators), while the user may specify that some advisors do both, resulting in two different advisor output signals that will be used in subsequent processing tasks. The user can associate advisors with particular strategy elements, creating the next section of the strategy chain, as described in more detail later. The user can also specify the output of any externally generated signal data as an advisor in the system, where the system will treat this raw data (actually the output of external advisor processing) in the same way it treats internally produced advisor output for use in subsequent processing tasks. In other words, the user can either build advisors using system utilities, they can code advisors for insertion into the system, or utilize a data feed format to bring externally processed advisor output signals into the system for use in the modeling processes.

[207] In Figure 11, at 500, a user can designate the Advisor Library, selecting from new (default, which includes all base advisors) or other advisor libraries that have already been created, including the advisor library included with the system (advisors that are included that are not part of the base set of machine learning based advisors), or other libraries created previously, consisting of any combination of base advisors and/or proprietary advisors supplied by the user. The grouping of both base advisors and user supplied or created advisors into libraries is important, because, for example, some advisors are specifically designed for particular asset classes or investment strategies, and it may be desirable to organize them this way (e.g., stock advisors, or sell-short strategy advisors). At 502 the user can enter a name for the new library. Defining a name can be left completely to the user's discretion or a name or partial name can be automatically

provided. For example, if this screen is accessed during a portfolio setup process, the automatically provided name might reference the name of the portfolio currently being created, in whole or in part.

[208] At 506, 512, 514, and 516 a user can select the format of the advisor(s) they wish to add, by checking the applicable boxes. The browse button at 508 appears after an advisor format is selected. A user can activate the button to open a window that will show the contents of the portion of the system directory where the advisor files of the respective file format type are located. The user can then select the desired advisor, which will close the browse window, displaying the file location/name as shown at 510. The applicant may provide, through a separate interface screen, some type of folder based organizational structure for the user to file indicators in their respective formats. For example, the user might be directed to transfer any Excel files containing advisor specifications into the folder marked "Excel Advisors," possibly through a similar browse function as at 508, except that the browse function would permit access to the user's entire hard drive or network. Other file formats can automatically be filed by the system, e.g., in cases where an advisor coded in the applicant's specified format is coded using a utility provided by the applicant. At 518, the example shows a New Library Content window displaying a grouping of all advisors that comprise the library being created or modified as specified at 500. Add/Delete (+/-) buttons at 504 can be used to add any number of additional files of the respective types without exiting the current screen, or delete those previously selected. Finish button 520 is used to finalize the operation and proceed to the next setup screen, or the main application screen, as applicable.

[209] Figure 12 illustrates the Advisor Association / Management screen display where a user can associate particular advisors with the instrument and/or factor groupings specified for the portfolio being created or modified as indicated, for example, as specified at 530. At 538 all available advisors contained in the Advisor Library specified by the user at 530 are shown. A default condition is to list all Base Advisors as available, regardless of which advisor library the user specifies, so in the example, only the "User Inserted Advisor" is a non-base advisor being shown as available in the library called Equities. The user can drag-and-drop the advisor symbol, icon or other identifier into Active Advisors-Instruments or Active Advisors-Factors windows at 542 and 548,

respectively. Figure 12 shows that the base advisor called "Nearest Neighbor" has been placed into the Active Advisors-Instruments window, and a user provided advisor called "User Inserted Advisor" has been placed into the Active Advisors-Factors window. As shown in 540 and 546, the advisor's source library name is displayed, and advisors that have been added can be designated as a Spectrum Advisor by checking the Enable Spectrum box, as in the examples. A default condition of an enabled Spectrum Advisor (described in more detail later) would be a spectrum of parameters designated by the applicant. The user can modify any advisor parameters through a separate interface provided by the applicant, illustrated by Figure 7, by double clicking on the advisor symbol, icon or other identifier.

[210] Placement of advisors into the Active Advisors-Instruments window associates the specified advisors with the instruments previously specified by the user in the operative portfolio, causing raw and indicator processed instrument data to be processed using the specified advisors. Similarly, placement of the advisors into the Active Advisors-Factors window associates the specified advisors with the factors previously specified by the user in the operative portfolio, causing the raw and indicator processed factor data, to be processed using the specified advisors. In a preferred embodiment, the system uses a default set of 7 base advisors, that operate on raw and indicator pre-processed instrument or other data types, to produce an output signal that represents the new value in a series of advisor output signals, which is used as input in subsequent modeling processes by combiners. The applicant may change the default set of base advisors and the data they operate on in the future.

[211] One important feature of the applicant's system is that it is domain independent, or in the case of a securities prediction embodiment, the system has no investment strategy bias. The system is, therefore, not itself a model, but rather a flexible, modular framework for managing a users own proprietary models, using a collection of advanced and uniquely implemented and deployed machine learning technologies. This is of critical importance, because the system is, because of this, suitable for any type of user, irrespective of their particular approach to the domain. In other words, by using the system the user is not adopting a new strategy, but extracting the best possible results from their existing strategy (unless they choose to create a new strategy). The 7 base

advisors that comprise the default set of advisors are listed in window 538. Six of the current set of 7 base advisors, namely Nearest Neighbor, Complex Nearest Neighbor, Decision Tree, Complex Decision Tree, Perceptron, and Multi-level Perceptron are strictly unique implementations of machine learning techniques, without any strategy bias, that may be used employed at various places within the system. The mathematical and logical underpinnings of these "unbiased" advisors (which are also used as default Base Overlay Advisors and Base Combiners, described later) do not need to be published because they do not conflict with any pre-existing user strategy bias or mandates, i.e., users do not require transparency with respects to these classifications of strategy elements any more than they require transparency into the source code used to program an interface screen.

[212] In the context of a preferred embodiment of the applicant's system, a decision tree, is informally a conditional series of "tests" that are applied to the input, where depending on the outcome of the tests (a path through the tree), a prediction is made. Given *n* periods of past data, the system uses a "minimum entropy" heuristic that attempts to approximate the smallest "explanation" of the data over that period. For example, a small decision tree might look like the following:

```
If 13mvag is > close
      if 23ema is < high
         then expect 2.2% gain next period (based on 5 samples)
           else expect 0.1% loss next period (based on 2 samples)
        else
    if up 3 days in a row explect 4.5% drop next period (one sample)
  else expect 0.5% gain (7 samples).
```

One use of a decision tree is to identify and then possibly "mimic" or "fade" what it expects other trading systems may have discovered about the current period. In the context of a securities embodiment of the system, to mimic means to accept the decision tree's explanation as it's final output, and to fade means to reject the decision tree's explanation when the pattern described is not expected to be repeated, where the output is multiplied by negative 1, with the product accepted as the decision tree's final output. The use of multi period decision trees generally applies the same algorithm to many different Look Back Time Frames, where the results are then combined with other outputs in the same processing layer. This allows for modeling of populations of traders

and other market participants across a wide range of different timeframes, creating hundreds of possible explanations or hypotheses for a single market action.

[213] A Nearest Neighbor is, informally, a component that creates a vector of the input values, and using table lookup finds the vector of values in previous periods of time that is most similar (based on a selected distance metric) and "assumes" what happened then will happen again, thus, its prediction can be said to be reasoned by analogy with past data or "case based." Usually the more periods the nearest neighbor has to consider the more reliable it will be. Variations on nearest neighbor processing include: (a) Use *k* (some number) as opposed to just 1 nearest neighbor, and average; (b) Rather than using just k previous - use all data points inversely weighted by their closeness; (c) In "complex nearest neighbor" the distance metrics are themselves learned by the system in a way that may weight different inputs; and/or, (d) Any of the above can also be multi-period where different Look Back Time Frames are considered and the nearest neighbors combined.

[214] A Perceptron is, informally, a type of neural network that attempts to learn a linear combination of its input weights to produce predictions that minimize their error. Perceptrons have some unique characteristics that make them suitable for determining market temperament: (a) since it iteratively adjusts itself, it most influenced by the most recent data, and (b) it tends to over compensate if the recent data point is misevaluated, and these characteristics are not unlike human emotion, a major factor in financial markets. Through a separate interface screen that is the same or similar to the one illustrated by Figure 7, the user can specify and add non-linear terms to single layer Perceptrons so that non-linear functions can be learned. Multi-level Perceptrons can have any number of layers and nodes, which can be specified by the user for more complex non-linear learning. In general, the system can use any type of algorithms or analysis techniques including heuristic, deterministic, non-deterministic, etc.

[215] Base advisor 7, as shown at 296, is a Short-Term Trend Advisor, which is a very simple indicator that the applicant recommends, and specifies as a default advisor. This simple advisor does contain strategy bias, and for this reason its specifications are published and made completely transparent to the user. The user may modify this advisor's parameters or, use its basic concept to create a their own proprietary

replacement. It is recommended not because it is a powerful component, but because it is useful when combined with other signals. It's specific formula/process is as follows:

Short Term Trend Indicator/Advisor (STT):

4. initialize STT to direction of last period change.
5. initialize hi to last period High
6. initialize lo to last period Low

Repeat for each period:

4. If the period Close > hi then STT = up
   hi replaced with period High
   lo replaced with period Low

   If the period Close < lo STT = down
   hi replaced with period High
   lo replaced with period Low

Using the above formula, the STT base advisor determines if the trend is up or down based on the current OHLC bar for a given frequency when its direction was changed (when its previous OHLC bar was violated). Average trend length can be calculated by counting the number of consecutive bars in a given direction.

