Using Trading Dynamics to Boost Quantitative Strategy Performance

Executive Summary

The objective of performance boosting, a process developed over the past two decades by Hood River Research, is increasing the returns that can be earned from the buy and sell signals issued by an existing stock selection strategy[1]. This process is premised on the notion that if the strategy is an objective algorithm[2] that can be backtested, it may be possible to predict the outcomes[3] of its buy and sell signals to an economically meaningful degree. Strategy returns are boosted by taking larger than normal positions on signals predicted to earn above average returns, smaller than normal positions on signals predicted to earn below average returns and no position when the predicted return is negative. The Hood River process incorporates specialized data transformation and modeling techniques specific to predicting signal returns.

Prediction of Signal Outcomes

The predictions, either in the form of a forecast of a signal's return or a probability estimate that the signal will result in a profit, are produced by a customized second-stage model(s) developed by Hood River Research for a client's stock selection strategy(s).

The information serving as input to the second-stage model is an extensive set of predictor variables that quantify a stock's trading dynamics at the time a buy or short-sale is generated by the stock selection algorithm. These

---

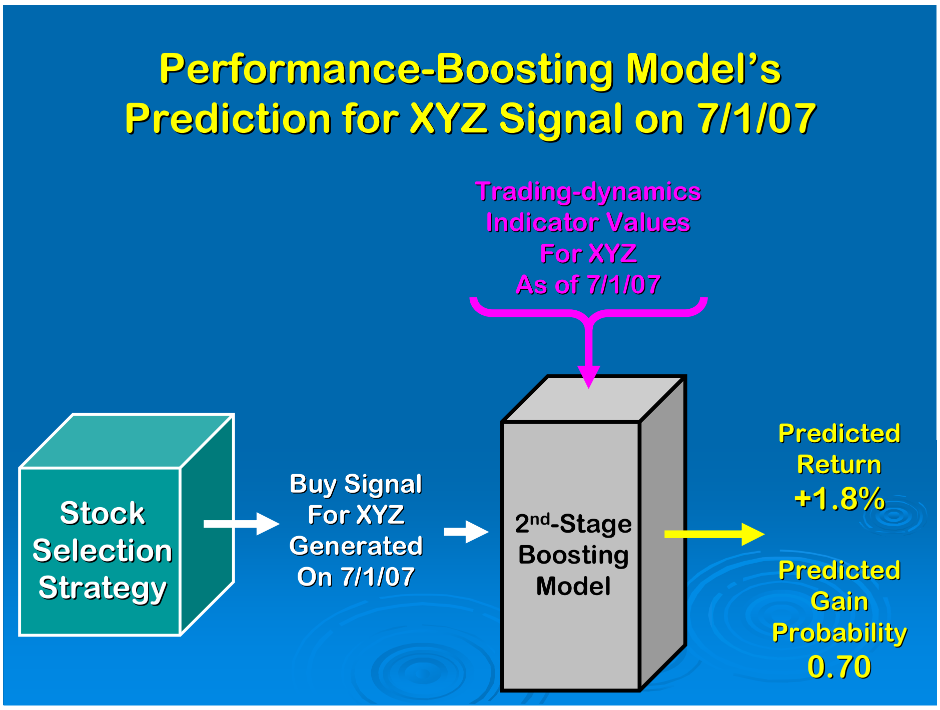
[1] The term strategy and buy / sell signaling algorithm are used interchangeably in this paper.

[2] An objective stock selection strategy is one that can be reduced to a computerized algorithm that can backtested, thus producing a sample of past signals and their return outcomes.

[3] Outcomes can be defined either as the positive or negative return earned by the signal at exit, or as a class (return > 0 = class 1, else outcome is class 2)

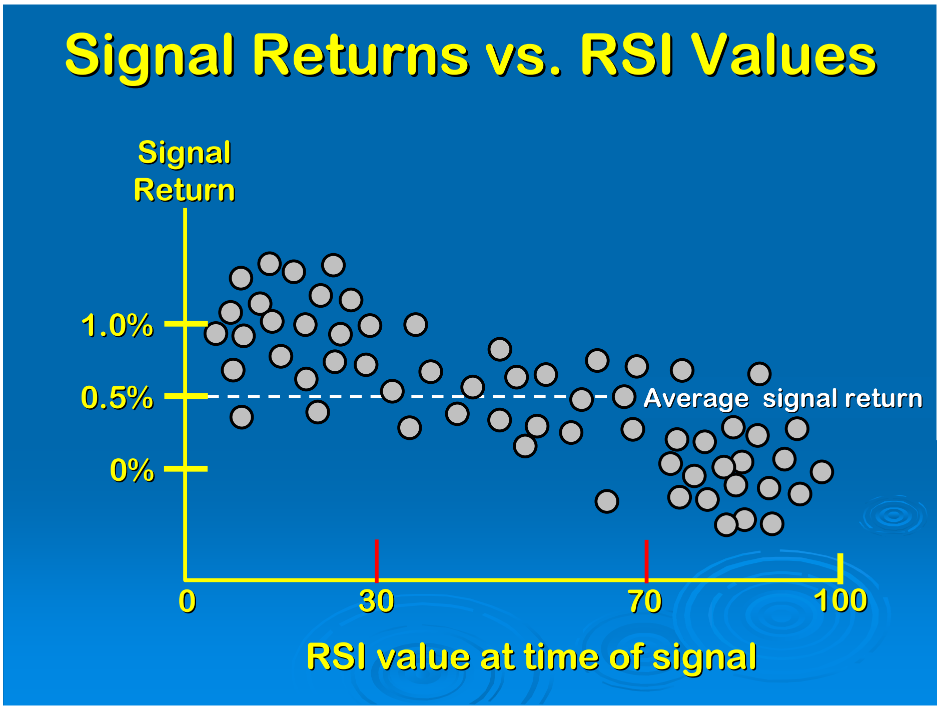
trading dynamics indicators are derived by applying various mathematical transformations to the raw price and volume data for the stock in question and to the price index representing the universe to which the stock belongs. It should be mentioned that indicators other than those based on trading dynamics, where appropriate, can also be employed as model inputs.

To clarify this concept of signal prediction, imagine that a historical backtest of a stock selection strategy shows its buy signals produce an average one month return of 0.5 percent above the universe and that 53% of the signals are profitable. Suppose a signal is given to buy XYZ stock on July 1, 2007. Given the values of XYZ's trading dynamics indicators as of July 1, the second-stage model forecasts this particular buy signal will earn 1.8% and has probability of profit is 0.70. This prediction is based on the fact that in the past when strategy signals were characterized by values similar to XYZ's they earned an average return of +1.8% and had a 0.70 probability of profit.

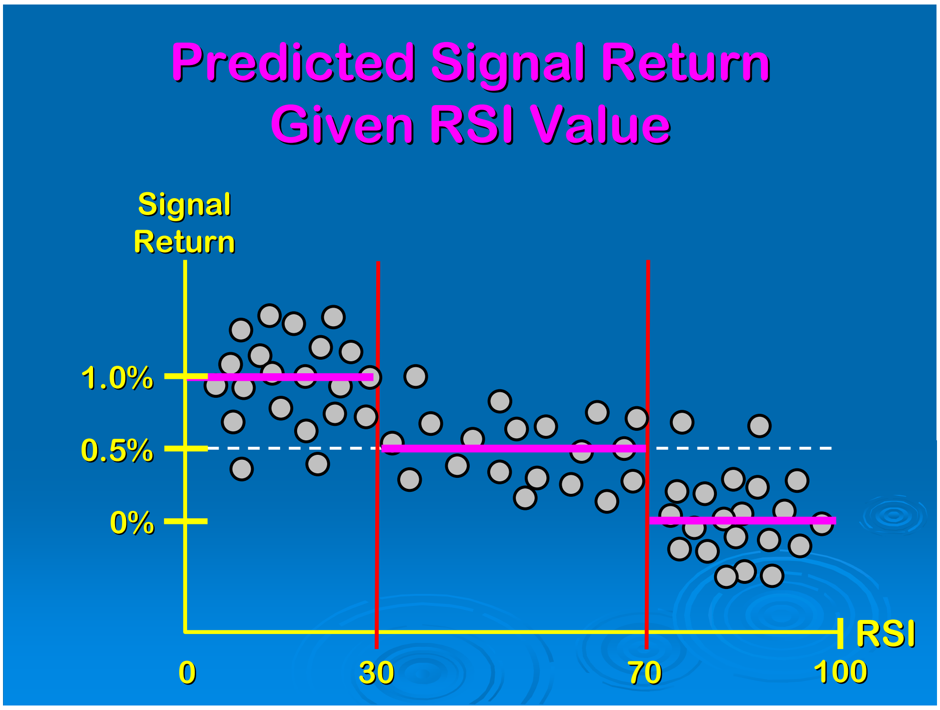


Using Trading Dynamics Indictors To Predict Signal Outcomes

The following is a simplified explanation of how the values of dynamics indicators registered at the time of a signal are translated into a prediction the of signal's return. Assume that only a single trading-dynamics indicator, like the Relative Strength Index (RSI), is being used as a predictor variable. RSI, a widely used technical indicator, quantifies the price trend of a security. This indicator has a possible range of 0 and 100, where values between 0 and 50 indicate a negative price trend while values in excess of 50 indicate a positive trend. Suppose that after analyzing the returns produced by a historical sample of the strategy's signals, it is discovered that when RSI has a value less than 30, signals have had above average returns but when RSI is greater than 70 signals produce below average returns. This situation is depicted in the illustration below.



