



School of Economics and  
Commercial Law  
GOTHENBURG UNIVERSITY

**-Department of Informatics-**

2003 05 30

# **Enhancing Trading with Technology**

## **-A Neural Network-Expert System Hybrid Approach-**

### **Abstract**

The identification of specific patterns in stock price derived from technical stock analysis heuristics, which after occurring resulted in a predefined price movement, was the subject of this research effort. The motive was to enhance the profitability of an investment method based on such patterns. To identify the specific patterns resulting in the predefined price movement, artificial neural networks were used. A theoretical model for combining an expert system, filled with knowledge from technical stock analysis heuristics, and artificial neural networks into a hybrid integrated system was presented. Experiments were then conducted in order to evaluate whether the proposed method actually could improve the profitability of the selected investment method. Neural networks were trained in these experiments to classify whether the outcome of an occurred pattern would result in a predefined price movement. The major findings of this research was that; using the proposed method we were able to enhance some specific patterns used in technical analysis, occurring in the Swedish OMX-index during the specific period.

**Keywords: Artificial Intelligence; Expert System; Hybrid System;  
Neural Network; Technical Analysis; Trading;**

Masters Thesis: 20 Swedish Credit Points  
Author: Daniel Liljeblad  
Supervisor: Faramarz Agahi  
Examiner: Rikard Lindgren

# Contents

Abstract.....	1
List of Figures.....	3
List of Formulas .....	3
List of Tables .....	3
1. Introduction .....	4
1.2. Research Definition.....	5
1.3. Research Question.....	5
1.4. Objective .....	6
1.5. Scope and Limitations .....	6
1.6. Disposition.....	6
2. Theory.....	7
2.1. Artificial Neural Networks .....	8
2.1.1. Architecture and Learning .....	11
2.2. Expert Systems .....	14
2.2.1. The Knowledge Domain: Technical Stock Analysis .....	15
2.3. A theoretical model for integrating Expert Systems and Artificial Neural Networks ..	21
2.4. Literature Review .....	22
2.4.1. Applications of Neural Networks in finance.....	22
3. Method.....	26
3.1. Research type.....	26
3.1.1. Experiment Setup .....	26
3.1.2. Neural Network Setup .....	28
3.1.3. Experiment Execution.....	29
3.2. Data collection and Data analysis .....	29
3.2.1. Performance metrics .....	30
3.2.2. Statistical tests .....	32
3.3. Inference .....	34
3.4. Reliability and Validity .....	35
4. Analysis of Results.....	36
4.1. The reversal day up pattern .....	36
4.2. The reversal day down pattern .....	38
4.3. The inside day pattern .....	39
5. Conclusion and Discussion.....	41
5.1. Conclusion.....	41
5.2. Discussion.....	41
5.3. Further research.....	42
References .....	43

## List of Figures

Figure 1: A model of a biological neuron (Aronson & Turban, 2001, p. 608). .....	9
Figure 2: A model of an artificial neuron (Haykin, 1999, p. 11) .....	9
Figure 3: A graph of a sigmoid activation function (Haykin, 1999, p. 13).....	10
Figure 4: Illustration of a multilayer perceptron network. ....	11
Figure 5: The learning process using backpropagation (Euliano et al, 2000) .....	11
Figure 6: Illustration of patterns in bar chart form. ....	16
Figure 7: Illustration of a trading chart.....	19
Figure 8: A model of an ES & ANN hybrid system. ....	22
Figure 9: An example of a pattern matching procedure (Lee & Liu, 2001).....	24
Figure 10: A decision tree for the inside day pattern. ....	32

## List of Formulas

Formula 1. The sigmoid function (Haykin, 1999, p. 13).....	10
Formula 2: The Mean Square Error formula. ....	12
Formula 3: Weight update algorithm (Haykin, 1999, p. 175).....	13
Formula 4: The formula for the statistic comparison test. ....	33
Formula 5. The formula for the confidence interval calculation. ....	33

## List of Tables

Table 1: The Reversal Day Up Pattern. ....	17
Table 2: The Reversal Day Down Pattern. ....	18
Table 3. The Inside Day Pattern. ....	19
Table 4: Explanation of the confusion matrix. ....	30

# 1. Introduction

The financial markets have over the years been the subject of intense research, many are those that have tried to develop methods or models of consistently beating the market. Many different techniques have been developed for forecasting and prediction of equities and futures instruments. Common classifications of these techniques are into fundamental and technical based analysis methods. Schwager (1994) states that to survive in today's turbulent financial markets one has to possess an edge over other speculators. This research investigates a combination of knowledge derived from seasoned speculators and the predictive ability of artificial neural networks to provide this edge.

The methods on which the investment philosophy used in this research relied on, comes from technical stock analysis (TA), which is a method of deriving the future direction of a securities price by studying its past price history. The reason for using technical analysis instead of the more widely accepted fundamental stock analysis, which is the classical way at deriving at a securities value by analysis of the underlying company's value, is because the relative ease of translating the rules and variables used in TA into computationally feasible inputs.

Systems used to support decision making using a predefined set of rules are called Expert Systems (ES) (Aronson & Turban, 2001). An ES is filled with knowledge of some usually narrow domain, by extracting this knowledge from human experts. This knowledge is then applied to real world problems by a user of the system to provide guidance of the task at hand. Thus the predefined rules from TA used to determine when to invest that are used in this research, can be regarded as an ES.

The technology of artificial neural networks (ANN's) has emerged from the field of artificial intelligence. Artificial neural networks can be thought of as software implemented models of the human brain. Thus artificial neural networks try to mimic the human ability to learn from experience or training, making it in theory, almost an ideal way to apply machine learning to technical analysis based trading. Artificial neural networks have been successfully applied to many business applications ranging from detecting payment card fraud (Hassibi, 2000) to predicting business bankruptcy (Boussabine & Wanous, 2000). In financial domains neural networks have outperformed traditional statistical techniques in forecasting and predicting stock and index returns (Refenes & Francis, 1995). Neural networks can be viewed as heuristic<sup>1</sup> procedures best applied where:

1. One can specify particular influences on a phenomenon whose outcome is known with certainty.
2. The relationship can not be described.
3. The relationship is not necessarily linear.
4. There are no known models.

This research will analyze and evaluate if an expert system based on some commonly known patterns used in technical analysis, can be further enhanced regarding improvement in predictive accuracy and profitability, by the use of artificial neural networks.

---

<sup>1</sup> Informal, judgemental knowledge of an application area that constitutes the "rules of good judgement" in the field.