[216] The applicant has developed other complex and powerful signal generating strategy elements that are suitable for use as advisors and Overlay Advisors, which are described in detail later. These strategy elements are not machine learning based, and they contain strategy bias of the applicant and others, e.g., in the case where the applicant has taken a published trading model and improved upon it. These "biased" base advisors may be included in the system for exploitation by those users that choose to select them for use in the modeling processes. In keeping with the governing principle of the system design and purpose, that of domain independence (i.e., zero strategy bias), any biased advisors included in the system will not be included in any default settings, except for the Quick Start portfolios, which are designed to provide demo and novice users of the system with everything they need to begin trading immediately after the instrument selection step. The applicant will organize Quick Start portfolios, and thus, the selection of strategy elements including biased advisors, by asset class, and some Quick Start portfolios will be geared towards particular strategy types within an asset class, e.g., long/short equity.

[217] Other instantiations of the system which would include biased advisors would be preconfigured or "closed systems" as might be used in a retail application where all users receive predictions from a single instantiation of the system, i.e., none of the users have the ability to chose strategy elements or tune processes, they only receive predictions for some covered group of instruments and/or other data types selected by the sponsor, e.g., a brokerage firm. Because users are not obligated to employ biased advisors and overlay advisors to get the best use out of the system, nor is the advantage gained through use of the system dependent upon their use, there is no need to provide transparency of these components. Therefore, the detailed mathematical and logical underpinnings of biased advisors and overlay advisors that are developed by the applicant will generally not be published, and will remain proprietary to the applicant.

[218] The use of unbiased base advisors in a financial time-series modeling embodiment of the system represents an additional form of data pre-processing, which is designed to provide a second order of signals that are derived from powerful machine learning technologies. The use of biased advisors also provides a second order of signals, which are derived from complex (generally more complex than indicators) models and trading systems. Generally, these types of models and/or trading systems constitute the highest order signals (the most advanced used in most financial applications, where they are used on a stand alone basis by market participants to perform systematic trading or risk management tasks, or as decision support tools with final decisions made at the discretion of a human user. Many of these models and trading systems are used by large populations of market participants, perhaps with only slight variations in their parameters. This is one reason why most such models and systems seem to degrade over time, becoming less useful or, frequently, less consistent. This pattern cancellation effect resulting from widespread use of low complexity, non-adaptive (i.e., parametric or rules based) models or systems, has been acknowledged by academics and practitioners alike. Therefore, signals such as those produced by biased advisors fluctuate widely in their accuracy and usefulness over time.

[219] In evolutionary dynamical systems like financial markets, better results can be achieved by using these types of signals as inputs in a far more complex system that is sophisticated and powerful enough to detect and exploit market activity resulting from

widespread use of these simple systems, and even systems of the same or similar complexity to itself. The applicant's system uses advisor signals in further machine learning based processing routines (described later) that are able to make the best possible use of the advisor signals by reviewing each individual advisor signal's historic and current values, detecting performance patterns and learning when each signal type is performing well in the current market environment. The signals showing the most strength presently or expected to show the most strength are then most influential in producing the system's final predictive output for each respective instrument or data type being modeled.

[220] At 550 and 552 the user can specify the Look Back Time Frame, or the method of determining the LBTF, for Instruments and Factors, respectively, by selecting one of the options from the drop down menu. In the example at 552, "Last n Periods" has been selected, causing the period specification field "where *n*=" to be displayed, as at 554. The user can then enter for *n*, the number of periods, or individual historical indicator values, that they want the system to use in the machine learning based portions of the modeling process. If, for example, the user specifies 500 days, the system will use the last 500 values produced by indicators associated with instruments, including the current period. If the user specifies a number of periods that exceeds the historical data available in the database for a given instrument or factor, (e.g., a newly listed stock of a newly created index included in the portfolio), the system will automatically use the maximum number of periods available, and will expand this to include new data until the specified number of periods exists (the applicant may provide an error notification pop-up window that requires user acknowledgement). The user can also specify that the LBTF for instruments and/or factors be "Dynamically Learned" as in the example at 308. This specifies that the LBTF will be dynamically adjusted to reflect the current optimum number of periods for each individual instrument or data type being modeled (i.e., each individual modeling subject has its own optimum which could potentially be different with each new predictive task). In general, the system will dynamically seek the largest LBTF that performs well, narrowing the LBTF when market conditions appear to be less predictable.

[221] At 556 and 558, the Learned Optimization for Instruments and Learned Optimization for Factors drop down menus, respectively, allow the user to specify their preferred optimization parameters based upon narrow tuning objectives such as Lowest Volatility, Lowest Drawdown and Highest Profit. As the names indicate, the Learned Optimization for Instruments operate only on advisors associated with the instrument grouping, and the Learned Optimization for Factors operate only on advisors associated with the factor grouping. These settings operate only on the base advisors that are machine learning components of the system, with subsequent machine learning components, such as Overlay Advisors and Combiners, having their own respective learning optimization controls. On the advanced Preferences screens there is a Learning Criterion-General window, as illustrated at 324 on Figure 7, that allows users to fine tune the learning optimization at each processing layer through more specific parameters, such as Maximize Correct % (direction), Maximize Correct % (magnitude), Minimize Consecutive Error (direction), and Minimize Consecutive Error (magnitude). The user can weight these preferences to create a balance that suits their needs for the portfolio currently being created or modified.

[222] The user can establish advanced parameters for determination of the optimal LBTF and specify other advanced features by clicking button Preferences at 536, which will open a separate interface screen that is the same or similar to Figure 7, except that all preferences will be associated with advisors only, and only with respects to portfolio currently being created or modified. Any specifications by the user on the advanced preferences screen accessed through the Advisor Association / Management screen will override any specifications from a prior advisor preferences screen for the operative portfolio. The applicant will specify a default LBTF for both instruments and factors, and can make available any possible parameter through the drop down menu, or other selection means. The user can also specify for each base and non-based advisor whether they will process raw data and/or indicator output signals.

[223] At 534 of Figure 12, the button Configure 1:1 provides the user with access to a separate interface screen, illustrated by Figure 13 where all strategy elements preceding the Overlay Advisor Layer (described later) can be linked on a one-to-one basis to form a strategy chain.

[224] Figure 13 illustrates the One-to-One Configuration - Advisors interface screen that is accessed through Configuration 1:1 button 534 on the Advisor Association / Management interface screen illustrated by Figure 12. Through this interface screen, the user can add to previous trees to create a strategy chain that covers strategy elements up to the Overlay Advisor Layer. The ability to string together strategy elements on a one-to-one basis allows the user to have the maximum possible control over how strategy elements are utilized within the system. At 590, the user can access the Preferences screen which is the same or similar to the advanced user preferences screen illustrated by Figure 7, except that the applicant will provide a drop down menu where the user can select, individually, any strategy element contained in the tree currently being created or modified. The applicable preferences for each strategy element will be available to the user on this preferences screen, while inapplicable preferences will be grayed out, e.g., if advisors are selected from the drop-down menu, the available preferences will be limited to operations specifically impacting the advisors. Any specifications by the user on the advanced preferences screen accessed through the One-to-One Configuration – Advisors screen will override any specifications from a prior preferences screen. These overrides will be applicable only to the particular strategy elements specified in the drop down menu that will be provided. For example, a change to one particular advisor will not effect other advisors in the same tree.

[225] At 580, the user can select the operative portfolio. The default is, in the order of priority, the portfolio currently being created or modified, or the most recently created or modified portfolio. At 582, the user can specify the active indicator tree. The default is to show the indicator tree most recently created that is not yet associated with any advisors. Drop down menu 584 allows the user to select the Advisor Library they wish to view in the Available Advisors window at 592. At 586, the user can designate which tree they wish to view for modification. The default tree display is New Tree which shows a basic tree structure with placeholder Advisor Area icons shown at each possible connection point, as at 600 and 602, which are automatically replaced when a user drags-and-drops advisors into their place.

[226] There are 2 different types of advisors that can be specified at the Advisor Level, both base and user created/provided, i.e., unbiased and biased. At 592, the Available

Advisors window displays all base advisors as well as all non-base advisors contained in the advisor library specified by the user. Any advisor can be dragged-and-dropped into the designated positions as shown in the example at 594, 596 and 598. Advisors may be removed by dragging them outside of the tree area. The Advisor Area placeholder at 602 is shown as empty for purposes of illustration only. In a preferred embodiment, a default condition is that all strategy elements from the preceding processing layer, e.g., the indicator processing layer, are operated on by all base advisors and any user specified advisors. For example, the branch at 600 shows the actual default condition that a user will see upon first accessing this screen, where All Base Advisors is the first advisor layer strategy element (i.e., MACD further up the branch is part of the indicator layer), followed by an advisor placeholder area, that may or may not be filled by the user. When the user removes the default symbol, icon or other identifier that represents All Base Advisors, an advisor placeholder area is displayed, as in the example at 602. When the preceding strategy element is not associated with any advisor then the preceding strategy element's signal data, e.g., indicator output signals, will be advanced in the database to the fields where the advisor data would have been stored, and will be operated upon by combiners as if it were advisor output signal data, to arrive at the consensus signal used as input for the next processing layer. The advisors are, by default, in the same form they are used in the portfolio, i.e., if they have been specified as spectrum advisors, they remain as such, unless the user re-specifies them.

[227] As shown in the example at 596, additional advisors can be added by dropping them over another advisor on a branch, where they will automatically be appended to the tree in the order in which they have been added.