This diagram tells us that realized signal returns are, to a meaningful degree, conditional upon (predicted by) the value of RSI at the time the signal was given. A boosting model is essentially the complete set of conditional return predictions at all possible values of the trading dynamics and other relevant indicator(s). Below we show a crude[4] but nevertheless useful boosting model based on this data. Note that the altitude of the purple horizontal line within each of the three regions along the RSI axis (0 to 30, >30 to <70 and 70 to 100) represents the boosting model's prediction of signal return given (conditional upon) the RSI value. For example, when RSI is in the range of 0-30 the signal's predicted return is +1.0% vs. the unconditional (without reference to RSI value) signal average of 0.5%



With this knowledge the user of this buy/sell algorithm would be able to take larger than normal positions when its signals were accompanied by RSI values less than 30, smaller than normal positions or no position on signals when RSI was greater than 70 and normal positions at all other times. If the historical relationship between RSI and signal returns were to persist into the future, the return on capital allocated to this strategy would be boosted. It can be said that performance boosting is focused on discovering the set of indicators that are best able to predict the outcomes of a strategy's buy and sell signals.

[4] The term crude is used to describe the model because the prediction within each given range of RSI is a constant value within that range. In actual practice, the model would be represented by one or more smooth nonlinear surfaces.

In actual practice trading dynamics indicators would not be nearly as strongly predictive as in the illustration and a single indicator could not be relied upon. Rather predictions would be based on numerous indicators and an ensemble of complementary boosting models whose forecasts would be combined into a prediction. (See below)

The Benefit of Performance Boosting: More Knowledge

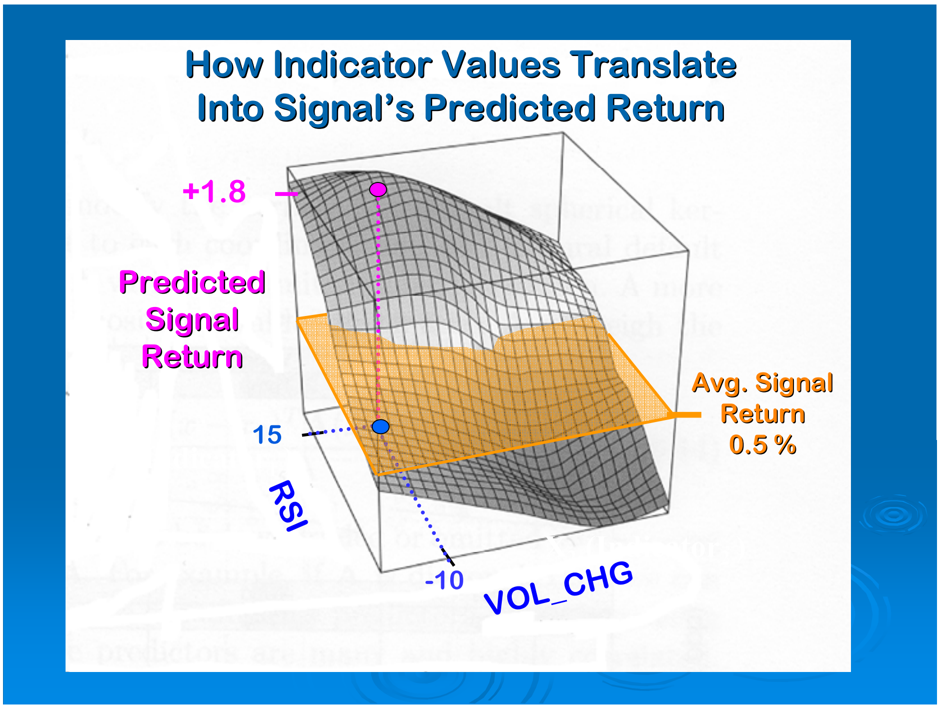
To understand the informational advantage conferred by the performance boosting model, consider the state of knowledge of the investor not using one. Suppose this investor employs a low-PE investing strategy[5]. Each month all stocks in the investor's universe are broken into deciles based on their PE ratio. A typical approach would be to buy the stocks in the lowest PE decile (decile 1). Let's also suppose that a historical backtest of the strategy has shown that stocks in the lowest PE decile produce an excess return versus the universe of 0.50% over the one month period following purchase. From this investor's state of knowledge, all that can be said each time the low-PE strategy signals the purchase of a security is that the expected one month excess return is 0.50%. All other information about the security is ignored – information that might help improve the prediction of the signal's return.

Now consider an investor who follows the same low-PE strategy but who is using a performance boosting model customized for the strategy. For ease of illustration, we will assume that the boosting model utilizes only two trading dynamics indicators: (1) the RSI indicator and (2) VOL_CHG, the rate of change in the stock's trading volume.

When the low-PE strategy issues a buy signal for a particular stock, the values of these two indicators for the stock serve as inputs to the boosting model. Let's suppose that as of the date of the buy signal RSI has a value of 15 and VOL_CHG has a value -10. Note in the diagram below these two values specify the coordinates of a point (blue dot) in a two-dimensional space, where one axis of the space represents RSI while the other represents VOL_CHG. Now note that this location is associated with a specific point

[5] Typically client strategies would be more complex and based on a variety of screening criteria or a ranking system based on a multitude of static or dynamic factors.

(purple dot) on the nonlinear surface floating above the 2-d indicator space. This location has an altitude of +1.8% with respect to the vertical or third dimension which represents the predicted return for the signal. Thus, +1.8% is the predicted return for this signal.



It can be said the shape of this nonlinear surface depicts the relationship between the values of the trading dynamics indicators and the predicted return of the signal discovered in a sample of prior signals. Note that signals characterized by similar indicator values, such as an RSI of 14 or 16 and a VOL_CHG of -11 or -9 would have a predicted outcome that is similar to a value of +1.8%.

This example was limited to two predictors so that the third dimension could be used to represent the signal's return. However, in practice the performance boosting model may contain numerous predictors (dimensions). Hence, the model's surface would be a multi-dimensional extension of the idea of a surface, called a "hyper-surface". Although this cannot be illustrated, one can think of a boosting model as a nonlinear hyper-surface floating in a space of n dimensions, where n-1 of the dimensions are allocated to the trading dynamics indicators and the remaining dimension is the return earned by the signal.

In the diagram the knowledge of the investor operating without the boosting model is represented by the level flat orange surface. It has a uniform altitude of +0.5%, irrespective of the value of either trading dynamics indicator or any other variable for that matter. This is simply another way of saying that the investor without the boosting model always expects (predicts) the same outcome for all signals, its historical average return of +0.5%. In contrast, the knowledge of the investor using a boosting model is represented by the non-linear surface. This investor brings a greater variety of information to bear in forming a return expectation for a given signal and in deciding what size position to take on it.