## 1.2. Research Definition

The aim of this research is to evaluate whether a combination of an ES, based on specific technical analysis heuristics, and ANN's can provide an increase in predictive accuracy and profitability over the sole use of the ES.

Patterns in stock price history, which according to technical stock analysis literature may have some influence on the future price direction, are to be classified where the goal is to identify those particular observations of the patterns which actually precede a predefined desired goal in price development. To find these particular instances the neural network examines the provided input, to find any relationship within this input that denotes the phenomenon of a pattern that is to meet a predefined goal. The motive is of course, to improve the investment method of using such patterns regarding profitability. The research is conducted on historical price data on the Swedish OMX-Index from 1987 10 08 to 2003 03 18, the OMX is a weighted index of the 30 most traded stocks on the Stockholm Stock Exchange.

Thus, first a definition of a desired goal that is to follow an occurrence of a pattern is formulated. For example, following a specific pattern we desire a 3 percent price increase within 5 days, if this occur following a pattern the outcome of the pattern is defined as good, if this does not occur the outcome is bad. So the outcome of every pattern is either good or bad, based on the definition of the patterns goal. This should not be mixed with the definition of the pattern itself, which is purely a description of what constitutes the pattern, or how it is identified. When all the observations of the selected form of pattern are gathered, every particular observation is coded as either "Good" or "Bad" depending on if it succeeds in meeting its desired goal. Then the distribution of the patterns probability of meeting its desired goal can be calculated. For example 25 percent of all the patterns of the specific form meets its desired goal, which in turn is a 3 percent price increase within 5 days.

The network is now trained to find any relationship in the selected input variables, which may influence the phenomenon of a good observation of the selected form of pattern. After training, the network is tested on unseen patterns and the percentage distribution of the networks classifications is compared to the percentage distribution of the sole use of the pattern, to evaluate whether the neural network can enhance this pattern.

## 1.3. Research Question

The research question stated in this research is based on theoretical assumptions of the feasibility of neural networks as classifiers on trading patterns derived from technical stock analysis. Neural networks have shown the ability to correctly classify a desired outcome based on relationships in some input variables, given that there is a predictive relationship of these input variables on the desired outcome. In this research it is assumed that the selected input variables derived from technical stock analysis, may have a predictive relationship regarding the outcome of the selected patterns in stock price, these assumptions are more thoroughly described in the theory section.

The primary and secondary research questions are formulated in the following way:

- **Can an artificial neural network provide an enhancement to specific technical rule based trading patterns?**
- **If any, how large improvement can be expected.**

Where the definition of enhancement is regarding to, improved sensitivity of the good patterns, sensitivity is defined as the proportion of events labeled as that class (good or bad) which are correctly detected. Profitability will also be measured by expected payoff according to the decision tree technique (figure 10). In the case of the sole use of the rule based trading patterns, used by the expert system, all found observations of the patterns are labeled as good.

## 1.4. Objective

The purpose of this research is to evaluate whether static technical trading rules can be significantly enhanced with the help of an artificial neural network. The primary interest in this research is the avoidance of false buys, thus it is viewed as better to pass up a chance of investing than to make an erroneous investment. The drawback is often that a lot of potentially good investment opportunities are passed up when trying to maximize the probability of a good investment. We will want to find a balance between this loss of opportunity and performance.

If the combination of technical stock analysis heuristics and artificial neural networks would prove useful in classifying whether certain patterns in index price would result in favorable investing opportunities, benefits of such a system when connected to a financial real-time trading system could be:

1. Screening of a very large number of equities, and automatically buying when detected opportunities present them selves.
2. Adapting to changes in market behavior and investing selection criteria in real-time.
3. Automatically performing portfolio optimization and derivative hedging.

Conventional systems based solely on predefined rules would not be able to quickly adapt to changing conditions without the need of substantial reprogramming and user involvement.

## 1.5. Scope and Limitations

The potential combinations of different financial market indices and stocks are almost limitless. The same is true for possible combinations of artificial neural network architectures as well as methods used in technical stock analysis. Therefore this research is limited to one stock index, one artificial neural network architecture and three specific patterns. When evaluating the profitability of the selected patterns, transaction costs and slippage<sup>2</sup> is disregarded. This research is also limited to the availability of historical price data, only data from 1987 to 2003 in the Swedish OMX-Index was available to the researcher.

## 1.6. Disposition

In section two, the theory behind artificial neural networks and expert systems as well as the domain of technical analysis is explained. A literature review highlighting important earlier research efforts and a theoretical model for integrating expert systems and artificial neural networks is presented. The methods used in this research are described in section three. The selected research type and process along with an explanation of the data analysis and selection procedure used are described. The inference procedure as well as reliability and validity is also discussed. In the fourth section, the results are analyzed for the three selected patterns and the fifth and final section concludes this research along with a more general discussion and some recommendations for further research.

---

<sup>2</sup> Losses due to insufficient market liquidity.

## 2. Theory

This section is aimed at providing the reader with a basic understanding of artificial neural networks and expert systems as well as the specific technical stock analysis heuristics used in this research. A theoretical model is introduced for how to combine technical stock analysis heuristics and neural networks into a hybrid system. Prior research of the applications of neural networks in finance is then reviewed.

According to Hawley, Johnson and Raina (1996) neural network systems are most effectively applied to pattern recognition tasks, such as classification. Classification involves the assignment of input to predefined groups or classes based on patterns that exists in the input data. One example of this is optical character recognition, where the inputs are hand written characters, which are to be classified into the predefined alphabetical equivalents.

When describing artificial neural networks several mathematical formulas are needed to highlight the different processes describing how these networks learns and operates. It may seem frightening for those with a weak mathematical foundation, but this is nevertheless necessary since many of the variables in the equations are the objects of alteration when designing the neural networks. A detailed understanding of the formulas are not needed to comprehend the basics of artificial neural networks, this can be understood by following the written explanations and illustrations.

The literature review is aimed at giving the reader an understanding of how actual neural network systems have been applied to the domain of finance, in earlier research. The research area of neural networks applied to finance is fairly new, emerging in the mid 1980's. The method of combining technical stock analysis heuristics, such as patterns in stock price, and ANN's, is currently emerging, only three articles, with an in depth description of the techniques used were found (in march 2003) , using the article database at the University Library of Gothenburg . Of these three articles two radically different methods are presented for incorporating pattern rules from technical stock analysis into a neural network system. Approximately forty articles referring to ANN's applications to finance has been thoroughly examined by the author of this research; ten of these articles have been selected and briefly described to highlight important methods previously found successful.