[228] The advisor layer portion of the tree is now linked or appended to the indicator portion of the tree specified in the example, complete the first and second part of the strategy chain and can be written in the following statements: (1) advisor User Advisor 594, will process instrument data AA's indicator output Change (AA has no other indicator associated with it), (2) advisors DTree and CDTree at 201 will process indicator EMA's output for instrument AA, (3) advisor NNeig at 598 will process factor data P/E's indicator output Change (P/E has no other indicator associated with it), and (4) advisors All Base at 600 will process indicator MACD's output of factor data AUX and no

advisors will process AUX's indicator Change. Finally, to save the specifications and return to the preceding screen, the user activates Finish button 604.

[229] Figure 14 illustrates an Overlay Advisor Creation and Selection screen. Overlay Advisors are components within the system that serve to further pre-process signal data generated or made available at the preceding processing layer, e.g., the indicator layer, where all advisor output signal data is then used as input in subsequent modeling processes, as described in more detail later. One primary difference between advisors and overlay advisors includes their respective hierarchies within the system, i.e., the processing layer in which they operate. In theory, especially outside of the context of the applicant's system, they serve the same general purpose. In practice, and in the context of the applicant's system, the more complex a signal generating strategy element is, the less additional processing a user would subject its output to. Therefore, overlay advisors are generally more complex than advisors and would typically consist of the recommended Base Advisors (described in more detail later), and the user's most advanced proprietary or selected third-party provided models or complete trading systems (the distinction between models and trading systems by the applicant is intentional, e.g., if the user is specifying a portfolio within the system for risk management purposes vs. active trading, the term models is more appropriate than trading systems). Because overlay advisor output signals receive the least amount of processing prior to the system's generation of a final prediction, overlay advisors can, when they are very effective on a stand-alone basis, have greater influence on the final prediction. While the applicant has determined that an overlay advisor processing layer is generally advantageous, and has specified that a default condition of a preferred embodiment of the system recommends their use, the use of an overlay advisor processing layer is not a requirement, and the advantages gained from using the system are not dependent upon the use of overlay advisors. Each user's unique strategy, objectives and independent proprietary research using the system will govern whether overlay advisors are used for a given portfolio.

[230] These overlay advisor models or trading systems can be purely mathematical functions of input data, i.e., static signal generators, or, they may have any number of desired rules or logic embedded in them, i.e., non-static signal generators, such as neural

networks or genetic algorithm based systems. In a case where the user has identified very powerful, i.e., accurate, signal generating strategy elements, the strategy element could be designated as an overlay advisor rather than an advisor, or, the user could specify its use in both the advisor and overlay advisor processing layers (the use could also use the same strategy element in the indicator processing layer, if desired). A default condition is to utilize all base overlay advisors, included with the system, which are machine learning based components described in more detail later. Other base overlay advisors may be added by the applicant in the future, and the default set may be re-specified.

[231] The user can specify whether the overlay advisors will operate on raw or pre-processed data, e.g., some overlay advisors may be designed to produce output signals from raw instrument or other raw data types, where some advisors may be designed specifically to produce output signals from indicator or advisor output signal data (i.e., raw data pre-processed with one or more indicators or advisors), while the user may specify that some overlay advisors do all three, resulting in three different overlay advisor output signals that will be used in subsequent processing tasks, by Combiners, which are described in detail later. The user can associate overlay advisors with particular strategy elements, creating the next section of the strategy chain, as described in more detail later. The user can also specify the output of any externally generated signal data as an overlay advisor in the system, where the system will treat this raw data (actually the output signals of external overlay advisor processing) in the same way it treats internally produced overlay advisor output signals for use in subsequent processing tasks. In other words, the user can either build overlay advisors using system utilities, they can code overlay advisors for insertion into the system, or utilize a data feed format to bring externally processed overlay advisor output signals into the system for use in the modeling processes.

[232] In Figure 14, at 630, a user can designate the Overlay Advisor Library, selecting from new (default) or other overlay advisor libraries that have already been created, including the base overlay advisor library included with the system, or other libraries created previously, consisting of any combination of base overlay advisors and/or proprietary overlay advisors supplied by the user. The grouping of both base indicators

and user supplied or created overlay advisors into libraries is important, because, for example, some overlay advisors are specifically designed for particular asset classes or investment strategies, and it may be desirable to organize them this way (e.g., stock overlay advisors, or sell-short strategy overlay advisors). At 632 the user can enter a name for the new library. Defining a name can be left completely to the user's discretion or a name or partial name can be automatically provided. For example, if this screen is accessed during a portfolio setup process, the automatically provided name might reference the name of the portfolio currently being created.

[233] At 636, 642, 644, and 646 a user can select the format of the overlay advisor(s) they wish to add, by checking the applicable boxes. The browse button at 638 appears after an overlay advisor format is selected. A user can click on the button to open window that will show the contents of the portion of the system directory where the overlay advisor files of the respective file format type are located. The user can then select the desired overlay advisor, which will close the browse window, displaying the file location/name as shown at 640. The applicant may provide, through a separate interface screen, some type of folder based organizational structure for the user to file indicators in their respective formats. For example, the user might be directed to transfer any Excel files containing overlay advisor specifications into the folder marked "Excel Advisors," possibly through a similar browse function as at 638, except that the browse function might permit access to the user's entire hard drive or network. Other file formats can automatically be filed by the system, e.g., in cases where an overlay advisor coded in the applicant's specified format is coded using a utility provided by the applicant. At 648, the example shows a New Library Content window displaying a grouping of all overlay advisors that comprise the library being created or modified as specified at 630. Add/Delete (+/-) buttons at 634 can be used to add any number of additional files of the respective types without exiting the current screen, or delete those previously selected. Finish button 340 is used to finalize the operation and proceed to the next setup screen, or the main application screen, as applicable.

[234] Figure 15 illustrates the Overlay Advisor Association / Management screen display where a user can associate particular overlay advisors with the instrument and/or factor groupings specified for the portfolio being created or modified as indicated, for

example at 660. At 668 all available overlay advisors contained in the Overlay Advisor Library specified by the user at 660 are shown. A default condition is to list all Base Overlay Advisors as available, regardless of which overlay advisor library the user specifies, so in the example, only the "User Inserted Overlay" is a non-base overlay advisor being shown as available in the library called Equities. The user can drag-and-drop the overlay advisor symbol, icon or other identifier into Active Overlay Advisors-Instruments or Active Overlay Advisors-Factors windows at 672 and 678, respectively. Figure 15 shows that the base overlay advisor called "Nearest Neighbor" has been placed into the Active Advisors-Instruments window, and a user provided advisor called "User Inserted Overlay" has been placed into the Active Advisors-Factors window. As shown in 670and 676, the overlay advisor's source library name is displayed, and overlay advisors that have been added can be designated as a Spectrum Overlay Advisor (where applicable) by checking the Enable Spectrum box, as in the examples. A default condition of an enabled Spectrum Overlay Advisor (described in more detail later) would be a spectrum of parameters designated by the applicant. The user can modify any overlay advisor parameters through a separate interface provided by the applicant, illustrated by Figure 7, by double clicking on the overlay advisor symbol, icon or other identifier. The SDL formula editor window not shown in Figure 7 will also be included.

[235] Placement of overlay advisors into the Active Overlay Advisors-Instruments window associates the specified overlay advisors with the instruments previously specified by the user in the operative portfolio, causing raw and indicator and advisor processed instrument data to be processed using the specified overlay advisors. Similarly, placement of the overlay advisors into the Active Overlay Advisors-Factors window associates the specified overlay advisors with the factors previously specified by the user in the operative portfolio, causing the raw and indicator processed factor data, to be processed using the specified overlay advisors. In a preferred embodiment, the system uses as a default set of 7 base overlay advisors (6 of which are unbiased), that operate on raw and indicator and advisor pre-processed instrument or other data types, to produce an output signal that represents the new value in a series of overlay advisor output signals, which is used as input in subsequent modeling processes by Combiners. These default base overlay advisors are the same or similar to the default base advisors

described earlier, and the 6 unbiased components' mathematical underpinnings are not disclosed because they are strictly machine learning mechanisms without any strategy bias. The applicant may change the default set of base overlay advisors and the data they operate on in the future. As discussed previously, the applicant has created "biased" strategy elements that are complex and powerful enough to be suitable for use as overlay advisors. These biased strategy elements are included in default sets of overlay advisors for Quick Start portfolios only, and are not required.