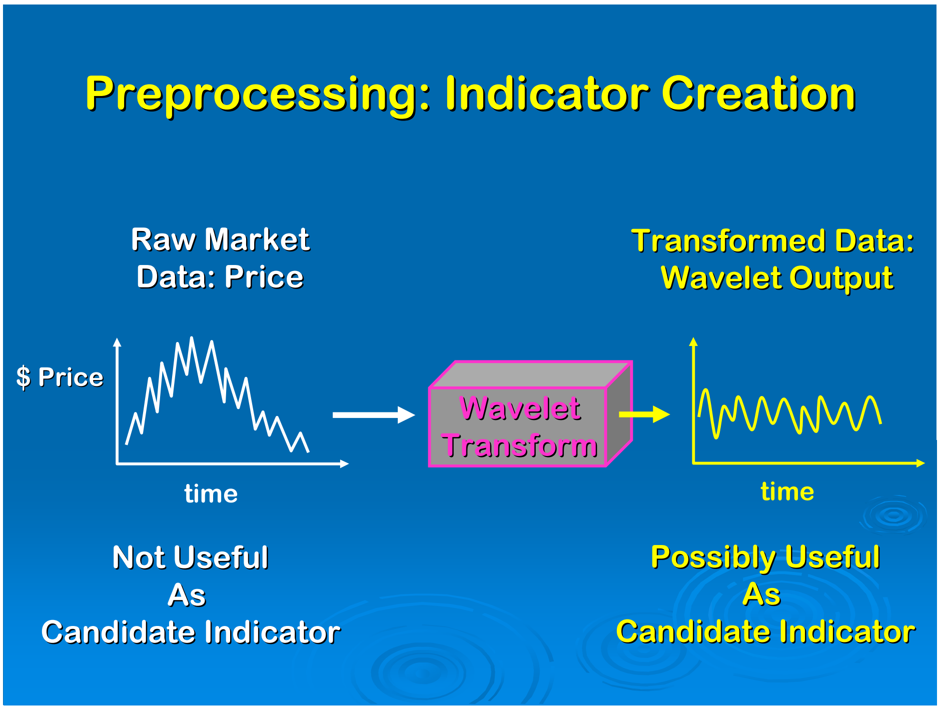
Candidate Predictors and Preprocessing

The most important determinant of success in performance boosting is the information content in the set of candidate trading dynamics indicators proposed for consideration by the automated modeling procedure. The burden of this crucial task falls on the shoulders of a human expert.

What is essential is that at least some of the candidate indicators contain information relevant to predicting signal returns. If this requirement is met even a relatively simple modeling procedure such as step-wise multiple linear regression, can produce a useful boosting model. However, if none of the proposed indicators carry information relevant to predicting signal returns not even the most powerful modeling methods will yield a good boosting model.

The core task in creating an information rich list of candidate indicators is *data preprocessing.* Data preprocessing refers to the transformation of raw financial market data into the indicator variables. Raw market data such as prices, volume data and a host of other publicly available statistics are rarely useful in their native form for modeling. This task does not yet lend itself to automation.

Data preprocessing involves the application of various mathematical operators (i.e., transformations) to the raw data. The transformations range from simple operators such as averages and ratios to advanced forms of digital filtering, compression and smoothing. An example of an advanced transformation that can be useful in predicting signal outcomes is the *wavelet* transform. Wavelets are useful for isolating transitory non-period fluctuations typical of financial market data that more conventional methods such as Fourier analysis and spectral analysis may miss.



Preprocessing serves several important functions. First, it assures that an indicator will be reasonably stationary. That is to say it maintains relatively stable statistical characteristics over time, such as a stable mean and variance. Without this property the modeling procedure is unable to discern if there is a useful relationship between the indicator and signal returns.

Second, preprocessing conserves a valuable and scarce modeling resource, the number of dimensions used in the indicator space. The number of indicators (dimensions) that can be used to good effect in a performance boosting model is quite limited. Although in theory, modeling procedures can construct performance boosting models using a nearly unlimited number of dimensions, as a practical matter, the amount of historical data (i.e., the quantity of historical signals provided by the client) severely limits this number. As pointed out above, each indicator consumes one dimension of the model's space. When the number of indicators becomes large relative to the number of historical signals, the data becomes too sparse within the model space. In other words, it becomes spread too thinly. This is a problem because there is a minimum level of data density required to discover the correct shape of the model's prediction surface. This problem, known as the *curse of dimensionality* tells us that as of the number of indicator variables is increased the required number of historical observations needed to adequately populate the model space goes up at an exponential rate. Hence, if 100 observations provide adequate data density for a model with two indicators then 1000 observations are needed to provide the same level of data density for three indicators, 10,000 for four and 100,000 for five, etc.

Preprocessing can conserve dimensions when, as a consequence of the analyst's expertise and insight two or more indicators can be condensed into a single more potent indicator[6]. Suppose, for purposes of this discussion, signal outcomes can be predicted by the degree of coherence between the price momentum of the stock and the price momentum of the universe to which the stock belongs. Also suppose that both the stock's price momentum and the universe's price momentum were supplied as candidate indicators. Eventually the modeling procedure would discover that the two momentum indicators used conjointly were useful. However, this would consume two dimensions in the model. But, if the analyst had the insight to propose an indicator which quantifies the coherence between the two momentum indicators the modeling procedure would have the opportunity to select it thus conserving a dimension.

A third function of preprocessing is noise reduction. To understand this concept it is useful to think of market data as a combination of information and noise. The more that can be done to reduce the noise component the more easily the modeling procedure can glean the informative component. This is called filtering. Much of preprocessing is about helping the modeling procedure be more effective by reducing uninformative random variation. Many technical indicators such as moving averages, RSI, ADX (average non-directional trend strength), ATR (average true range), etc. can be viewed as simple but often effective filters that reduce noise of raw market times series data.

[6] There are statistical methods for condensing a set indicators with redundant information into fewer dimensions that can also be used.

Indicators Used in Performance Boosting

Typically the number of preprocessed trading dynamics indicators offered as candidates to the modeling algorithm number is in the range of 200 to 500. This section gives an overview of the approach taken by Hood River Research.

The candidate indicators used in Hood River's boosting process fall into three general categories: 1) measures that pertain to an individual stock, 2) measures that pertain to the universe to which the stock belongs, and 3) measures that quantify the degree of coherence or divergence between an individual stock and its universe with respect to a specific indicator.

Indicators based on an individual stock are derived from the stock's price and volume data. While conventional studies are typically limited to the first derivative (e.g., price and volume momentum) and low order moments of the price and volume distribution (averages and variances), the Hood River process uses higher order derivatives such as acceleration, change in acceleration, and higher order moments of the price and volume distribution such as skewness and kurtosis. These often neglected indicators can offer additional independent and potentially useful information. Moreover, other transformations such as trend strength, irrespective of direction, and the degree of order & disorder in the price and volume time series are used.

The second type of indicator is the same set of transformations applied to individual stocks but applied to the index or universe average to which the stocks belong. These predictors can provide valuable contextual information. For example, it may be discovered that signals earn the best returns when the stock universe is operating within a particular regime such as a high volatility down trend.

The third type of predictor measures the degree to which an individual stock's behavior is conforming to or diverging from its universe. Often there is useful information in these indicators that is independent[7] of that found in indicators based the individual stock or the universe.

One of the most attractive aspects of applying performance boosting to stock strategies is the large number of observations that are obtained by

[7] Independent means that the indicator's information is orthogonal to the other types of indicators.

aggregating signals across an entire universe of securities. As pointed out above, a large the number of observations allows more indicators to be used in a boosting model without the data becoming too sparse[8]. However, aggregating signals is only feasible if the trading dynamics indicators are robust to differences in stock trading dynamics. Because stocks can vary greatly from one another indicators must be defined so that they are comparable. For example, an indicator that has a historical range of 40 to 60 for one security but 20 to 80 for another security would not permit the signals from both stocks to be aggregated. Hood River Research has developed cross-sectional normalization methods that allow signals to be aggregated across a universe of differently behaving securities.

Booster Model Development

The development of a boosting model is a process of discovery in which two types of information are discovered by the modeling procedure(s). The first is discovering which, if any, of the large list of candidate trading-dynamics indicators proposed for consideration, are helpful in predicting signal returns. The term "candidate predictor" is appropriate because the decision as to which indicators will be selected is left to the modeling procedure.

The second type of information is the shape of the surface that best describes the relationship between the selected trading dynamics indicators and signal returns. One widely used modeling technique called multiple linear regression, assumes that the shape of the surface is linear (flat with no hills and valleys). Only the slope of the surface with respect to each axis (i.e., the weight of each indicator) is left open to discovery. However, modern data modeling techniques have allowed the constraining assumption of a flat surface to be eliminated. This allows the modeling procedure to discover the most appropriate shape for the model's hyper-surface. This added flexibility allows the data to speak their message without forcing the answer to conform to a preconceived and often overly simplistic assumption. The distinction between traditional linear modeling and modern flexible non-linear modeling is discussed below.

[8] The Curse of Dimensionality problem.