## 2.1. Artificial Neural Networks

According to Azoff (1994) neural networks can in a statistical context be regarded as nonlinear models that can be trained to map past and future values of a time series, and thereby extract hidden structure and relationships governing the data. Traditional statistical methods such as regression techniques, are based on a linear relationship governing the input to objective function, if this relationship is not linear as may be the case of stock price fluctuations, these methods do not fully apply.

In a context of traditional statistical methods neural networks can be described as: *multivariate nonlinear nonparametric inference technique that is data driven and model free* (Azoff, 1994, p. 1). Multivariate refers to the neural network input comprising many different variables whose interdependencies and causative are exploited in predicting future behavior of a temporal sequence. Nonparametric, model free is a consequence of the lack of any presumptions regarding the relation between input variables and extrapolations into the future. The network is trained by adoption of free parameters to discover any possible relationships, devoid of model constraints, that is driven and shaped solely by the input data. In a statistical model sense, nonparametric means the fact that no predetermined parameters are required to specify the mapping model. The free parameters are weights associated with the signal communication lines between the neurons (described below), which attenuate the passing signals or data.

Thus neural networks are non parametric, non linear models that can be trained to map past values of a time series, for purposes of classification or prediction. The properties of such a network, a feed forward neural network with back propagation learning will be described and defined mathematically to provide a thorough understanding of the technology. Feedforward networks trained with backpropagation are the most common (Chien-Hui & Fadlalla, 2001) type of neural networks, used in a wide variety of areas and known for its generalization ability.

Haykin (1999, p. 2) defines a neural network in the following way:

A neural network is a massively parallel distributed processor made up of simple processing units which have a natural propensity for storing experiential knowledge and making it available for use. It resembles the human or animal brain in two respects:

1. Knowledge is acquired by the network from its environment through a learning process.
2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.

Accordingly, a neural network is a model that mimics a biological neural network, such as the human or animal brain. The human brain is composed of billions of cells called neurons. These neurons function in groups called networks; the brain can thus be viewed as a collection of neural networks. Our memory and ability to learn are stored in these networks and preserved because the neuron cells, unlike other cells do not die.

There are according to Aronson & Turban (2001) approximately hundred different types of neuron cells, figure 1 show a simplified version of the basic properties of a biological neuron.



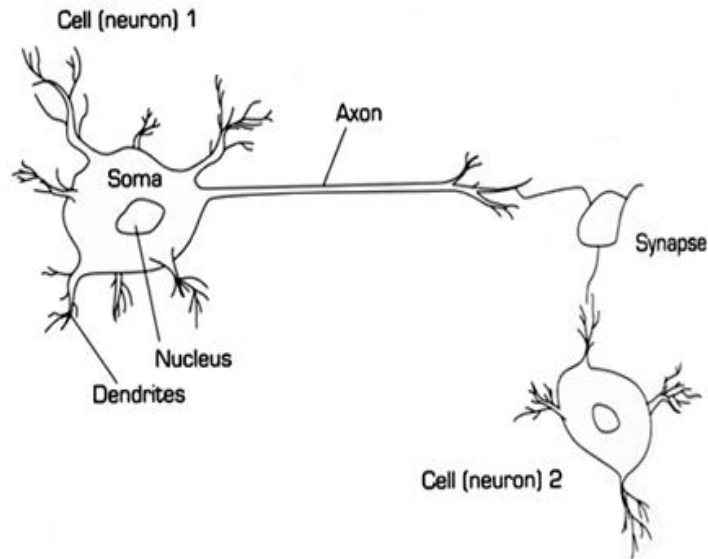


Figure 1: A model of a biological neuron (Aronson & Turban, 2001, p. 608).

The neuron receives input stimuli from other neurons through its dendrites, the axon provides an output connection to other neurons. The axons output signal passes through a synapse before it is transferred to the other cells dendrites. The synapse can alter the strength of the signal before it is passed over to other cells. It's in the synapse that the information is stored.

The artificial counterparts use a very limited set of concepts based on our knowledge of biological neural networks. These concepts are used to build software based models of processing elements (PE's), the equivalent of neurons, interconnected in a network architecture.

The basic properties of an artificial neuron according to Haykin (1999, p.11) are shown in figure 2 below:

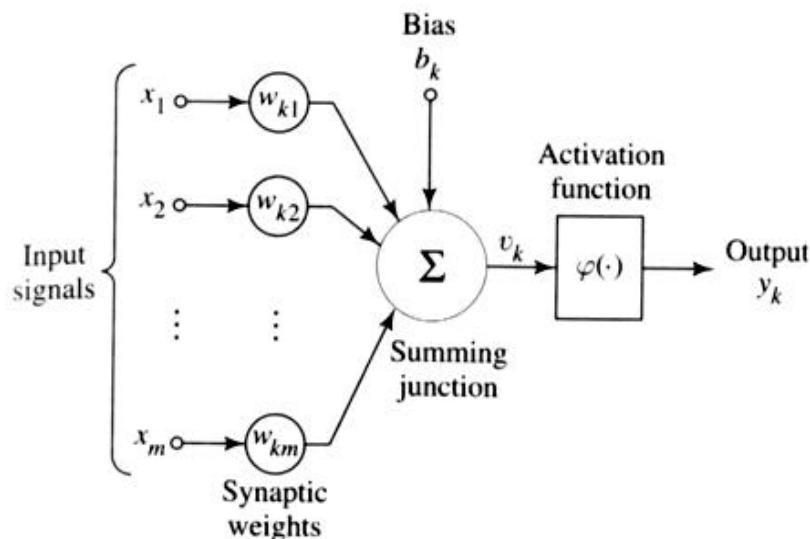


Figure 2: A model of an artificial neuron (Haykin, 1999, p. 11)

This model is composed of the following elements:

A set of synapses or connection links, each characterized with a weight of its own. An input signal  $X_j$  at the input of synapse  $j$  connected to neuron  $k$  is multiplied by the synaptic weight  $W_{kj}$ , where the first subscript ( $k$ ) refers to the neuron in question and the second subscript ( $j$ ) refers to the input end of the synapse of which the weight refers. The synaptic weights may lie in a range that includes positive as well as negative values.

The input signals are summed up at a summing junction or adder, where the inputs are weighted by the respective synapses.

An activation function is used to limit the possible amplitude range of the output of a neuron. Thus the activation function limits the permissible amplitude range to some finite value. It also provides nonlinearity to the neuron. A common activation function is the sigmoid function which is described in detail below in formula 1.

The model also includes a bias which can increase or lower the net input of the activation function, depending on whether it is positive or negative.