[236] The use of unbiased base overlay advisors in a financial time-series modeling embodiment of the system represents an additional form of data pre-processing, which is designed to provide a third order of signals that are derived from a blend of powerful machine learning technologies. The use of biased overlay advisors also provides a third order of signals, which are derived from complex (generally more complex than advisors) models and trading systems that can then be combined with outputs of unbiased overlay advisors to arrive at a final prediction. Generally, these types of models and/or trading systems constitute the very highest order signals (i.e., the most advanced used by the most sophisticated financial applications, where they are used on a stand alone basis by market participants, to perform systematic trading or risk management tasks, or as decision support tools with final decisions made at the discretion of a human user. While many of these models and trading systems are used by smaller populations of market participants than less complex systems such as the advisors described previously (e.g., the most sophisticated hedge funds and banks with staffs of Ph.D.s), they may also be employed with only slight variations in their parameters from firm to firm. Thus, as the number of firms that can afford to conduct research and development at this level grows, or, as an example, as the techniques they are using become popularized through offerings made by software or other sophisticated vendors, these models and systems may also suffer from diminished usefulness. Even the most complex proprietary user models and trading systems can benefit substantially through use within the applicant's system. For example, a user with more than one very effective complex model or trading system can combine their outputs and get the best use of each of them without having to make subjective judgments to determine which to use in a given instance (e.g., model switching in high/low volatility environments). Additionally, a user can add strategy

elements, such as fundamental data or even an individual trader or risk manager's discretionary viewpoint, that ordinarily could not be coupled with their models or trading systems in a systematic and dynamically adaptive way (i.e., used to the extent it is helpful). The effectiveness of a single, very complex model or trading system can also be improved by the applicant's system, e.g., by using the system to learn when it is most likely to produce a correct signal. In other words, the applicant's system might assign a Confidence Level (described in more detail later) that corresponds to a low probability of correctness, thereby keeping the user out of the market (or spurring another action such as reduction of position size or the increase of a hedge position), resulting in avoidance or reduction of a loss. Even a user's proprietary machine learning based (e.g., neural networks and genetic algorithms) can benefit as just described, with the applicant's system adding additional stability to the user's existing nonparametric model. The applicant's system is sophisticated and adaptive enough to detect and exploit market activity resulting from widespread use of these complex models and trading systems, and even systems of the same or similar complexity to itself, ensuring continuing benefits to the user.

[237] The applicant's system uses overlay advisor output signals in further machine learning based processing routines performed by Combiners (described later) that are able to make the best possible use of the overlay advisor signals by reviewing each individual advisor signal's historic and current values, detecting performance patterns and learning when each signal type is performing well or likely to perform well in the current market environment. The signals showing the most strength presently or expected to show the most strength are then most influential in the producing the system's final predictive output for each respective instrument or data type being modeled. At 680 and 684 the user can specify the Look Back Time Frame, or the method of determining the LBTF, for Instruments and Factors, respectively, by selecting one of the options from the drop down menu. In the example at 680, "Last n Periods" has been selected, causing the period specification field "where *n*=" to be displayed, as at 682. The user can then enter for *n*, the number of periods, or individual historical overlay advisor values, that they want the system to use in the machine learning based portions of the modeling processes. If, for example, the user specifies 500 days at 682, the system

will use the last 500 values produced by overlay advisors associated with instruments, including the current period. If the user specifies a number of periods that exceeds the historical data available in the database for a given instrument or factor, (e.g., a newly listed stock of a newly created index included in the portfolio), the system will automatically use the maximum number of periods available, and will expand this to include new data until the specified number of periods exists (the applicant may provide an error notification pop-up window that requires user acknowledgement). The user can also specify that the LBTF for instruments and/or factors be "Dynamically Learned" as in the example at 684. This specifies that the LBTF will be dynamically adjusted to reflect the current optimum number of periods for each individual instrument or data type being modeled (i.e., each individual modeling subject has its own optimum which could potentially be different with each new predictive task).

[238] At 686 and 688, the Learned Optimization for Instruments and Learned Optimization for Factors drop down menus, respectively, allow the user to specify their preferred optimization parameters based upon narrow tuning objectives such as Lowest Volatility, Lowest Drawdown and Highest Profit. As the names indicate, the Learned Optimization for Instruments operate only on advisors associated with the instrument grouping, and the Learned Optimization for Factors operate only on advisors associated with the factor grouping. These settings operate only on the base advisors that are machine learning components of the system, with subsequent machine learning components, such as Combiners, having their own respective learning optimization controls. On the advanced Preferences screens there is a Learning Criterion-General window, as illustrated at 324 on Figure 7, that allows users to fine-tune the learning optimization at each processing layer through more specific parameters, such as Maximize Correct % (direction), Maximize Correct % (magnitude), Minimize Consecutive Error (direction), and Minimize Consecutive Error (magnitude). The user can weight these preferences to create a balance that suits their needs for the portfolio currently being created or modified.

[239] The user can establish advanced parameters for determination of the optimal LBTF and specify other advanced features by clicking button Preferences at 590, which will open a separate interface screen that is the same or similar to Figure 7, except that all

preferences will be associated with overlay advisors only, and only with respects to portfolio currently being created or modified. Any specifications by the user on the advanced preferences screen accessed through the Overlay Advisor Association / Management screen will override any specifications from a prior overlay advisor preferences screen for the operative portfolio. The applicant will specify a default LBTF for both instruments and factors, and can make available any possible parameter through the drop down menu, or other selection means.

[240] At 664 of Figure 15, the button Configure 1:1 provides the user with access to a separate interface screen, illustrated by Figure 15, where all strategy elements preceding the Overlay Advisor Layer (described later) can be linked on a one-to-one basis to form a strategy chain. The user can activate Finish button at 690.

[241] Figure 16 illustrates the One-to-One Configuration – Overlay Advisors interface screen that is accessed through Configuration 1:1 button 664 on the Overlay Advisor Association / Management interface screen illustrated by Figure 15. Through this interface screen, the user can add to previous trees to create a strategy chain that covers strategy elements up through the Overlay Advisor Layer. The ability to string together strategy elements on a one-to-one basis allows the user to have the maximum possible control over how strategy elements are utilized within the system. At 720, the user can access the Preferences screen which is the same or similar to the advanced user preferences screen illustrated by Figure 7, except that the applicant will provide a drop down menu where the user can select, individually, any strategy element contained in the tree currently being created or modified. The applicable preferences for each strategy element will be available to the user on this preferences screen, while inapplicable preferences will be grayed out, e.g., if advisors are selected from the drop-down menu, the available preferences will be limited to operations specifically impacting the advisors. Any specifications by the user on the advanced preferences screen accessed through the One-to-One Configuration – Overlay Advisors screen will override any specifications from a prior preferences screen. These overrides will be applicable only to the particular strategy elements specified in the drop down menu that will be provided. For example, a change to one particular advisor will not effect other advisors in the same tree.

[242] At 710, the user can select the operative portfolio. The default is, in the order of priority, the portfolio currently being created or modified, or the most recently created or modified portfolio. Drop down menu 714 allows the user to select the Overlay Advisor Library they wish to view in the Available Overlay Advisors window at 722. At 716, the user can designate which tree they wish to view for modification. The default tree display is New Tree which shows a basic tree structure with placeholder Overlay Advisor Area icons shown at each possible connection point, as at 728 and 732, which are automatically replaced when a user drags-and-drops advisors into their place.

[243] There are 2 different types of overlay advisors that can be specified at the Overlay Advisor Level, both base and user created/provided, i.e., unbiased and biased. At 220, the Available Overlay Advisors window displays all base overlay advisors as well as all non-base overlay advisors contained in the advisor library specified by the user. Any overlay advisor can be dragged-and-dropped into the designated positions as shown in the example at 724, 726 and 730. Overlay advisors may be removed by dragging them outside of the tree area. The Overlay Advisor Area placeholders at 728 and 732 are shown as empty for purposes of illustration only. In a preferred embodiment, a default condition is that all strategy elements from the preceding processing layer, e.g., the advisor processing layer, are operated on by all base overlay advisors and any user specified overlay advisors. For example, the branch at 730 shows the actual default condition that a user will see upon first accessing this screen, where All Base Overlay Advisors is the first overlay advisor layer strategy element (i.e., MACD further up the branch is part of the indicator layer), followed by an advisor placeholder area, that may or may not be filled by the user (not filled in the example). When the user removes the default symbol, icon or other identifier that represents All Base Overlay Advisors, an overlay advisor placeholder area is displayed, as in the example at 728 and 732. When the preceding strategy element is not associated with any overlay advisor then the preceding strategy element's signal data, e.g., indicator or advisor output signals, will be advanced in the database to the fields where the overlay advisor data would have been stored, and will be operated upon by combiners as if it were advisor output signal data, to arrive at the consensus signal used as input for the next processing layer. The overlay advisors are, by default, in the same form they are used in the portfolio, i.e., if they have

been specified as spectrum overlay advisors, they remain as such, unless the user re-specifies them.

[244] As shown in the example at 201, additional advisors can be added by dropping them over another advisor on a branch, where they will automatically be appended to the tree in the order in which they have been added.

[245] The indicator and advisor portions of the tree is now linked or appended to the overlay advisor portion of the tree specified in the example, which completes the first, second and third part of the strategy chain (the entire recommended default strategy chain) and can be written in the following statements: (1) User Overlay Advisor 724 will process the User Advisor output, (2) Overlay Advisor Decision Tree 726 will process advisor level Decision Tree and Complex Decision Tree outputs, and (3) Overlay Advisor Average of All 730 will process MACD output. Finally, to save the specifications and return to the preceding screen, the user activates Finish button 734.

[246] In Figure 17, at 750, a user can designate the Combiner Library, selecting from new (default) or other combiner libraries that have already been created, including the base combiner library included with the system, or other libraries created previously, consisting of any combination of base combiners and/or proprietary combiners created or supplied by the user. The grouping of both base combiners and user supplied or created combiners into libraries is important, because, for example, some combiners can be specifically designed for particular asset classes or investment strategy, and it may be desirable to organize them this way (e.g., stock combiners, or sell-short strategy combiners). At 752 the user can enter a name for the new library. Defining a name can be left completely to the user's discretion or a name or partial name can be automatically provided. For example, if this screen is accessed during a portfolio setup process, the automatically provided name might reference the name of the portfolio currently being created, in whole or in part.