Unlike preprocessing, which relies on human expertise, model development has been automated to a high degree. This is fortunate because numerous studies[9] have shown that human intelligence does not perform this task well. The studies compared the accuracy of predictions made by experts using subjective judgment with predictions produced by multiple linear regression models using the same set of predictor variables. One typical study showed that the accuracy of the regression model was nearly four times that of the experts[10]. Both the experts and models used the same set of predictor variables.

Advanced Modeling vs. Traditional Linear Regression

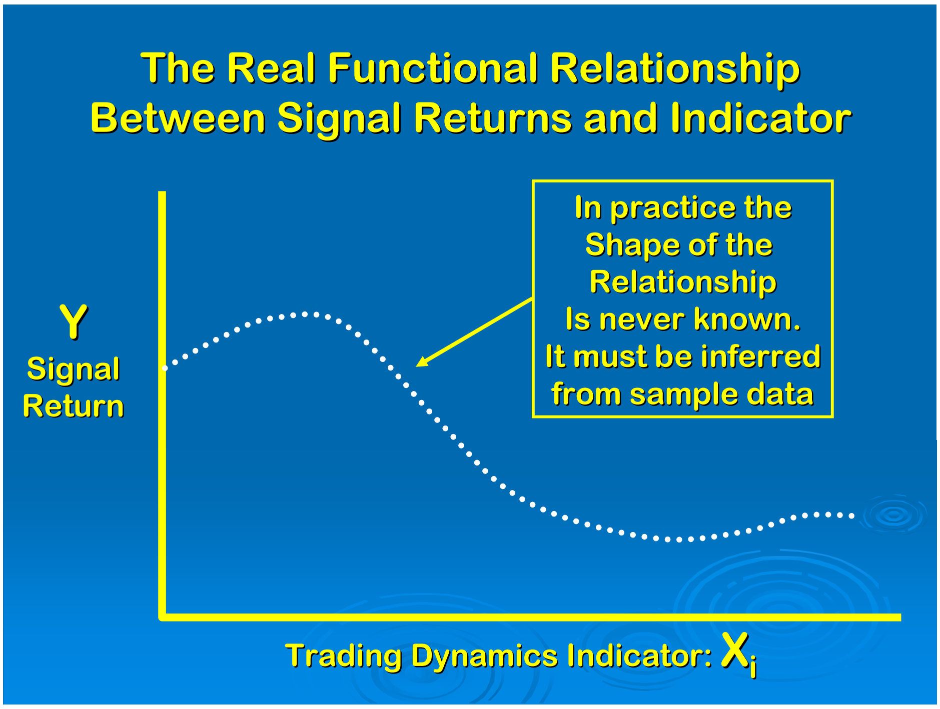
As powerful as multiple linear regression is relative to the intuitive judgment of human experts, greater predictive power can be attained with more advanced modeling methods that are not constrained by the simplifying assumption of linearity. This creates the opportunity for more accurate predictions of signal outcomes. Advanced methods such as kernel regression can detect complex non-linear relationships.

To give an intuitive sense of how non-linear modeling differs from traditional linear regression we present a series of diagrams below. For simplicity, the illustrations depict a single indicator (X_i) on the horizontal axis and the return earned by the signal on the vertical axis. In the first illustration we begin by showing the true, but unknown, functional relationship between signal returns and X_i. It is unknown in the sense that the true shape of the function sought in any predictive modeling problem is by definition unknown and remains to be inferred from an observed sample of data. Note that the relationship is not linear.

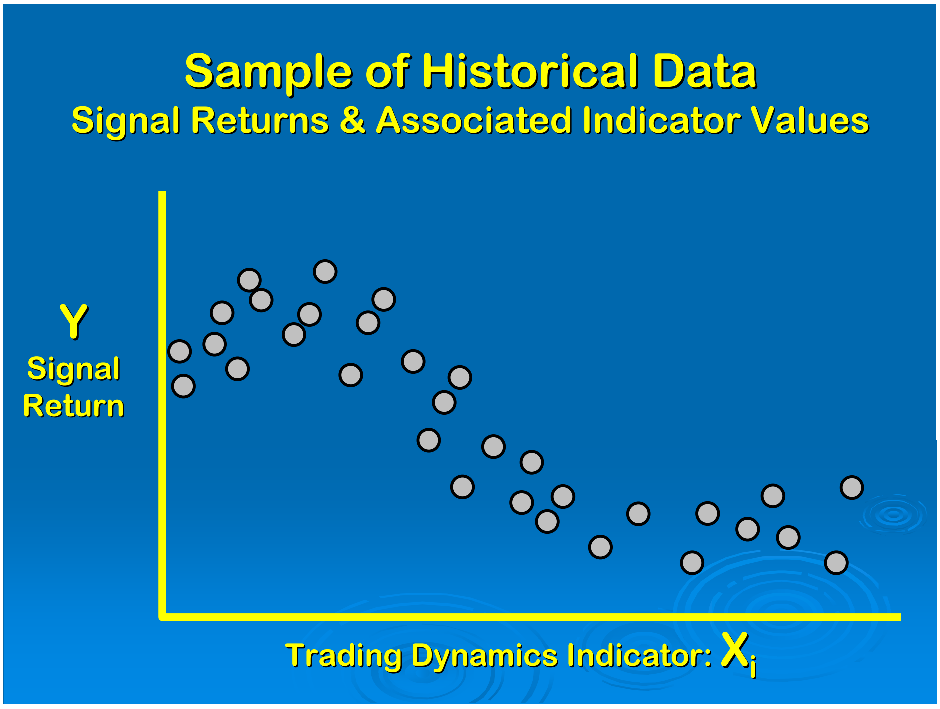
---

[9] For a listing of studies see, Aronson, David, Evidence-Based Technical Analysis, Wiley & Sons 2006. See endnotes 20-26 Chapter 2 and Chapter 9 endnotes 33 – 43.

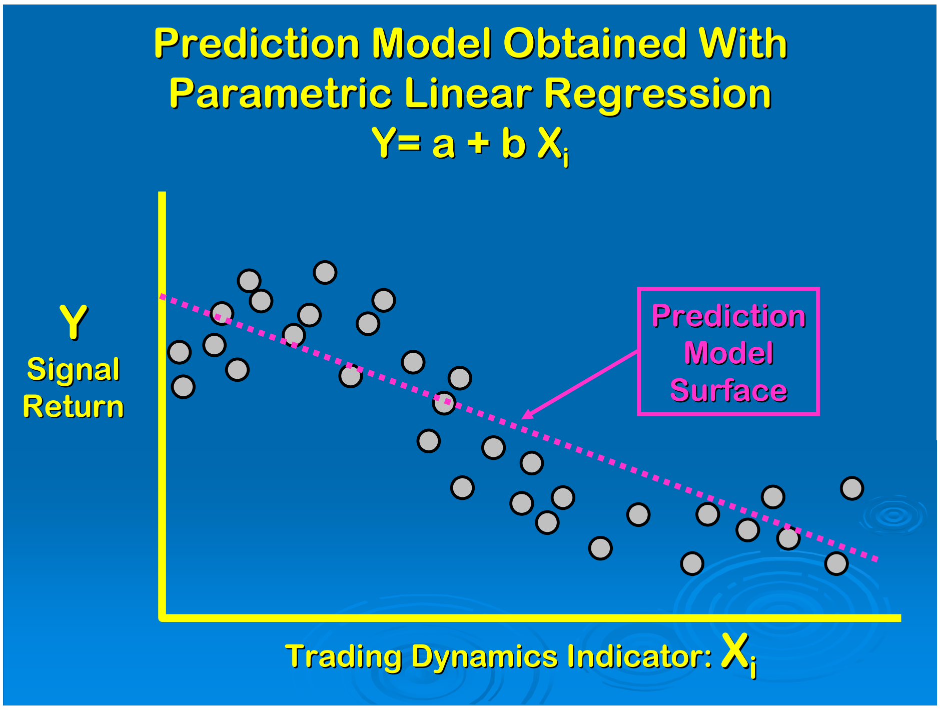
[10] Based on a summary of studies presented by J.E. Russo and P.J.H. Shoemaker in "Decision Traps" Doubleday, 1989. Among the studies cited were: "Clinical Versus Statistical Prediction" by Meehl, Dawes and Faust, *Science*, 1989 "General Conditions for the Success of Bootstrapping Models" by , Camerer, Colin, in Organizational Behavior and Human Performance, 1981.