In mathematical terms, we may describe a neuron  $k$  by writing the following equations:

$$U_k = \sum_{j=1}^m w_{kj} X_j$$

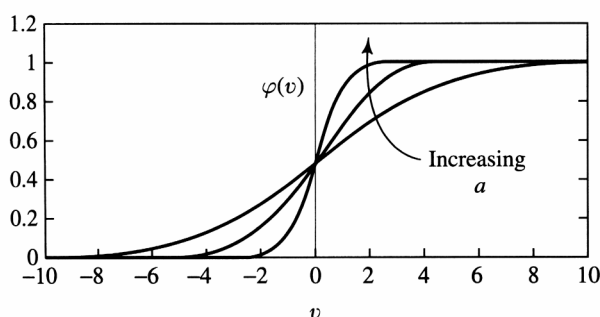
And

$$Y_k = \mathbf{j}(U_k + B_k)$$

The sigmoid activation function, whose graph is s shaped (figure 3) is the most common activation function used in the construction of artificial neural networks. An example of the sigmoid function is the logistic function defined by:

$$\mathbf{j}(v) = \frac{1}{1 + \exp(-av)}$$

**Formula 1. The sigmoid function (Haykin, 1999, p. 13)**



**Figure 3: A graph of a sigmoid activation function (Haykin, 1999, p. 13)**

Where  $a$  is the slope parameter of the sigmoid function. By varying parameter  $a$  one obtains sigmoid functions of different slopes (figure 3). The logistic sigmoid function assumes a continuous range of values between 0 and 1.

### 2.1.1. Architecture and Learning

The manner in which the neurons are structured constitutes the networks architecture. The most common type of architecture is the multilayer feedforward network also referred to as multilayer perceptrons (MLP's). This architecture distinguishes itself by the presence of one or more hidden layers of neurons. Figure 4, shows a neural network with 3 neurons in the input layer, 2 neurons in the hidden layer and 1 neuron in the output layer, or a 321 MLP network.

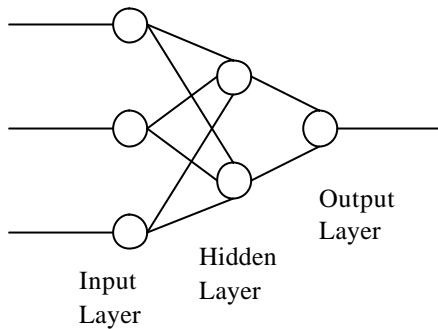


Figure 4: Illustration of a multilayer perceptron network.

Multilayer perceptron networks are trained with error correction learning which means that the desired response must be known. This is also called supervised learning since the networks output is compared to some known desired response. The most widely used learning algorithm for MLP networks is the Backpropagation algorithm.

Backpropagation essentially works in the following way, the weights in the network are first assumed to be initialized to some random values.

1. The input is fed forward through the network nodes until an output is acquired, this is the forward pass.
2. The output is compared to the desired response, and if the error is “acceptably low” according to our error criterion, the training is stopped. If not go to step three.
3. Move backwards through the network, changing the weights, and repeat step 1.

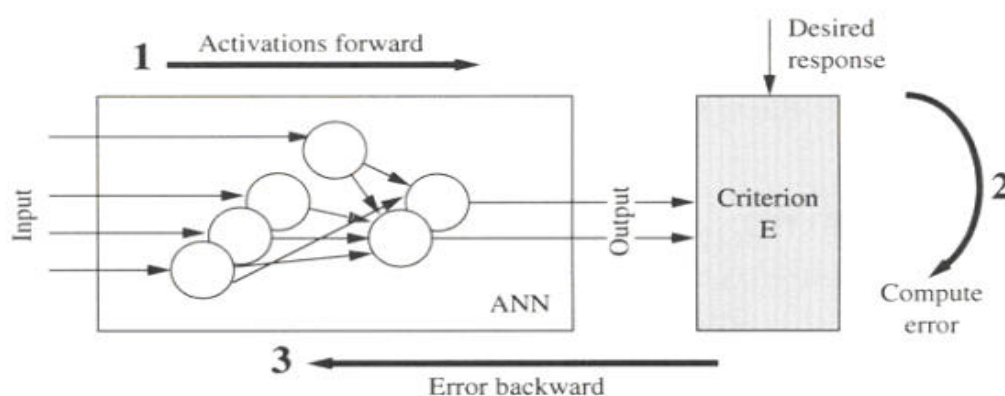


Figure 5. The learning process using backpropagation (Euliano et al, 2000)

The error criterion or cost function most widely used and also used in this research is the mean square error (MSE), that is, the sum of the square difference between the desired response and the actual output. The formula is:

$$MSE = \frac{\sum_{j=0}^P \sum_{i=0}^N (d_{ij} - y_{ij})^2}{NP}$$

**Formula 2: The Mean Square Error formula.**

Where: P = number of output processing elements, N = number of exemplars (observations) in the data set,  $y_{ij}$  = network output for exemplar  $i$  at PE  $j$ ,  $d_{ij}$  = desired output for exemplar  $i$  at PE  $j$ .

The goal of the classifier is to minimize this cost function by changing its free parameters (Euliano, Lefebvre & Principe, 2000).

We are now going to provide a mathematical outline of the backpropagation computer algorithm according to Haykin (1999).

The algorithm cycles through the training sample  $\{(x(n)d(n))\}_{n=1}^N$  as follows:

1. *Initialization.* Pick the synaptic weights and thresholds from a uniform distribution whose mean is zero.

2. *Presentations of Training Examples.* Present the network with an iteration (epoch) of training examples. For each example in the test set, ordered in some fashion, perform the sequence of forward and backward computations described under points 3 and 4, respectively.

3. *Forward computation.* Let a training example in the epoch be denoted by  $(x(n), d(n))$ , with the input  $x(n)$  applied to the input layer of neurons and the desired response  $d(n)$  presented to the output layer of neurons. Compute the induced local fields and function signals of the network by proceeding forward through the network, layer by layer. The induced local field  $v_j^{(l)}(n)$  for neuron  $j$  in layer  $l$  is:

$$v_j^{(l)}(n) = \sum_{i=0}^{m_0} w_{ji}^{(l)}(n) y_i^{(l-1)}(n)$$

Where  $y_i^{(l-1)}(n)$  is the output (function) signal of neuron  $i$  in the previous layer  $l - 1$  at iteration  $n$  and  $w_{ji}^{(l)}(n)$  is the synaptic weight of neuron  $j$  in layer  $l$  that is fed from neuron  $i$  in layer  $l - 1$ .