[247] At 756, 762, 764, and 766 a user can select the format of the combiner(s) they wish to add, by checking the applicable boxes. The browse button at 758 appears after an overlay advisor format is selected. A user can click on the button to open window that will show the contents of the portion of the system directory where the overlay advisor files of the respective file format type are located. The user can then select the desired

overlay advisor, which will close the browse window, displaying the file location/name as shown at 760. The applicant may provide, through a separate interface screen, some type of folder based organizational structure for the user to file indicators in their respective formats. For example, the user might be directed to transfer any Excel files containing overlay advisor specifications into the folder marked "Excel Advisors," possibly through a similar browse function as at 758, except that the browse function might permit access to the user's entire hard drive or network. Other file formats can automatically be filed by the system, e.g., in cases where an overlay advisor coded in the applicant's specified format is coded using a utility provided by the applicant. At 768, the example shows a New Library Content window displaying a grouping of all combiners that comprise the library being created or modified as specified at 750. Add/Delete buttons (+/-) at 754 can be used to add any number of additional files of the respective types without exiting the current screen, or delete those previously selected. Finish button 770 is used to finalize the operation and proceed to the next setup screen, or the main application screen, as applicable.

[248] Figure 18 illustrates the Combiner Association / Management screen display where a user can associate particular combiners with the indicator, advisor and overlay advisor processing layers for the portfolio being created or modified as indicated, for example, as specified at 780. At 788 all available combiners contained in the Combiner Library specified by the user at 780 are shown. A default condition is to list all Base Combiners as available, regardless of which overlay advisor library the user specifies, so in the example, the "User Inserted Combiner" is the only non-base combiner being shown as available in the library called Equities. The user can drag-and-drop the combiner symbol, icon or other identifier into Active Combiner-Level 1 (indicator processing layer) window at 806, Active Combiner-Level 2 window (advisor processing layer) at 810, and/or the Active Combiners-Level 3 window (overlay advisor processing layer) at 814. Figure 15 shows that the base combiner called "Nearest Neighbor" has been placed into the Active Combiners-Level 1 window, a base combiner called "Decision Tree" has been placed into the Active Combiners-Level 2 window, and a combiner called "Average of All" has been has been placed into the Active Combiners-Level 3 window. The combiner called Average of All is not itself an actual combiner, but rather, a short cut

feature for specifying the inclusion of all base combiners at any processing layer, and deriving the final prediction from the simple mathematical average of all combiner outputs. If the Average of All feature is used, then any non-base combiner added by the user to the same Active Combiner window will be included in the calculation, and equally weighted in its contribution to the final prediction. A default condition of a preferred embodiment of the system is to use all base combiners at all processing layers and derive the final prediction at each processing layer from a simple mathematical average of all combiner outputs, where the final prediction of the last processing layer is the prediction received by a human user through an output interface screen.

[249] As shown in 804, 808, and 812, the combiner's source library name is displayed, and combiners that have been added can be designated as a Spectrum Combiners by checking the Enable Spectrum box. A default condition of an enabled Spectrum Combiner (described in more detail later) would be a spectrum of parameters designated by the applicant. Spectrum Combiners could process a spectrum of Look Back Time Frames, and/or can process a spectrum of other parameters, e.g., a spectrum of non-linear terms in a perceptron. The user can modify any combiner parameters through a separate interface provided by the applicant, illustrated by Figure 7 or 10 by double clicking on the combiner symbol, icon or other identifier.

[250] Placement of combiners into the Active Combiners-Level 1 window associates the specified combiners with the indicators previously specified by the user in the operative portfolio, causing indicator processed instrument, factor and other data types (i.e., indicator output signals) to be processed using the specified combiners. Each individual instrument or other data type being modeled is pre-processed by a minimum of one indicator (the default minimum indicator called Change, which is simply the calculated change in the value of the current period over the prior period), and each new indicator value is used to produce its own prediction generated by each respective combiner. In other words, if there is (a) one instrument being modeled (as opposed to a data type with a single data point), and (b) the default Open, High, Low, and Close raw data points are specified for inclusion (i.e., the user didn't exclude any default data points), and (c) there is only the minimum default indicator Change used (i.e., the user didn't specify any other indicators), and (d) all 6 default combiners are being used (i.e.,

the user didn't exclude any default combiners or specify any non-base combiners), then there are 24 different signals (one for each data point and indicator pairing (4 data points multiplied by1 indicator in the example) multiplied by one for each combiner (6 in the example)). These 4 data points, each with one indicator in the example, have 6 predictions each, which are then distilled into a single prediction per indicator by using a Consensus Function. The default Consensus Function is to calculate the simple mathematical average of all 6 combiner predictions for each indicator, which produces a new value that is operated on by advisors at the next processing layer. The user can specify other Level 1 Consensus Functions through the advanced Preferences button at 786, such as custom weighting of the base combiner outputs.

[251] Placement of combiners into the Active Combiners-Level 2 window associates the specified combiners with the advisors previously specified by the user in the operative portfolio, causing advisor processed instrument, factor and other data types (i.e., advisor output signals) to be processed using the specified advisor layer combiners. After each individual instrument or other data type being modeled has passed through the indicator processing layer, each new indicator layer combiner value is processed using the default set of advisors (if not excluded by the user) and any other user specified advisors. For example, if there is (a) one instrument being modeled, and (b) only the default set of 5 base advisors is used (i.e., the user didn't specify any other advisors), and (c) all 6 default advisor layer base combiners are being used (i.e., the user didn't exclude any default combiners or specify any non-base combiners), then there are 6 different signals (one for each indicator layer combiner (1 data point from the indicator layer combiner consensus, multiplied by 5 base advisor outputs in the example) multiplied by 1 for each combiner (6 in the example) = 30). These 5 advisor outputs, have 6 different combiner predictions each, which are then distilled into a single prediction per advisor by using a Consensus Function. . The default Consensus Function is to calculate the simple mathematical average of all 6 combiner predictions for each advisor, which produces a new value that is operated on by overlay advisors at the next processing layer. Adding either a single additional instrument or a single additional advisor increases the number of values that are produced and processed by 6 values. Enabling the single advisor as a spectrum, multiplies the number of values that are produced and processed (30 in the

example) by the number of different period variables in the spectrum, e.g., a 2-10 day spectrum specifies 9 different variations, or 9 multiplied by 24.

[252] Placement of combiners into the Active Combiners-Level 3 window associates the specified combiners with the overlay advisors previously specified by the user in the operative portfolio, causing overlay advisor processed instrument, factor and other data types (i.e., overlay advisor output signals), to be processed using the specified overlay advisor layer combiners. After each individual instrument or other data type being modeled has passed through the advisor processing layer, each new advisor layer combiner value is processed using the default set of overlay advisors (if not excluded by the user) and any other user specified overlay advisors. For example, if there is (a) one instrument being modeled, and (b) only the default set of 6 base overlay advisors is used (i.e., the user didn't specify any other overlay advisors), and (c) all 6 default overlay advisor layer base combiners are being used (i.e., the user didn't exclude any default combiners or specify any non-base combiners), then there are 6 different signals (one for each advisor layer combiner (1 data point from the advisor layer combiner consensus, multiplied by 6 base overlay advisor outputs in the example) multiplied by 1 for each combiner (6 in the example) = 36). These 6 overlay advisor outputs, have 6 different combiner predictions each, which are then distilled into a single prediction per advisor by using a Consensus Function. The default Consensus Function is to calculate the simple mathematical average of all 6 combiner predictions for each overlay advisor, which produces a new value that is the final prediction published to the interface screen illustrated by Figures 16 and 17 (theoretically, the user may specify another processing layer using add/delete control at 420, where these new values will be operated on by level 4 overlay advisors). Adding either a single additional instrument or a single additional advisor or overlay advisor increases the number of values that are produced and processed by 6 values. Enabling a single overlay advisor as a spectrum, multiplies the number of values that are produced and processed (30 in the example) by the number of different period variables in the spectrum, e.g., a 2-10 day spectrum specifies 9 different variations, or 9 multiplied by 24.

[253] The 6 base combiners that comprise the default set of advisors are listed in window 296. All 6 of the current set of 6 base combiners namely Nearest Neighbor,

Complex Nearest Neighbor, Decision Tree, Complex Decision Tree, Perceptron, and Multi-level Perceptron are strictly unique implementations of machine learning techniques, without any strategy bias. The mathematical and logical underpinnings of these "unbiased" machine learning components (which are also used as default Advisors, Overlay Advisors and Combiners) do not need to be published because they do not have any intrinsic strategy bias. In other words, they are strictly speaking, technology, not trading or risk management models.

[254] The applicant's system uses Combiners to make the best possible use of the signal generating strategy elements by reviewing each individual signal's historic and current values, detecting interrelationships and performance patterns and learning when each signal type is performing well in the current market environment. The signals showing the most strength or expected to show the most strength presently are then most influential in the producing the system's final predictive output for each respective instrument or data type being modeled. At 818, 822 and 828 the user can specify the Look Back Time Frame, or the method of determining the LBTF, for Level 1, Level 2 and Level 3, respectively, by selecting one of the options from the drop down menu. In the example at 822, "Last n Periods" has been selected, causing the period specification field "where *n*=" to be displayed, as at 824. The user can then enter for *n*, the number of periods, or individual historical overlay advisor values, that they want the system to use in the machine learning based portions of the modeling processes. If, for example, the user specifies 500 days at 824, the system will use the last 500 values produced by Level 2 combiners, including the current period. If the user specifies a number of periods that exceeds the historical data available in the database for a given instrument or factor, (e.g., a newly listed stock of a newly created index included in the portfolio), the system will automatically use the maximum number of periods available, and will expand this to include new data until the specified number of periods exists (the applicant may provide an error notification pop-up window that requires user acknowledgement). The user can also specify that the LBTF for combiners' values at the prior level be "Dynamically Learned" as in the example at 818. This specifies that the LBTF will be dynamically adjusted to reflect the current optimum number of periods for each individual instrument

or data type being modeled (i.e., each individual modeling subject has its own optimum which could potentially be different with each new predictive task).