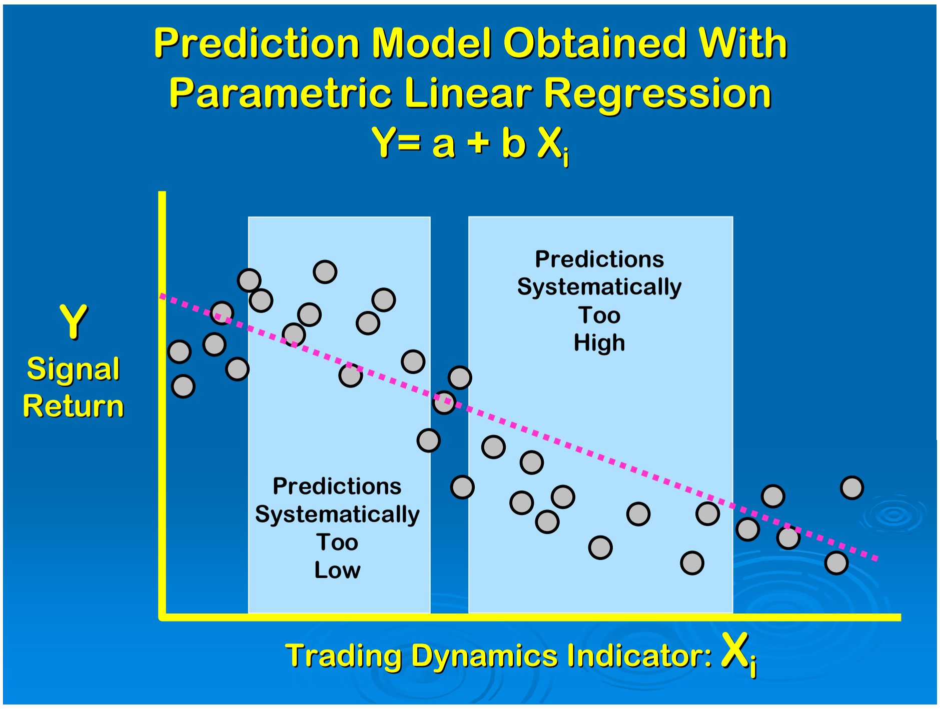
In the next diagram the sample of signals provided by the client is shown. Each datum represents a single signal. The position with respect to the vertical axis is the return earned by the signal and the position with respect to the horizontal axis is the value of indicator X_i at the time of the signal was given. For purposes of clarity, the data was created to show an obvious relationship between X_i and signal return (i.e., the indicator is strongly predictive of signal returns. In practice indicators this potent are unlikely to exist.



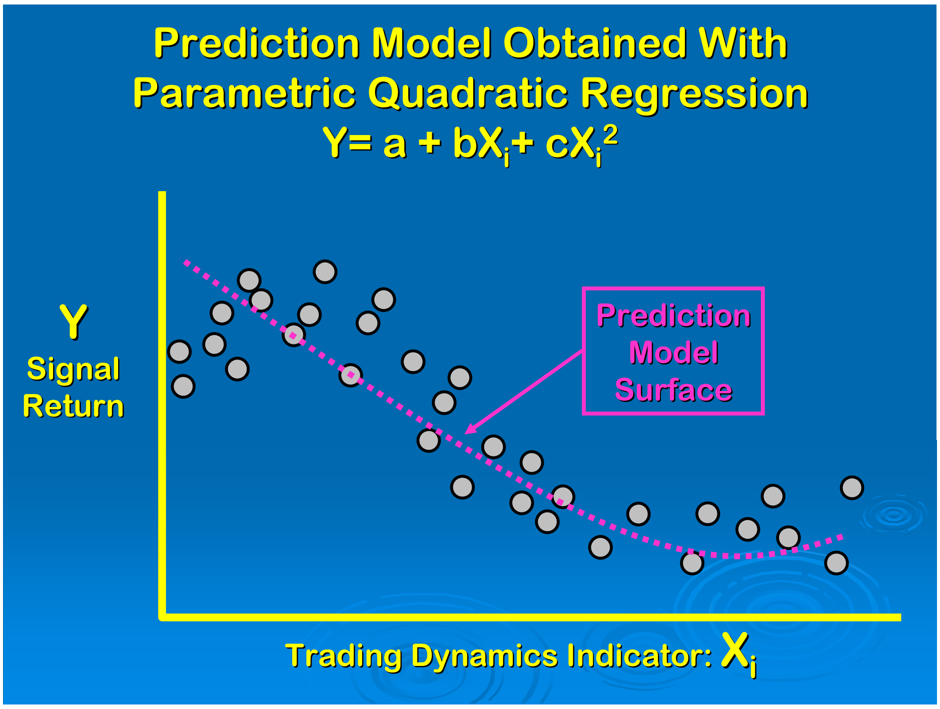
If one were to model this data with linear regression the diagram below depicts the model surface that would be obtained.



The linear model is too simple and wrong in a systematic sense. That is to say the assumption that the model surface is flat throughout the range of the predictor variable X_i causes the model to make systematic errors (i.e., the model is biased). In some ranges of X_i the model's predictions of signal return are systematically too low while in other ranges of X_i the predictions are systematically too high. Note that systematic errors are different from random errors that show no consistent pattern to the way the model is wrong. Systematic errors are symptomatic of a model surface that does not accurately represent the underlying functional relationship between the indicators and signal return. The illustration shows regions of Xi where the linear model makes systematic errors.



A reasonable solution would be to propose a more complex model by visual inspection, such as a parabola. While as the illustration below shows, the parabola does fit the data better, visual analysis is not a practical solution when predictive power of the indicator is poor or there are multiple predictors involved.



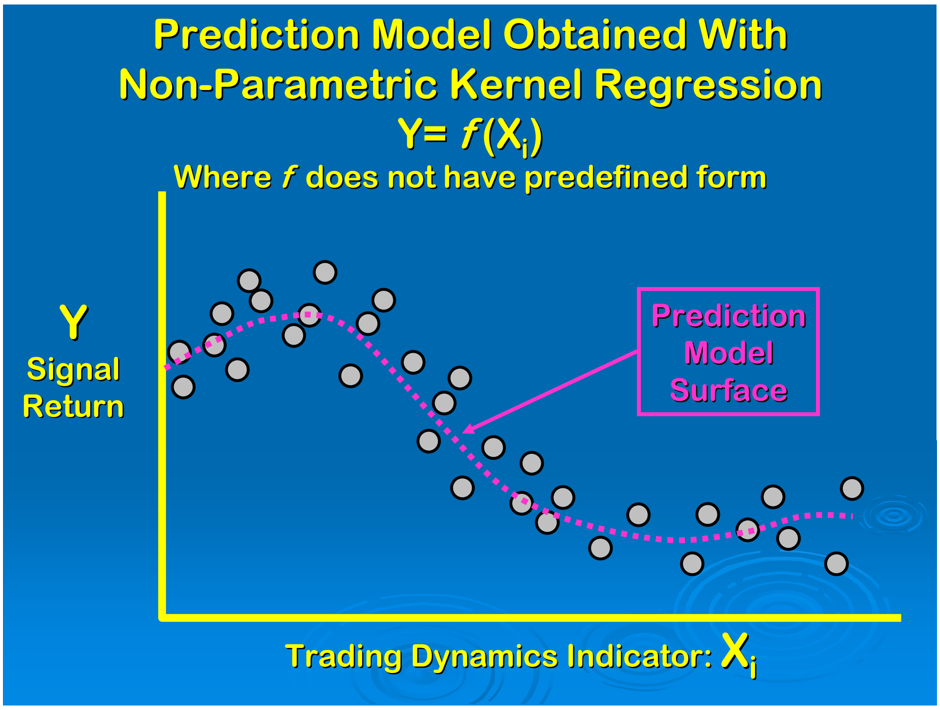
It should be noted that in the case of a linear model, a quadratic (parabolic) model, a cubic model where Xi is raised to the third power, or any model where the functional relationship is proposed (assumed) prior to the analysis, the discovered model is constrained to adopt the assumed form. This approach, which is called parametric[11] modeling, is perfectly legitimate when there is well established theory that suggests what the correct functional form should be. This is the case in many scientific applications.

However, for many phenomena characterized by high complexity and high randomness, such as financial markets, there is no well established theory and thus no sound rationale for assuming any particular functional form. In these situations constraining the analysis to an assumed functional form is too limiting. What is needed is an approach to modeling that eliminates the need to assume the shape of the model's surface prior to analysis. This

[11] It is called parametric modeling because once the functional form is assumed the only remaining task is to estimate the values unknown parameters of the assumed equation. For example in the case of a linear model the unknown parameters are the constant (Y-intercept) and the weight for each of the predictor variables.

allows the correct form to be discovered unencumbered by unwarranted assumptions. This is the domain of non-parametric[12] modeling.