1. For  $i = 0$ , we have  $y_0^{(l-1)}(n) = +1$  and  $w_{j0}^{(l)}(n) = b_j^{(l)}(n)$  is the bias applied to neuron  $j$  in layer  $l$ . Assuming the use of a sigmoid function, the output signal of neuron  $j$  in layer  $l$  is:

$$y_j^{(l)} = \mathbf{j}_j(v_j(n))$$

If neuron  $j$  is in the first hidden layer (i.e.,  $l = 1$ ), set:

$$y_j^{(0)}(n) = x_j(n)$$

Where  $x_j(n)$  is the  $j$ th element of the input  $x(n)$ . If neuron  $j$  is in the output layer (i.e.,  $l = L$ , where  $L$  is referred to as the depth of the network), set:

$$y_j^{(L)} = o_j(n)$$

Compute the error signal:

$$e_j(n) = d_j(n) - o_j(n)$$

Where  $o_j(n)$  is the  $j$ th element of the output pattern which is compared to  $d_j(n)$  which is the  $j$ th element of the desired response  $d(n)$ .

4. *Backward Computation.* Compute the  $\mathbf{d}$  s (local gradients) of the network, defined by:

For neuron  $j$  in output layer  $L$ : 
$$\mathbf{d}_j^{(L)}(n) = e_j^{(L)}(n) \mathbf{j}'_j(v_j^{(L)}(n))$$

For neuron  $j$  in hidden layer  $l$ : 
$$\mathbf{d}_j^{(l)}(n) = \mathbf{j}'_j(v_j^{(l)}(n)) \sum_k \mathbf{d}_k^{(l+1)}(n) w_{kj}^{(l+1)}(n)$$

Where the prime in  $\mathbf{j}'_j(\cdot)$  denotes differentiation with respect to the argument.

Adjust the synaptic weights of the network in layer  $l$  according to the generalized delta rule:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \mathbf{a} [w_{ji}^{(l)}(n-1)] + \mathbf{h} \mathbf{d}_j^{(l)}(n) y_i^{(l-1)}(n)$$

**Formula 3: Weight update algorithm (Haykin, 1999, p. 175).**

Where,  $\mathbf{h}$  is the learning-rate parameter and  $\mathbf{a}$  is the momentum constant (these two parameters often need to be fine tuned to provide good generalization).

5. *Iteration.* Iterate the forward and backward computations under points 3 and 4 by presenting new epochs of training examples to the network until the stopping criterion is met.

According to (Back, Burns, Giles, Lawrence & Tsoi, 1998) it has been shown theoretically that MLP's approximate Bayesian a posteriori<sup>3</sup> probabilities when the desired output are one of many and squared-error cost functions are used, assuming: a large enough neural network, training finds a global minimum (meaning that the input data has a predictive relationship to the output to being classified), infinite training data and the a priori class probabilities of the test set are correctly represented in the training set. Back et al (1998) also states that MLP's have shown to accurately estimate the Bayesian a posteriori probabilities for certain experiments in practice. This means that in theory, given the assumptions above, a neural network is able to map input that have a predictive relationship to classify a phenomena into different classes.

---

<sup>3</sup> Bayes theorem has shown to transform prior probabilities of the occurrence of a certain event into posterior probabilities of the same event.

## 2.2. Expert Systems

Medsker (1994) defines an expert system as a system that performs reasoning using previously established rules for a well defined and narrow domain. The purpose of such a system is to imitate the reasoning process of a human expert and make the expert's knowledge available to a non expert, the user of the system. ES are usually domain specific in the sense that they are designed to solve or help solve problems related to specific tasks or occurrences. The system itself contains expertise knowledge derived from a human expert of the particular domain. The knowledge is encoded in some way, often as rules that can be expressed by AND, IF and THEN statements.

The basic properties of an ES according to Aronson & Turban (2001) are the knowledge base, inference engine and user interface. The knowledge base contains the relevant knowledge extracted from human experts necessary for understanding or solving problems. It includes facts such as the problem situation and special heuristics or rules that direct the use of knowledge to solve specific problems in a particular domain. The rules express the informal judgmental knowledge of the application area, providing knowledge not mere facts is the essence of ES. The inference engine is the control structure or the rule interpreter of the ES. This component provides directions of how to use and apply the knowledge stored in the ES. It organizes what actions can be taken from the facts and rules and how to apply them. The user interface is essentially an interface for communication between the computer and the user of the system. Input to the system is often provided by a user or a data base and the inference engine evaluates the input in order to see if any rules are fulfilled or fired, thereby resulting in a response from the system.

In this research an ES is created and filled with knowledge from the domain of technical stock analysis, creating a simple system for investment decision making. The reason for using technical stock analysis is primarily because of the relative ease of expressing these rules in a computationally feasible manner, and secondary because of data availability. The rule base consists of definitions of three patterns derived from the domain of TA, these patterns are used to recommend whether to buy or sell, thus occurrences of a pattern type results in a buy or sell recommendation depending on the specific type of pattern. The inference engine searches a data base in order to evaluate whether any of the rules, constituting an occurrence of a specific pattern, are fulfilled, if so a buy or sell is recommended. Thus in this case all found observations of patterns are regarded as "good".

The expert system is then used in conjunction with artificial neural networks creating a hybrid system, where the ANN is trained to recognize "good" investment opportunities based on the rules of the expert system. Thus the expert system is responsible for pre selecting the instance of patterns or objects, which the neural network is trained to classify as either "good" or "bad" according to their outcome.

In the latter case the ES is also responsible for providing additional input data in the form of specific indicators used in technical stock analysis from the database to the ANN.

The actual performance of the sole ES, equivalent to the performance of the selected investment method is then compared to the performance of the hybrid system, regarding predictive accuracy and profitability, to evaluate if the hybrid system is able to provide an enhancement over the sole use of the expert system.

### **2.2.1. The Knowledge Domain: Technical Stock Analysis**

In the course of years of stock market study, two distinct schools of thought have arisen, regarding the method of arriving at the answer of the investor's problem of what and when to invest. One of these is commonly known as the fundamental or statistical approach and the other as the technical.

According to Edwards and Magee (1998) the stock market fundamentalist depends on statistics. The analysis of auditor's reports, profit-and-loss statements, quarterly balance sheets, dividends records, sales data, etc, is used to derive an estimate of future business conditions. Taking all this into account the stock is evaluated; if it is currently selling below the appraisal, it is regarded as a good buy.