[255] At 820, 826 and 830, the Learned Optimization for combiners specified for use at Level 1, Level 2, and Level 3, respectively, allow the user to specify their preferred optimization parameters based upon narrow tuning objectives such as Lowest Volatility, Lowest Drawdown and Highest Profit. These settings operate on default parameters within the machine learning mechanisms themselves, by selecting a set of parameters that the applicant has specified as a default for each tuning objective, e.g., highest profit. On the advanced user Preferences screens there is a Learning Criterion-General window, as illustrated at 202 on Figure 7, that allows users to fine-tune the learning optimization at each processing layer through more specific parameters, such as Maximize Correct % (direction), Maximize Correct % (magnitude), Minimize Consecutive Error (direction), and Minimize Consecutive Error (magnitude). The user can weight these preferences to create a balance that suits their needs for the portfolio currently being created or modified. The applicant may, in the future, re-specify the default settings for system tuning, and may add additional tuning options.

[256] The user can establish advanced parameters for determination of the optimal LBTF and specify other advanced features by activating button Preferences at 786, which will open a separate interface screen that is the same or similar to Figure 7, except that all preferences will be associated with combiners only, and only with respects to portfolio currently being created or modified. Any specifications by the user on the advanced preferences screen accessed through the Combiner Association / Management screen will override any specifications from a prior overlay advisor preferences screen for the operative portfolio. The applicant will specify a default LBTF for both instruments and factors, and can make available any possible parameter through the drop down menu, or other selection means.

[257] At 784 of Figure 18, the button Configure 1:1 provides the user with access to a separate interface screen, that is the same or similar to the one illustrated by Figure 7, where all strategy elements that have been linked on a one-to-one basis to form a strategy chain can be associated with particular combiners.

[258] A default condition for the system specifies three processing layers, Level 1 the indicator processing layer, Level 2 the advisor processing layer, and Level 3 the overlay advisor processing layer. The user can activate Add/Delete buttons (+/-) 816, to add or delete a level for the portfolio currently being created or modified. Adding or deleting a level will duplicate or remove, respectively, all of the other features associated with a processing layer. For example, a user may wish to create a portfolio that will itself be specified as an advisor or overlay advisor in another portfolio, and since the final prediction will be subjected to further processing, the user may wish to omit the Level 3 overlay advisor processing layer. The applicant will provide collections of pseudorandom number generators that can be used to break ties in decisions/selections by combiners, and add noise to prevent over commitment to exact values, and to model chaotic system activity. Being pseudo-random these processes are actually following a deterministic rule that might actually be correlated with a system's activity. The applicant will specify the defaults with respect to pseudorandom number generators, and will make the settings available to the user through the advanced user preferences screen accessed at 786, or through a separate interface screen.

[259] In a preferred embodiment, the entire collection of system settings that operate on a particular processing layer/level can be optimized using one or more genetic algorithms. These genetic algorithms are seeded by either the default settings, or the user specified settings, and will then create combinations of system settings and evaluate the impact of these new combinations based upon a user supplied evaluation criterion (e.g., highest profit, or lowest volatility). The system's settings, as specified through the user interface screens, are either directly written into and/or translated into the applicant's intermediate language called Strategy Description Language ("SDL"), and stored in the database in SDL format. It is this SDL formatted settings data that the genetic algorithms used in the Dynamic Network Optimization feature of the system operate upon. The applicant may provide the user with the ability to specify their own genetic algorithms and/or modify default genetic algorithms through one or more additional interface screens.

[260] At 790, 794, and 798, the user can enable the Dynamic Network Structure Optimization ("DNSO") feature of the system, for Level 1, Level 2, and Level 3,

respectively. At 792, 796, and 800, the user can access an advanced Preferences screen, where the DNSO parameters can be specified, including the particulars settings or other controls that will be operated upon by the genetic algorithms and the degrees of freedom granted in the genetic optimization as various points within the system. The user can proceed to the next screen by clicking Finish button at 832.

[261] Figure 19 illustrates the Portfolio Management interface screen, which allows the user to perform a number of portfolio management functions. At 850 the user can activate a sort function by selecting a sort type from the drop down menu. Sort types can include the degree of similarity between portfolios, as in the example, asset class, research or executing portfolio types or any other possible sort type. Available Portfolios window 856, lists the symbol, icon or other identifier for each portfolio specified by the user selected sort type. A default condition is to show all portfolios in the system in alphabetical order. The user can drag-and-drop any portfolio symbol, icon or other identifier into Active Portfolio window at 858, which specifies the selected portfolio as the operative on in any subsequent action specified through this interface screen. A default condition is that only one portfolio can be modified at a time. At 860, the user can select an action to perform on the specified portfolio from the drop down menu. In the example, the selected action "Convert to Overlay Advisor," requires additional specifications such as what portfolio or library the new overlay advisor will be added to, as shown in the example. Other selection mechanisms that present information required to complete an action will be automatically displayed, as shown in the example drop-down menus at 864, and 866, which allow the designation of either a portfolio or library destination, and the name of the portfolio or library, respectively. The user can specify additional actions for the same operative portfolio, such as renaming it, or delete actions by using the Add/Delete button (+/-) 862.

[262] At 852 and 854, the user can access the Configure One-to-One and Preferences interface screens, respectively, for the specified portfolio. On the Configure One-to-One screen, the user will be provided with a drop-down menu or other selection mechanism, where the user will be required to specify the portion of the operative portfolio that they wish to modify, e.g., strategy elements within Level 1, Level 2 or Level 3, etc. Similarly, on the Preferences screen the user will be provided with a drop-down menu or other

selection mechanism, where the user will be required to specify the portion of the operative portfolio for which they wish to modify the preferences, e.g., indicators, advisors, overlay advisors, combiners, etc. After the user has finished specifying the desired actions, they can activate Finish button 868.

[263] Figure 20 illustrates the Back Test / Forward Test Creation and Management interface screen. The user can conduct back tests designed to test the performance of the selected portfolio historically by processing historical data through the current portfolio configuration (including strategy elements, machine learning components and tuning parameters, etc.) as well and forward tests which will test the performance of the selected portfolio using new, previously unseen data processed at each specified frequency. Back tests involve a defined historical number of periods where the data is used for training, and out-of-sample data (not used in the training data set, but not necessarily newly occurring data) from a second defined period is used for evaluating the performance of the portfolio configuration. The user can select any individual component of the system for testing from complete portfolios, trees from any processing layer, or any other signal generating strategy elements, e.g., indicators, advisors, overlay advisors, and combiners. The ability to back test and forward test any system component provides the user with maximum flexibility in research and strategy development.

[264] At 880, the user can use the drop-down menu to specify the Library of strategy elements they wish to select from. In the example, the user has selected "All." A default condition is to show all portfolios specified as research portfolios by the user, e.g., "All Research Portfolios." The contents of the library the user specifies will be displayed in Available Strategy Elements window 886. At drop-down menu 882, the user can specify the test type, e.g. back or forward or a combination. The user can drag-and-drop the symbol, icon or other identifier of any available strategy element into a New Test Group window as shown at 888. If more than one strategy element is selected, the test will be run with identical parameters for each selected strategy element. At 906 and 908, the user can specify type of range for the test, i.e., date or period, and the range, e.g. start date of the test and end date for the test, can be entered into the fields at 910 and 912. By activating the Add/Delete buttons (+/-) at 904, the user can specify more than one date range.

[265] To select a default set of parameters for the test, the user can select from drop-down menu 892, a Test Set. Default test sets are organized by asset classes and strategy types, so that variables such as slippage, e.g., currency futures vs. spot FX, and which signals to simulate execution of, e.g., dedicated short bias, are predefined for ease of use. At Preferences button 884, the applicant will provide advanced user preferences settings through a separate interface screen. This preferences screen will provide access to the full range of variables that a user could adjust, to reflect their perception of the actual market environment and its impact on the execution of their strategy. For example, the user can configure Test Set templates which record preferences which include, but are not limited to: the period covered in the test the benchmarks used for comparative analysis, the type of signal to include in the evaluation (such as the particular confidence levels, particular gain/loss ratios, etc), simulated stop loss and take profit levels, actual system recommended stop loss and take profit levels (may also be adjusted by a user defined variable), commission rates (as a flat dollar price or some function of the number of shares or other units), the user defined permissible long/short exposure ratio, leverage factors, etc. The preferences screen will also provide the user with an option to save and name the preferences set, so that it can be referenced for future use.