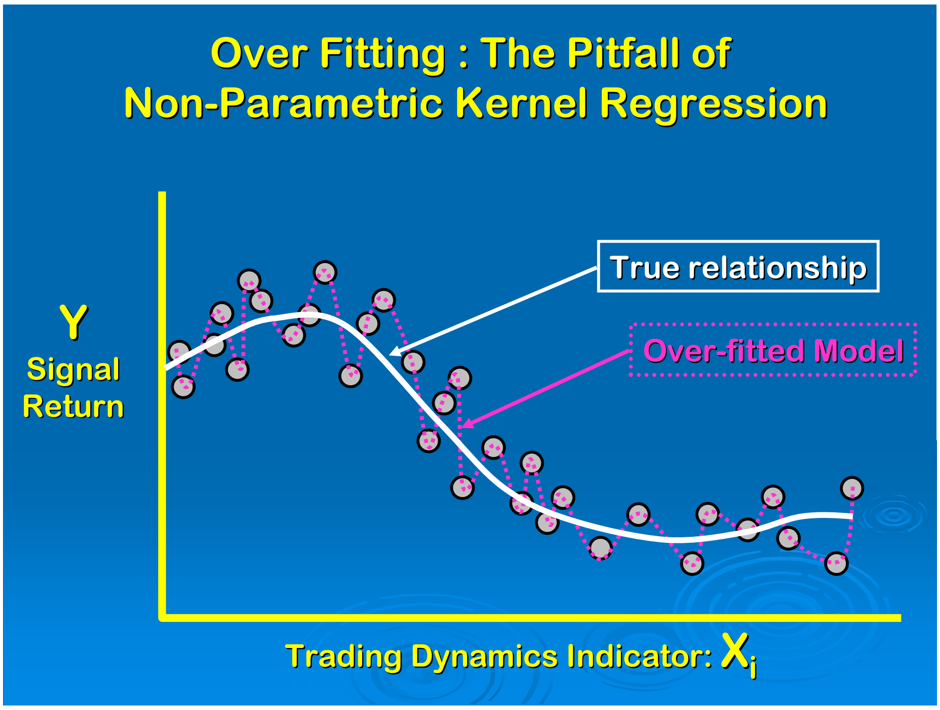
One type the several types of non-parametric modeling employed by Hood River in its performance-boost modeling is *kernel regression*. The illustration below shows the model surface that would be obtained by applying kernel regression to the sample data. Note the discovered model surface conforms closely to the true relationship depicted in the earlier illustration. Put in simple terms, kernel regression discovers the correct shape of the surface by estimating the value of the dependant variable Y (signal return) within small or "local" ranges along the X_i axis. The simplest approach to kernel regression takes an average of the Y values in each little "local" X_i neighborhood. This becomes the altitude of the model surface in that region of X_i. A more sophisticated kernel method fits linear models to each small X_i neighborhood.



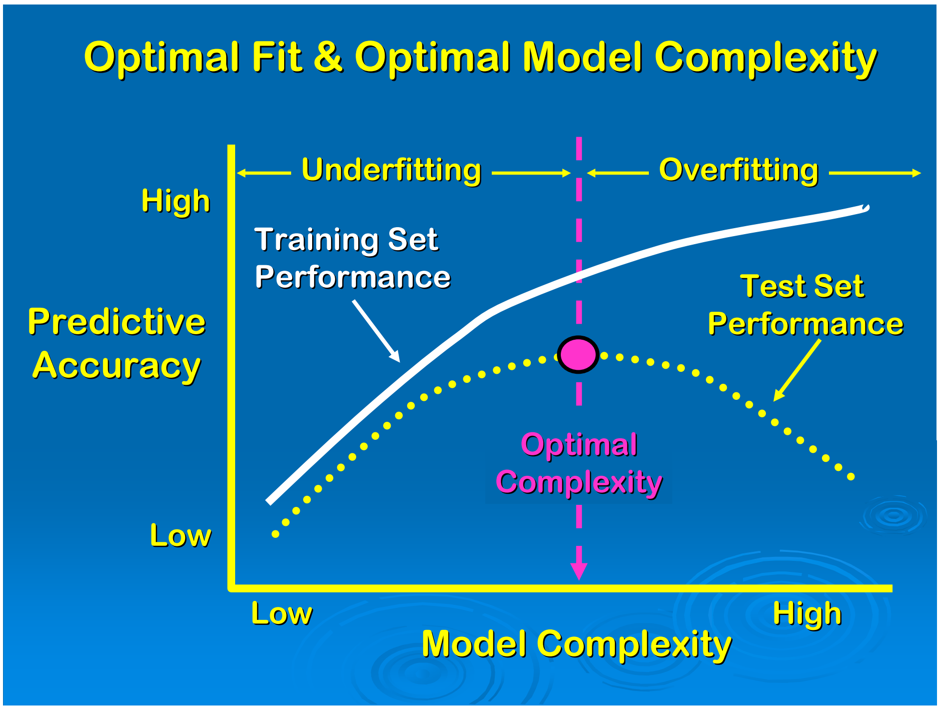
[12] The term "non-parametric" refers to the fact that there is no need to assume any particular parametric form prior to modeling the data. Parametric models, like linear, parabolic, that can be written as compact algebraic expression, have fixed shape with only the parameters of the function allowed to vary.

Kernel-regression belongs to a family of advanced modeling methods that also includes; neural networks, decision-trees, support-vector machines, regression-splines and host of other methods developed within the past several decades. Each approaches the search for a best model in a somewhat different way.

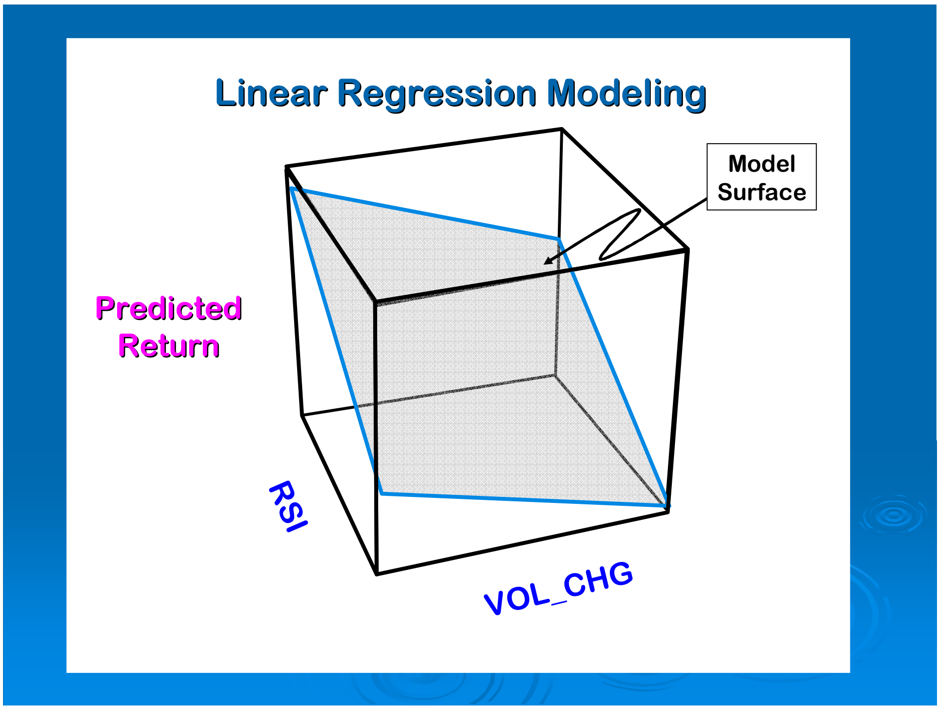
A potential pitfall of these more advanced modeling methods stems from their primary strength – flexibility. While flexibility is desirable in that it allows the model surface to conform to the true underlying relationships in the data, if not properly constrained, flexibility can cause the model to fit the data too closely, a problem called overfitting. When overfitting takes place, the model's surface becomes contaminated with random effects in the data. As a consequence, the surface not only describes the authentic relationship between signal returns and indicator reading but also describes the random variation in the particular sample of data used to develop the performance-boost model. This is depicted in the illustration below.