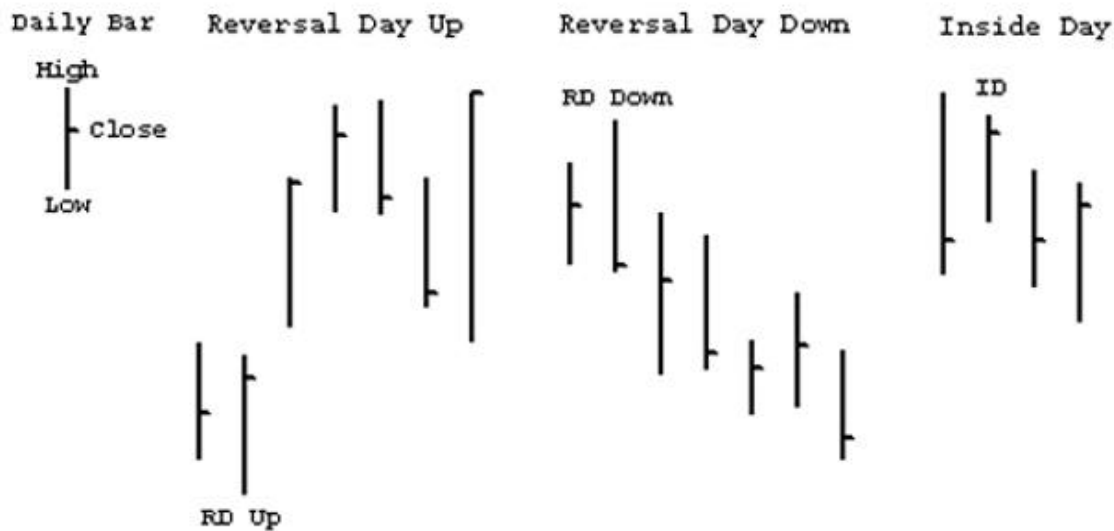
The term technical, in its application to the stock market, has come to have very special meaning, quite different from its dictionary definition. Edwards and Magee (1998) defines technical analysis as: The study of the action of the market itself. As opposed to the study of the goods in which the market deals. Technical analysis is the science of recording, usually in graphical form, the actual history of trading (price changes, volume of transactions, etc) in a certain stock or index, and then deducing from that picture the probable future trend. Figure 7 show an example of a graphical trade chart.

Common techniques of arriving at buying and selling decisions using technical stock analysis are through the use of patterns in stock price and indicators, the patterns can often be expressed using IF, THEN statements and the indicators are usually mathematical formulas derived from price and/or volume, plotted as lines or histograms in a chart.

In this research some specific trading patterns and indicators from best selling technical stock analysis books have been selected to form the knowledge base of an ES.

#### **2.2.1.1. The Patterns**

The patterns used by the expert system are called: Inside day, Reversal day up and Reversal day down. The patterns are explained more closely in their respective sections below. A trading day is usually denoted as a bar in most technical analysis charting programs, where the high, low and closing price are denoted as seen in figure 6. A pattern is denoted by a specific way in which the bars have formed the high, low and closing values. Every bar symbolizes a trading day.



**Figure 6: Illustration of patterns in bar chart form.**

Apart from the patterns we also define a goal for what the outcome should be, following the occurrence of one of these patterns, for us to regard a particular observed pattern as either good or bad.

For example, when a reversal day up (RD up) pattern is found, we want a price increase from the high of the RD up to the highest high of any of the following five days of minimum 3 percent AND that the lowest price value of any of the five days following the RD up is not below the low of the RD up.

A simple evaluation of whether the selected patterns have any inherent power to generate the desired outcome compared to the probability of any given day resulting in the same desired outcome. This means that if the probability of a pattern resulting in the desired outcome is lower than the probability of any day resulting in the same outcome the pattern probably do not have an inherent power to generate the desired goal, and we could instead train a neural network to classify whether any given day results in the desired outcome. Without using an ES to pre select the data on which the ANN is trained.

A drawback of the former approach would be the enormous time needed for training on a desktop PC, since the data set for every day between 1987 10 08 and 2003 03 18 holds 3869 trading days compared to the largest set of the three selected patterns which is only 407 observations and because the training time required have a tendency to increase considerably with the number of training exemplars.

A comparison between the patterns probability of resulting in the defined goal compared to any given day between 1987 10 08 and 2003 03 18 resulting in the same goal is conducted by: comparing the conditional probability of a good outcome given a pattern and the probability of any day resulting in a good outcome. The results of the calculations are found table format under the respective patterns section below.

An evaluation of the profitability of the respective pattern was conducted by creating a decision tree and calculating the expected payoff of the respective patterns. All of the patterns showed to be unprofitable over the long run by this analysis technique, this does not matter



since we are interested in whether we can improve results by letting the neural network select which patterns to trade.

### 2.2.1.2. Reversal Day Up

The reversal day up (RD up) pattern (Torssell & Nilsson, 1998), consists of the current day making a lower low and a higher close, than the preceding bar (to the left), see figure 6 above. The theory behind this pattern according to Torssell and Nilsson (1998) is that there often is a psychological behavior behind the pattern, first price turns down below yesterdays low but then something happens that drives the price higher making the close higher than yesterdays close.

Torssell and Nilsson (1998) does not state an explicit target for what constitutes a good RD up only that the stop loss is to be placed at the low of RD up bar. In order to train the neural network with supervised training a definition of a target for classifying the pattern into good or bad observed instances is needed.

The definition of a good RD up used in this research is that the highest high of any of the five bars following the RD up must be at least 3 percent or more than the high of the RD up bar, and no low of the five bars following the RD up bar may be lower than the low of the RD up bar. This definition constitutes that at least a relatively large movement (3%) will follow the signal within 5 days, the stop loss that is where the trade is terminated (sold in this case) is placed at the low of the RD up bar.

The pattern is traded in the following way: if a RD up has been formed and labeled as “good”, a buy order is placed at the high of the RD up bar. From the definition of our target we expect a 3 percent movement from this level within 5 days, if our order is filled within the five days a sell order is placed at the low of the RD up. If price breaks below the low of the RD up at any time all current positions (if any) are sold, the pattern is regarded as a false buy, and further disregarded. Any positions are sold if the target of a 3 percent rise is achieved. If the pattern does not fulfill its goal within 5 days, but also not violate the stop loss level, it is sold at the end of the fifth day.

One problem arises if prices open above the high of the RD up bar, then one would have to decide whether to take the trade or not. One could also simply buy at the (or close to) the close of the bar when the RD up is formed at the end of the trading day, this would probably be feasible in most cases since the neural networks performs the classification in less than a second.

Table 1 shows the probabilities of the defined goal (definition of good) occurring in the total data set from 1987 10 08 to 2003 03 18 without the pattern and with the pattern, respectively.