[266] At 894, the user can select one or more benchmarks for the test group from a drop-down menu. The applicant will provide default benchmarks, which will range from random signal generation to buy and hold comparisons, to popular indices. Selection of a benchmark will produce reports that allow for quick and easy comparison, through graphical representations, performance figures and industry standard statistical metrics. At 896, the user can specify the Report Type they would like generated, e.g., full or summary, and the format, e.g., .pdf or .xls. At 890, the user is provided with a Save Group As field for naming the test group for archiving and re-testing purposes. Modeling Frequencies for the test can be selected by checking the boxes at 486, which correspond to the prediction frequencies that will be tested. If a user elects to designate a custom frequency, they will enter notation that will be specified by the applicant, e.g., Day(09:00-15:30)NY.GMT, which specifies a custom day consisting of price activity between 9:00 am and 3:30 pm in New York. A default condition is that all strategy elements included in the test group will be tested all designated frequencies. At 898, the

Test Scheduler drop-down menu allows the user to specify when the testing will be performed by the system. The user can specify that the test run immediately upon activating Finish button 918, or they can specify that tests run at an off peak time, such as after execution portfolio processing is finished (e.g., after the close of market), or they can specify the exact time they want the test to run as shown in the example. If the user selects Specify Run Time, a field for entering the time will be displayed as at 900.

[267] The decision path transparency (described in more detail later) provided for each type of strategy element over the entire test period, provides the user with a very robust research tool that can lead to discovery of unique strategy ideas.

[268] Figure 21 illustrates the Portfolio View Output Screen which is one example of an interface screen based interactive report that allows the user to organize and analyze the system's output, and information derived from the system's output. A default condition is that this screen is the primary interface screen, i.e., the home page displayed after the user logs in, from which all other system screens can be accessed. The user can specify the portfolio they wish to view by selecting from all research and execution portfolio types available under drop down menu labeled Portfolio Name at 932. At 934, the Current View drop down menu allows the user to quickly change what information is displayed. For example, the user can view all information for the entire portfolio, or they could view only the long or short side of the portfolio. Drop down menu Sort Mode, at 938, allows the user to sort the prediction output data for both the long and short side of the portfolio in a variety of ways, e.g., by Confidence-Ascending (confidence levels are discussed in more detail later) as in the example, or by any number of other output types and formats, such as Magnitude-Descending, etc. At Quotation Basis drop down menu 940, the user can specify whether they want to see currency and/or percentage based quotation for data such as instrument prices and predicted magnitude moves, etc. The example at 942 shows the Date/Time window, where the report date and time as well as the current local portfolio time is shown.

[269] With each new prediction task, the system provides complete decision path transparency, which is displayed as in the example at 948. At 944, the Show Rankings For drop down menu allows the user to specify which category of strategy elements they want to view, and at 946, the Show drop down menu allows the user to specify the

number of strategy elements they want to be displayed. A default condition is to show the top 10 user provided/created strategy elements (i.e., excluding unbiased base machine learning components) of all types, as shown in the example. Both the current and previous rank and strength is also shown for each strategy element displayed. Export History button 950 allows the user to export a complete decision path history. Activating the button displays a separate interface screen where the user can select the export file type, e.g., .xls, .csv, .txt, .pdf or html, as well as the date or period based range for the data to be exported. The applicant will specify a default file format and data layout.

[270] At 952 and 966, the example shows the Long – One Hour window and Short – One Hour window (includes the entire tabular display immediately below each respective label) in one possible configuration, where long and short recommendations for a one hour modeling frequency are shown, respectively. The first four columns display the instrument name (or other data type being modeled), symbol, current price (real-time), the change over last (i.e., the difference between the close of the respective modeling frequency period and the prior period), which represent forms of data originating from a data feed, that is either raw, e.g., the current price, or derivative of raw data, i.e., is not an output of the system processes, like the change over the last periods closing price.

[271] The column at 954 displays the Predicted Next Close, which is the price the instrument (or other value for non-instrument data types) that is expected to be reached before or at the close of the current period for the specified modeling frequency. The column at 956, Magnitude, is the predicted magnitude of the movement in the price/value for the instrument or other data type for the current period (i.e., between the prediction output time and the next prediction output), and can be displayed in currency units and/ or a percentage of the base instrument price or data type value. The column at 958, Confidence – Probability, displays the Confidence Level (scale of 1-10 with 1 being the lowest confidence, and 10 being the highest confidence) and it's corresponding Probability statistics (expressed as a percentage) of the system's prediction for each respective instrument or data type, for the current period. The column at 960 and 962, Stop Loss and Take Profit, display the recommended stop loss and take profit levels for each respective instrument, respectively (non-instrument data types that can not be traded directly or indirectly, e.g., unemployment data, return N/A). Stop loss levels for long

positions are a function of the system's predicted Low price for each respective instrument for the current period and prediction error each respective instrument's prediction error. Take profit levels are a function of the predicted magnitude, (which can be either the expected High or Close, depending on what the user specifies) for each respective instrument for the current period, and the prediction error. The stop loss and take profit levels for the short positions are derived from the same data and function, except in a precisely converse way, e.g., stop losses for short positions are based upon the High and prediction error. These dynamically optimized stop loss and take profit levels can be used to replace or augment the arbitrarily fixed stop loss and take profit levels that are widely used in the financial sector. Because each instrument is modeled individually by the system, such levels are tailored to the unique characteristics of each instrument's price activity in the current market environment, which is something that can not be accomplished with other methods, e.g., always placing a stop loss so that not more than 0.25% is lost on a single instrument. Stop loss and take profit levels are also used to produce other valuable decision support data, that the can be used to evaluate instruments in the portfolio and filter opportunities and select the most advantageous recommendations to follow based upon a user's own investment objectives and risk tolerance. For example, column Gain/Loss Ratio at 964, provides the user with a very simple way to quickly and easily filter out the opportunities with the lowest risk vs. reward, which is measured by dividing the expected gain if the take profit level is reached by the expected loss if the stop loss level is reached first. In the example, the gain/loss ratio column is sorted in descending order (default), and if the user was going to choose to execute only one trade from both the long and short positions, the user might select the first instrument in each table, perhaps further filtering this data by reviewing the confidence levels and only accepting trades with a confidence level of 9 or above, in which case the user would execute the second trade in the long list (QQQ) and the first on the short list (DIA).

[272] The foregoing data is either produced directly from processes unique to the system, e.g., the magnitude and confidence, or they are derived from such data, as with the predicted next close, recommended stop loss and take profit levels and gain/loss ratio. All data that is available for long positions is also available for short positions (i.e.,

instruments or other data types which values are expected to decrease in the current period, finishing lower than the closing value from the prior period). At 968, the Portfolio ID drop down menu shows the portfolio the that is charted immediately below it, with the operative portfolio being specified by default, At 970, Chart Type drop down menu allows the user to select from a variety of default chart types or chart types created through the separate interface screen provided by the applicant that can be accessed by activating Chart Preferences button 974. At 972, the Compare To drop down menu allows the user to add an additional item to the chart, e.g., another instrument or basket of instruments, or a benchmark. At 976, the user can enable or disable a smoothing function that will operate on plotted items. The applicant can make available any possible chart utilities.

[273] The particular format and content of Figure 21 are for illustrative purposes only, and the applicant can make available any number of different windows that display the same or derivative data any possible format, as well as provide mechanisms for the user to utilize any new display type or feature, e.g., through advanced user preferences at 974.

[274] Figure 22 illustrates one possible configuration of an Instrument View – Output Screen, which the user can access, for example, by clicking on any instrument in the long or short position tables on the Portfolio View screen. The example shown is for SPY (S&P Deposit Receipts). The features that differ from the Portfolio View screen illustrated by Figure 21, are minimal for the sake of brevity. At 990, the Frequency drop down menu shows that Multi-Period has been selected. This selection specifies what information is shown in the window at 992 (including the entire tabular display below the label), which in the example, the selection multi-period effects Frequency at 994, causing all available prediction frequencies for this instrument to be shown. This selection will also effect other displays, defaulting them to the matching specification, as shown at 996 in the example of Chart Type – I, which is set to Extended Prediction with Confidence, displayed immediately below it. All features that are available for an entire portfolio view are also available for a single instrument view, and some of the features shown in Figure 22 for instruments, could also be applied to entire portfolios, e.g., for basket trading, etc. Any number of charts can be viewed by the user, and at 1002, there is a selection mechanism for Chart Type – II, which, for purposes of illustration, displays the

Prediction Error chart immediately below. At 998, a user can select a Summary Type, which summarizes all of the vital statistics based upon the type of summary chosen. At 1000, the example shows the Summary – Optimum, which provides information that is useful in making a decision based upon the larger information set in the window where all frequency data is displayed. The applicant will provide summary type templates which can be modified in whole or in part by the user through the advanced user preferences screen, where the user can chose to display any prediction output data or metric available within the system. All windows/displays in both Figures 21 and 22 can be moved and resized to create a custom view that is recalled for the user each time they log in.

[275] Although the invention has been discussed with respect to specific embodiments, these embodiments are merely descriptive, and not restrictive, of different aspects of the invention. For example, although specific user interface controls such as buttons, sliders, menus, text boxes, etc., are disclosed, any type of user interface controls can be adapted for use with the present invention. For example, voice detection, gestures or positional information, three-dimensional or "virtual reality" or other types of controls or input devices are possible. Also, any suitable type of visual output or representation can be used in place of, or in addition to, the example displays of the disclosed embodiments. Other output formats can be used including audio, three-dimensional, etc.

[276] Although specific factors, indicators, instruments, algorithms, parameters, values or other computational items are disclosed, other embodiments may use any other suitable types of computational items including data structures, data representations, etc. Although functionality has been described with respect to a preferred system or collection of features, other embodiments can include only portions of the functionality disclosed herein. Various of the functions can be used as auxiliary functions, or "plug-ins," to other analysis systems. The functionality discussed herein can be used as a standalone system or in conjunction with other programs or systems.