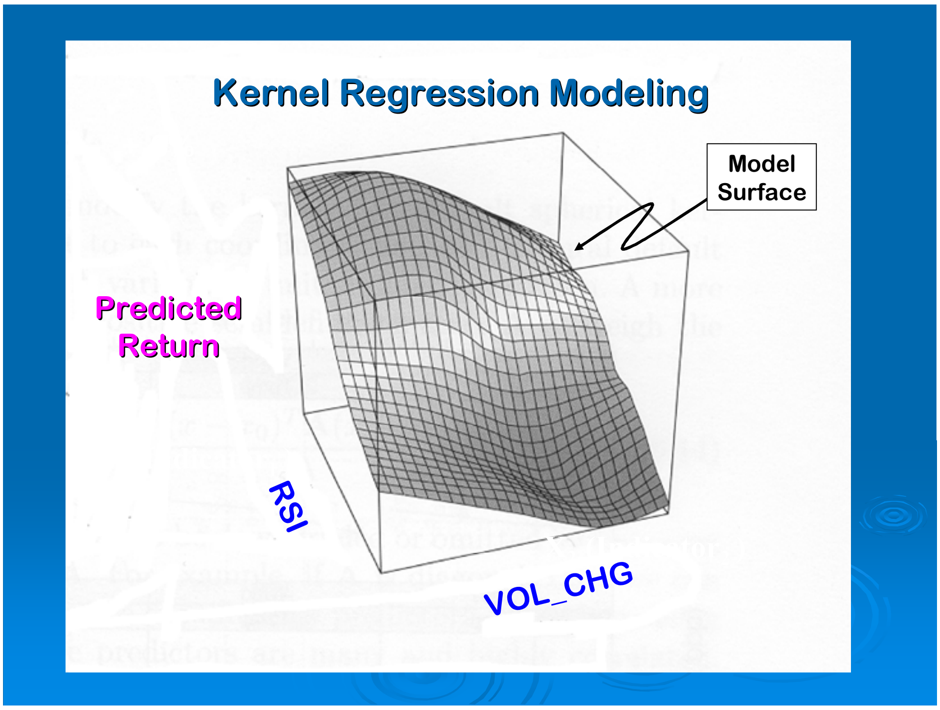


The problem of over fitting can be mitigated by using a method called cross-validation. This involves breaking up the historical sample into several subsets: training, testing and evaluation. Starting with the simplest possible model, a single indicator and a flat model surface, the modeling procedure tests a progression of ever more complex models utilizing more indicators and surfaces that bend ever more closely to the data. All the while the modeling procedure is cycling back and forth, or cross-validating, between the training and testing sets to discover the degree of complexity that yields the most accurate predictions. In other words, the training and testing subsets are used to discover which of the candidate indicators warrant a place in the model space and how much the model's surface should be allowed to bend in order to fit the data without the surface becoming over fitted. Overfitting is detected when an increase in model complexity improves fit on the training set but degrades fit on the testing set. When the model of optimal complexity has finally been discovered, then and only then, it is applied to the evaluation data. Because this third subset of data did not participate in the modeling process, it provides an unbiased estimate of the model's true efficacy. The notions of optimal model complexity, overfitting, and underfitting is illustrated in the diagram



For clarity the foregoing illustrations were confined to a single predictor variable. In practice the number of predictors is larger. Below we illustrate the difference between a two predictor linear model and a two predictor kernel regression model.


Linear Regression Modeling
Model Surface
Predicted Return
RSI
VOL_CHG


Kernel Regression Modeling
Model Surface
Predicted Return
RSI
VOL_CHG

Because financial markets are themselves complex nonlinear systems it is likely that the governing predictive relationships will also be non-linear. Limiting oneself to linear modeling imposes a significant constraint on what can be discovered from a given set of data. Nevertheless, linear regression should not be discarded out of hand as it can prove to be a valuable method when combined with non-linear methods. Hood River uses both linear and non-linear methods in its performance boosting research.

Prediction Model Ensembles

In fact, there are numerous advanced modeling procedures each utilizing a somewhat different mathematical paradigm and search procedure. Because of these differences, if a multitude of modeling methods are applied to the same set of signals the boosting models produced will differ and tend to make prediction errors that differ (i.e., uncorrelated). This leads to a key idea behind Hood River's performance boosting approach, ensembles of prediction models.

Just as in investing it makes sense to combine securities whose returns are uncorrelated into a portfolio, in performance boosting it makes sense to combine the predictions of an ensemble of models whose prediction errors are uncorrelated. In other words, the degree of volatility of returns experienced by a portfolio of securities is analogous to the size of the prediction errors made by a combined forecast. When the predictions are combined the errors tend to negate one another resulting in a combined prediction that has a smaller average error. This fact has been well established within the field of predictive modeling[13].

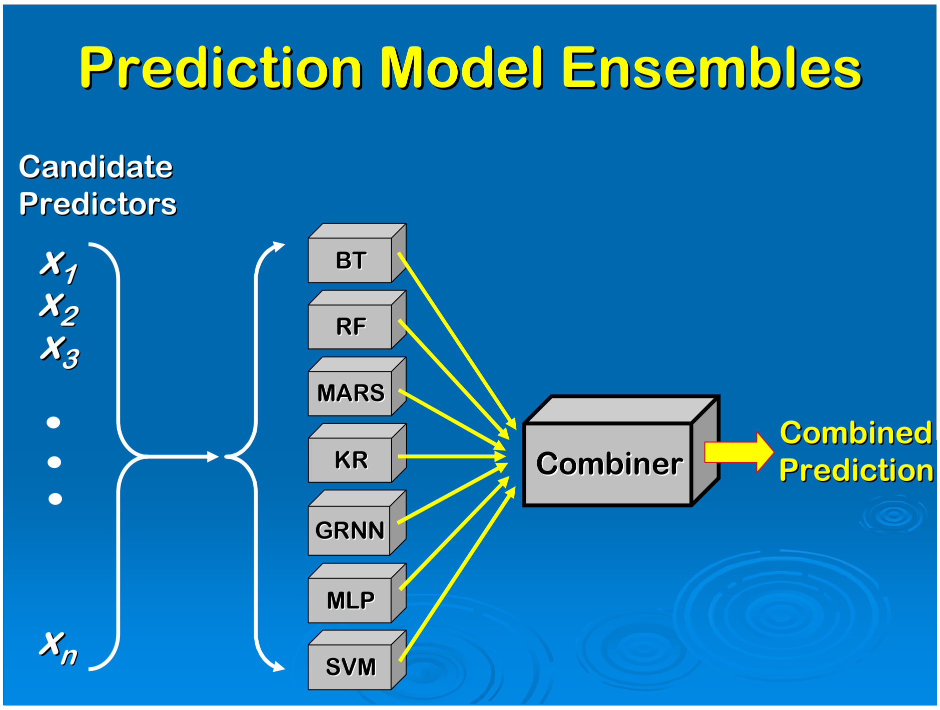
Two recently developed modeling methods used by Hood River, Boosted Trees and Random Forests, are based on the ensemble concept. However, these methods combine the predictions of numerous models of the same

---

[13] **On Combining Forecasts: Some Extensions and Results**, Anthony E. Bopp *Management Science*, Vol. 31, No. 12 (Dec., 1985), pp. 1492-1498; J.Scott Armstrong, form Long Range **Forecasting: From Crystal Ball to Computer**, John Wiley and Sons, 1985

modeling paradigm, in this case the recursively generated binary-tree, a popular data mining method[14].

While ensembles based on the same paradigm are effective Hood River has taken the concept of ensembles to a higher level to derive further improvements in accuracy. We create ensembles based on different modeling paradigms. The variety of methods include: boosted trees (BT), random forests (RF), multivariate regression by splines (MARS), kernel regression (KR), generalized regression networks (GRNN), multi-layer feed-forward perceptron or neural network (MLP), SVM (Support Vector Machine. The diagram below gives a sense of how a prediction ensemble is assembled and the combined forecast is derived.



The appropriate method for combining the set predictions issued by a set of models depends on whether the component models are intended to predict signal return or intended to predict the signal's class (i.e, gain or loss). If signal returns are being predicted, something as simple as a simple average of the predictions can be surprisingly powerful. A somewhat more

[14] These methods induce a multitude of models with independent errors by submitting different random subsets of the data to the modeling algorithm.

sophisticated approach is *constrained linear combination*. Here the component models are allowed different levels of influence. Finally, one can take into account the fact that models may perform with different accuracy in different regions of the model space. Several sophisticated algorithms exist for handling this situation.