Data	Goal Ok	Total n	Probability
Total data set	578	3869	0,149
Only RD Up	81	329	0,246

**Table 1: The Reversal Day Up Pattern.**

From this we can derive that the probability of the RD up pattern resulting in a 3 percent rise within five days without violating the stop loss level is higher than the same occurring any given day in the total data set.

### 2.2.1.3. Reversal Day Down

This pattern is the direct opposite of the reversal day up pattern. Meaning that, today the security makes a higher high than the preceding bar, but closes down below yesterday's close. The theory behind this pattern is similar as for the previous; the only difference is that a new high and lower close than the previous day are formed instead of the opposite.

The definition of a good reversal day down (RD down) is the opposite of the reversal day up pattern, the lowest low of any of the five bars following the RD down must be at least 3 percent lower or more than the low of the RD down bar, and no high of the five bars following the RD down bar may be higher than the high of the RD down bar. This definition constitutes that at least a relatively large movement (3%) will follow the signal within 5 days, the stop loss that is where the trade is terminated (sold) is placed at the high of the RD down bar. Figure 6 clarifies this.

This pattern is traded in a directly opposite way, to the reversal day up pattern.

Table 2 shows the probabilities of the defined goal (definition of good) occurring in the total data set from 1987 10 08 to 2003 03 18 without the pattern and with the pattern, respectively.

Data	Goal Ok	Total N	Probability
Total data set	538	3869	0,139
Only RD Down	80	407	0,197

**Table 2: The Reversal Day Down Pattern.**

In the case of the reversal day down pattern we come to the conclusion that the probability of an observed pattern resulting in our target is higher than any given day in the data set resulting in the same goal.

### 2.2.1.4. Inside Day

The inside day (ID) pattern occurs when the current bars high and low are "inside" the preceding bars high and low. Thus today the high is lower than yesterday's high and today's low is higher than yesterday's low (figure 6). This is a modification of the NR 7 pattern derived from Farley (2001), the NR 7 pattern is originally based on the current bars range (high - low), if today's bars range is smaller than the smallest range of the seven preceding bars, a NR 7 has occurred.

The theory behind both of these patterns is that when they occur, the probability of a short term volatility increase is eminent. Meaning that price will make a short term movement up or down. The direction of the movement is not known beforehand, so one waits until the following trading day and if price breaks up above the high of the inside day, one buys, and the opposite if price breaks down below the inside days low.

The definition used in this research for what constitutes a good inside day is: for buys, the day after the ID price breaks up over the high of the ID and within 2 days makes a 2 percent movement from the high of the ID end the highest high of any of the following 2 days but does not trade below the low of the ID. The stop loss is placed at the low of the ID. For sells the opposite applies (figure 6).

This pattern is traded in the following way: after an ID is formed and labeled as "good", a buy order is placed at the high of the ID and a sell order at the low of the ID. If any of the orders

are filled the other one remains as a stop loss protection. When the target of a 2 percent price movement is met, the position is terminated. If neither the target nor the stop loss is met within 2 days the position is terminated. If price opens sharply higher or lower than the respective entry levels of the ID the pattern should be disregarded.

Table 3 shows the probabilities of the defined goal (definition of good) occurring in the total data set from 1987 10 08 to 2003 03 18 without the pattern and with the pattern, respectively.

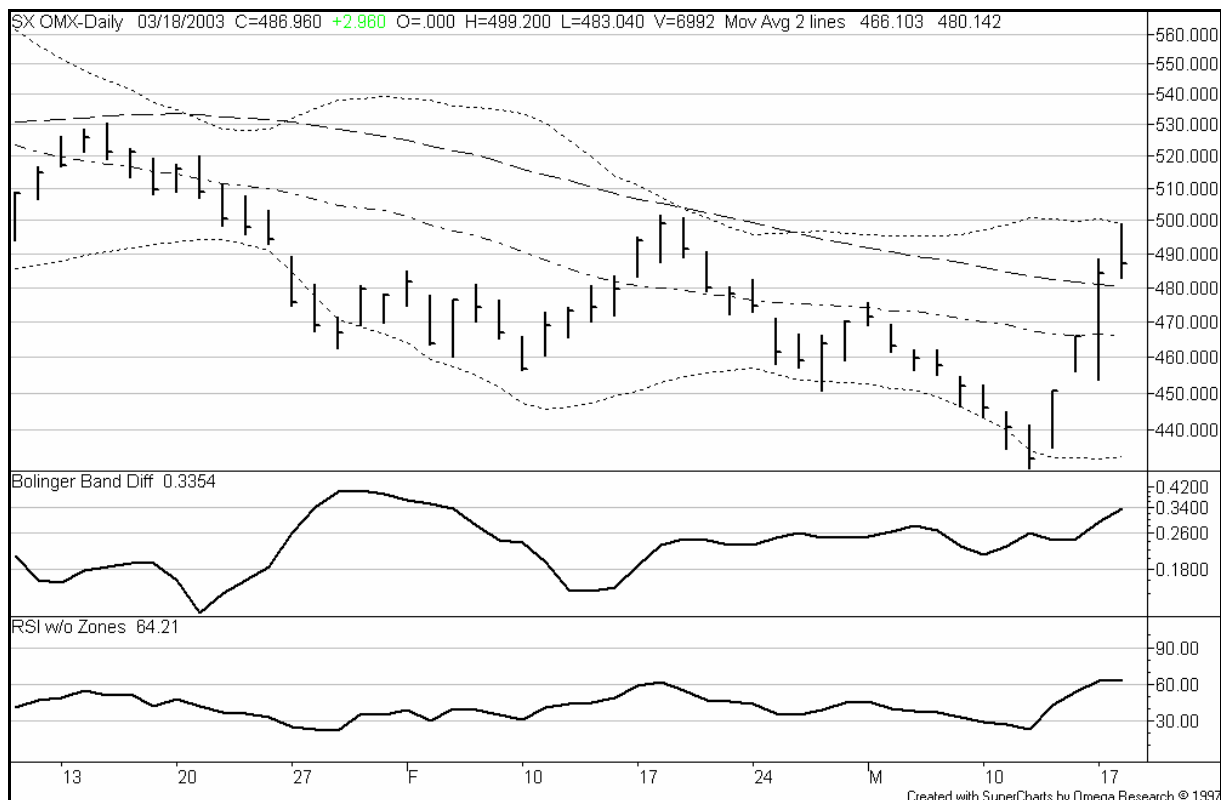
Data	Goal Ok	Total N	Probability
Total data set	1014	3869	0,262
Only Inside days	117	392	0,298

**Table 3. The Inside Day Pattern.**

The inside day has a slightly higher probability of resulting in the desired goal than any day in the total data set.