[277] The present invention, in some embodiments, also relates to apparatus for performing the operations herein. This apparatus may be specially constructed for the required purposes, or it may comprise a general purpose computer selectively activated or

reconfigured by a computer program stored in the computer. Such a computer program may be stored in a computer readable storage medium, such as, but is not limited to, any type of disk including floppy disks, optical disks, CD-ROM's, and magnetic-optical disks, read-only memories (ROM's), random access memories (RAMs), EPROMs, EEPROMs magnetic or optical cards, or any type of media suitable for storing electronic instructions, and each coupled to a computer system bus.

[278] The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various general purpose systems may be used with programs in accordance with the teachings herein, or it may prove convenient to construct more specialized apparatus to perform the required method steps. The required structure for a variety of these systems will appear from the description below. In addition, the present invention is not described with reference to any particular programming language, and various embodiments may thus be implemented using a variety of programming languages.

[279] From the foregoing, it will be appreciated that specific embodiments of the invention have been described herein for purposes of illustration, but that various modifications may be made without deviating from the spirit and scope of the invention. In some instances, reference has been made to characteristics likely to be present in various or some embodiments, but these characteristics are also not necessarily limiting on the spirit and scope of the invention. In the illustrations and description, structures have been provided which may be formed or assembled in other ways within the spirit and scope of the invention.

[280] In particular, the separate modules of the various block diagrams represent functional modules of methods or apparatuses and are not necessarily indicative of physical or logical separations or of an order of operation inherent in the spirit and scope of the present invention. Similarly, method have been illustrated and described as linear processes, but such methods may have operations reordered or implemented in parallel within the spirit and scope of the invention.

[281] The foregoing description of illustrated embodiments of the present invention, including what is described in the Abstract, is not intended to be exhaustive or to limit the invention to the precise forms disclosed herein. While specific embodiments

of, and examples for, the invention are described herein for illustrative purposes only, various equivalent modifications are possible within the spirit and scope of the present invention, as those skilled in the relevant art will recognize and appreciate. As indicated, these modifications may be made to the present invention in light of the foregoing description of illustrated embodiments of the present invention and are to be included within the spirit and scope of the present invention.

[282] In the description herein, numerous specific details are provided, such as examples of components and/or methods, to provide a thorough understanding of embodiments of the present invention. One skilled in the relevant art will recognize, however, that an embodiment of the invention can be practiced without one or more of the specific details, or with other apparatus, systems, assemblies, methods, components, materials, parts, and/or the like. In other instances, well-known structures, materials, or operations are not specifically shown or described in detail to avoid obscuring aspects of embodiments of the present invention.

[283] Reference throughout this specification to "one embodiment", "an embodiment", or "a specific embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the present invention and not necessarily in all embodiments. Thus, respective appearances of the phrases "in one embodiment", "in an embodiment", or "in a specific embodiment" in various places throughout this specification are not necessarily referring to the same embodiment. Furthermore, the particular features, structures, or characteristics of any specific embodiment of the present invention may be combined in any suitable manner with one or more other embodiments. It is to be understood that other variations and modifications of the embodiments of the present invention described and illustrated herein are possible in light of the teachings herein and are to be considered as part of the spirit and scope of the present invention.

[284] It will also be appreciated that one or more of the elements depicted in the drawings/figures can also be implemented in a more separated or integrated manner, or even removed or rendered as inoperable in certain cases, as is useful in accordance with a particular application.

[285] As used in the description herein and throughout the claims that follow, "a", "an", and "the" includes plural references unless the context clearly dictates otherwise. Also, as used in the description herein and throughout the claims that follow, the meaning of "in" includes "in" and "on" unless the context clearly dictates otherwise.

[286] The foregoing description of illustrated embodiments of the present invention, including what is described in the Abstract, is not intended to be exhaustive or to limit the invention to the precise forms disclosed herein. While specific embodiments of, and examples for, the invention are described herein for illustrative purposes only, various equivalent modifications are possible within the spirit and scope of the present invention, as those skilled in the relevant art will recognize and appreciate. As indicated, these modifications may be made to the present invention in light of the foregoing description of illustrated embodiments of the present invention and are to be included within the spirit and scope of the present invention.

[287] Thus, while the present invention has been described herein with reference to particular embodiments thereof, a latitude of modification, various changes and substitutions are intended in the foregoing disclosures, and it will be appreciated that in some instances some features of embodiments of the invention will be employed without a corresponding use of other features without departing from the scope and spirit of the invention as set forth. Therefore, many modifications may be made to adapt a particular situation or material to the essential scope and spirit of the present invention. It is intended that the invention not be limited to the particular terms used in following claims and/or to the particular embodiment disclosed as the best mode contemplated for carrying out this invention, but that the invention will include any and all embodiments and equivalents falling within the scope of the appended claims.

WHAT IS CLAIMED IS:

1. A method for predicting the behavior of a system, the method using a processor coupled to a user input device and a display, the method comprising:

displaying a plurality of possible instruments;

accepting a signal from a user input device to select one or more instruments as predictor instruments, wherein the predictor instruments include data points;

accepting a signal from a user input device to designate one or more possible indicators as active indicators;

using one or more of the active indicators to process one or more of the data points; and

indicating a result of processing a data point.

2. The method of claim 1, wherein an instrument includes a financial security.

3. The method of claim 2, wherein the instrument includes a stock.

4. The method of claim 2, wherein the instrument includes a bond.

5. The method of claim 2, wherein the instrument includes an option.

6. The method of claim 2, wherein the instrument includes a commodity future.

7. The method of claim 1, further comprising

accepting a signal from a user input device to designate one or more factors.

8. The method of claim 7, wherein an instrument includes data points having data types, wherein a factor includes a data type designation, the method further comprising:

using the factors to determine data points having data types in accordance with a factor's data type; and

using the determined data points as inputs to the active indicators.

9 An apparatus for predicting the behavior of a system, the apparatus comprising:

a processor coupled to a user input device and a display;

a machine-readable medium including instructions executable by the processor, the machine-readable medium including

one or more instructions for displaying a plurality of possible instruments;

one or more instructions for accepting a signal from a user input device to select one or more instruments as predictor instruments, wherein the predictor instruments include factors;

one or more instructions for accepting a signal from a user input device to designate one or more possible indicators as active indicators;

one or more instructions for processing the output of one or more active indicators; and

one or more instructions for displaying a result of the processing.

10. A method for analyzing a financial instrument, the method comprising:

identifying a plurality of instruments having data points;

allowing user selection of a plurality of processes to process the data points and provide outputs;

allowing user selection of a combining of the outputs; and

adjusting performance of an analysis based on a comparison of an output with a past event.

11. The method of claim 10, wherein the past event includes a performance of a financial instrument.

12. A method for applying analysis techniques to a collection of data, the method executing on a computer system, the computer system including a processor coupled to a display screen, the method comprising:

displaying a plurality of data sources on the display screen, wherein the data sources include data points;

displaying a plurality of data types on the display screen;

accepting a signal from a user input device to select a data type;

identifying data points of the data sources having the selected data type;

displaying a plurality of advisor processes on the display screen, wherein the advisor processes accept inputs and provide outputs;

accepting a signal from a user input device to select one or more advisor processes; and

applying the data points having the selected data type to the selected advisor processes to provide an input.

13. The method of claim 12, further comprising:

accepting a signal from a user input device to allow selection of a modeling frequency; and

using the modeling frequency to obtain a result based on a plurality of data points sampled at the modeling frequency.

14. The method of claim 12, further comprising:

displaying a plurality of indicators on the screen, where an indicator performs pre-processing on one or more data points; and

accepting a signal from a user input device to allow a user to select an indicator for use in pre-processing.

15. A method for predicting a stock price comprising:

pre-processing stock data from a large set of mathematical indicators to produce indicator output signals;

entering the indicator output signals into a database;

processing with advisors the indicator output signals to produce advisor output signals;

enter the advisor output signals into a database; and

inputting the advisor output signals into a neural network to produce a prediction of a stock price;

entering the neural network prediction into the database; and

iteratively updating the neural network weights for all stocks and system components upon receipt of new data.

16. The method of claim 15, wherein the indicators can be any form of signal generating algorithm or output device.

17. The method of claim 15, wherein the minimum default indicator is the calculated change of a data value over the prior data value in the series.

18. The method of claim 15, wherein machine learning based advisors process the indicator output signals.

19. A method for predicting securities prices and other data types comprising:

Pre-processing securities price data and other data types using mathematical indicators to produce indicator output signals;

Entering the indicator output signals into a database;

Processing with advisors the indicator output signals to produce advisor output signals;

Entering the advisor output signals into a database;

Inputting the advisor output signals into a neural network to produce a prediction of a securities price or other data types;

Entering the neural network prediction into the database;

Processing output signal data with overlay advisors to produce overlay advisor output signals;

Entering the overlay advisor output signals into a database;

Inputting the overlay advisor output signals and lower-level neural network output signals into a second high-level neural network to produce a final prediction of securities price or other data types; and

Iteratively updating the neural network weights for all securities and other data types and system components upon receipt of new data.

20. The method of claim 19, wherein a machine learning based system for predicting securities prices and other data types is modularized to provide insertion points for use by a non-technical user to fully configure the system for use;

21. The method of claim 19, comprising:
an insertion point for securities instruments and factor data;
an insertion point for mathematical indicators; and
an insertion point for advisors; and an insertion point for overlay advisors;

22. The method of claim 20, wherein the insertion points can accept any number of additions from the user;

23. The method of claim 19, wherein the indicators can be any form of signal generating algorithm or output device.