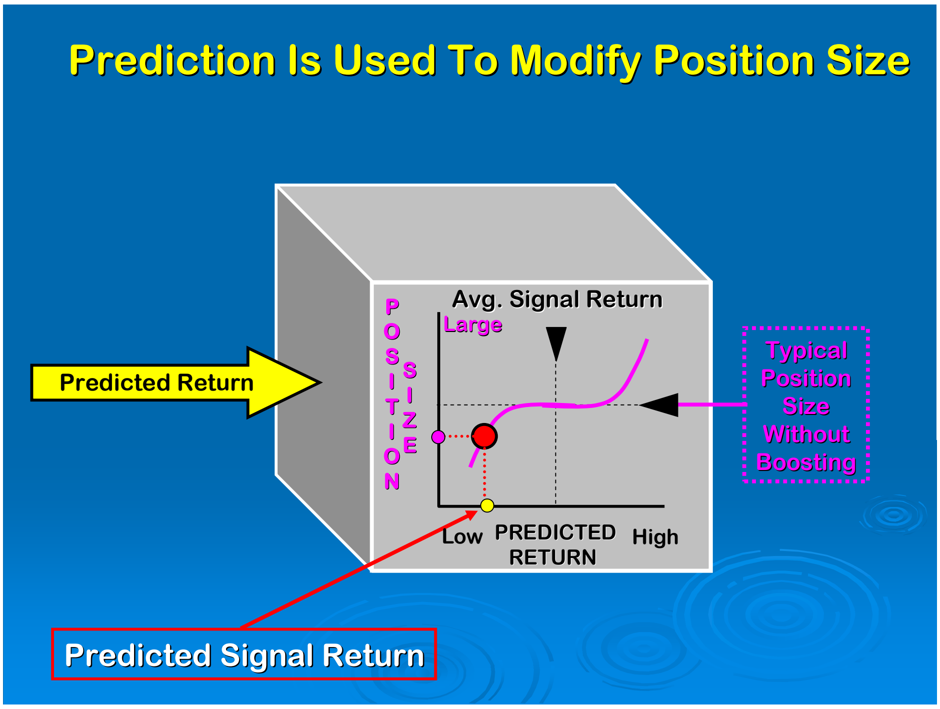
If, on the other hand, the models are intended to predict the signal's class a wide variety of ensemble techniques are available. Simple voting schemes can be extended using a method called Borda counts. A type of regression modeling called *logistic regression* can be an effective method means of combining these kinds of predictions. If the classification scheme involves a multitude of classes, such as big profit, average profit, average loss, big loss, a technique known as pair-wise coupling can be used. Finally, a technique called maximizing the fuzzy integral provides an enormously powerful technique for combining models that predict class using a sophisticated algorithm from fuzzy set theory.

What Clients Need to Provide

Performance boosting by Hood River does not require that the client reveal how their proprietary buy/sell algorithm works or that they alter it in any manner. What is required is as large a sample as possible of the strategy's prior signals, including the date of the signal, the name of the security, the type of position that was initiated by the signal (long or short) and the percent return of the earned at position exit, either in absolute terms or relative to a benchmark, depending upon which is of interest. In addition, the client needs to provide the same raw historical price and volume files for the stocks in their universe of interest.

Boosting Strategy Returns

Predictions issued by the Hood River boosting model are used to enhance the strategy's performance by altering the size of the position normally taken. If a particular signal is predicted to earn a higher than average return or has a higher than average probability of being profitable a larger amount of capital would be allocated to that signal. Conversely, if the signal is predicted to produce a below average or negative return a reduced position size would be warranted or no position altogether. This is illustrated in the diagram below. Note that the signal is predicted to have a below average return, thus justifying a smaller than position size than one typically taken in absence of a boosting model.

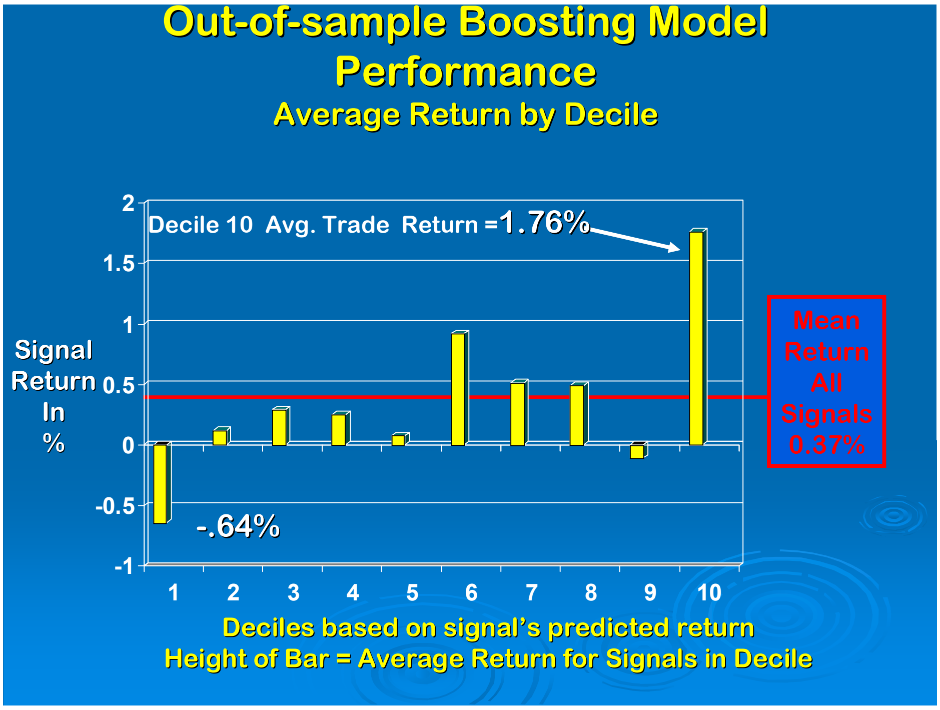


The boosting model can also help scale up a strategy. For example, assume that only the top 10% ranked stocks by the strategy are used for taking positions. It may be possible that stocks in the top 20% or 30% can produce attractive returns gains when signals are augmented with the boosting model's prediction.

Boosting Case Study

The following is an example of applying the boosting process to a two-factor technical strategy that takes long positions in stocks. The strategy is based on two factors: a stock's recent price change normalized by the stock's volatility and its recent change in trading volume. Called the C-K rule, it buys stocks that have recently fallen on declining trading volume. It is based on research by Michael Cooper and Lars Kestner.

The diagram shows the performance of the boosting model on data that was not used in the model's development (i.e., out-of-sample validation data). This is the acid test of a boosting model's efficacy. The boosting model was developed on an in-sample data set (1984-1999) and was then used to predict returns for signals in an out-of-sample data set (2000 - 2004). The out-of-sample signals were then grouped into ten groups or deciles based on the boosting model's predicted return. In the Figure below note that signals that were predicted to do the best (decile 10) did indeed have the highest average returns (+1.76%) and performed better than the average of all signals(+0.37%). The signals predicted to have the worst returns (decile 1) earned an average return of negative 0.64%. By taking smaller positions on signals predicted to do the worst or avoiding them entirely and taking larger than normal positions on signals predicted to do the best the overall strategy returns produced by the C-K rule would have been increased relative to a strategy of taking the same position size on all signals.



The Principals: David Aronson & Dr. Timothy Masters

David Aronson is an adjunct professor of finance at Baruch College, City University of New York, where he teaches a graduate level course in statistical market analysis and predictive modeling. He is a Chartered Market Technician and has been involved in the application of advanced data analysis and modeling to financial markets since 1982 when he founded Raden Research Group, an early adopter of machine learning financial market forecasting. He was a proprietary trader with Speae Leeds and Kellogg and founded AdvoCom, a firm that specialized in the evaluation of commodity trading advisors and hedge funds. He is author of the book *Evidence Based Technical Analysis: Applying the Scientific Method and Statistical Inference to Trading Signals* (John Wiley 2006).

Dr. Timothy Masters, a Ph.D in statistics, has been active in the application of advanced data modeling and machine learning methods to various areas of prediction including military, medical and financial applications since 1978. Dr. Masters has developed a suite of proprietary software that is used by Hood River in its performance boosting service. He is author of four books: *Practical Neural Network Recipes in C++* (Academic Press 1993), *Signal and Image Processing with Neural Networks* (John Wiley 1994), *Advanced Algorithms for Neural Networks* (John Wiley 1995a), *Neural, Novel, and Hybrid Algorithms for Time Series Prediction* (John Wiley 1995b)