### 2.2.1.5. Indicators

The inputs to the neural networks used in this research are the same for every pattern to be classified and are indicators commonly used in technical analysis, to provide decision support when determining whether to invest or not. These indicators are plotted in a chart in figure 7, the dotted lines in the top window are the Bollinger Bands, the dashed plot is a 50 moving average of closing price and the dashed/dotted line is a 21 day moving average. The middle window shows the Bollinger band difference indicator and the bottom most window depicts the RSI indicator. To the right of the windows closing values of price, Bollinger band difference and RSI are printed.



**Figure 7: Illustration of a trading chart.**

The indicators selected provide some important information required to analyze market action: Trend, Momentum, and Volatility. The two moving averages gives the medium and long term trend, the Bollinger bands a sense of extreme values, the Bollinger band diff the volatility and the RSI the velocity or momentum of the price.

### **Bollinger bands**

The Bollinger band indicators default settings, is based on a 20 day plot of a moving average (MA) of the closing price. Around this MA +/- 2 standard deviations (stdev) of the 20 day MA are plotted, giving 3 lines. The purpose is to provide relative definitions of high and low, prices near the upper band (+2 stdev of MA) are high and prices near the lower band (-2 stdev of MA) are low (Bollinger, 2002).

### **Bollinger Band Difference**

This is a volatility indicator (Bollinger, 2002) based on the difference of the two Bollinger bands and normalized by the moving average. The indicator is calculated by subtracting the lower Bollinger band from the upper Bollinger band and dividing this value by the value of the moving average. The settings used are a 13 period MA with +/- 5 standard deviations

### **Moving Averages**

A simple 21 and 50 day moving average (MA) is used to determine the intermediate and long trend. The 21 day MA denotes the intermediate trend and the 50 day MA the long term trend (Nilsson & Torssell, 1998). The slope of the MA curve denotes the direction of the trend, making the MA a trend indicator. It is calculated of the closing price each day.

### **Relative Strength Indicator**

The Relative Strength Indicator (RSI) is a front weighted, price velocity indicator (Pring, 1993). The RSI compares the price of a stock or index, relative to itself, and is therefore relative to its past performance. The formula is:  $RSI = 100 - \left[ \frac{100}{1 + RS} \right]$  where  $RS$  equals the

average of the closes of the up days divided by the average of the closes of the down days.

As seen in the equation the value is normalized between 0 and 100. The RSI is used to determine the velocity (momentum) of the price and signaling whether prices are overbought and oversold. Where overbought and oversold means that the price is likely to turn direction, Pring (1993) states that RSI values of 70 signals an overbought price and RSI values of 30 signals oversold conditions.

## **2.3. A theoretical model for integrating Expert Systems and Artificial Neural Networks**

A model outline for combining technical stock analysis and neural networks based on the theory discussed earlier is presented here. The model developed should be applicable for any hybrid systems based on expert systems and neural networks.

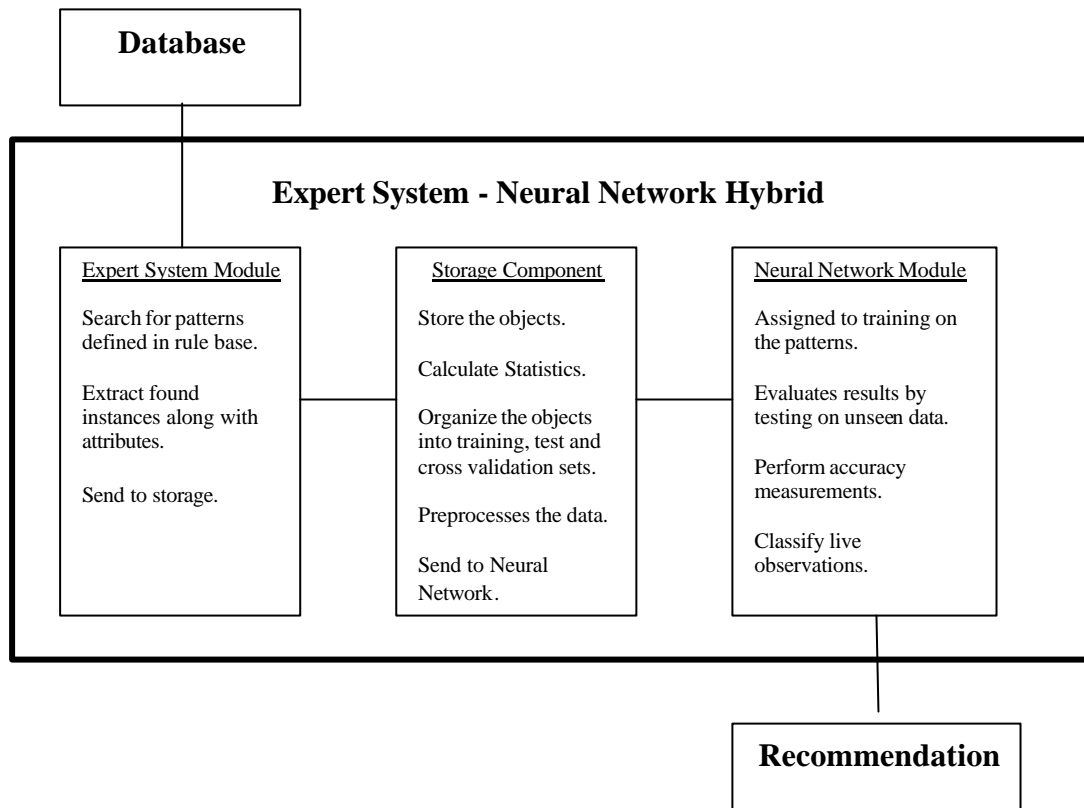
The model consists of three components, an expert system module, a storage component and an artificial neural network module. A database in this case containing historical price and volume information is connected to the expert system module. The expert system module facilitate knowledge stored as rules of which data that should be extracted from the data base, the selected data is extracted and sent to the storage component. In the storage component the data is stored and organized into different files one file for each form of pattern in this case, the data is coded into different classes, randomized and divided into training data, cross validation data and testing data. The processed data file is then provided as input to the neural network module, where a neural network is trained to classify the different observations of patterns into good or bad occurrences.

There are two basic models for integrating expert systems and neural networks, stand-alone and transformational models (Medsker, 1994). The stand-alone model of combined expert system and neural network applications consists of independent software components. The components do not interact in any way. In this case the same task is assigned to both of the systems independently, resulting in both returning a solution. I the solutions differ the user have to select the solution to implement.

A transformational model is similar to stand-alone models in that the end result of development is an independent model that does not interact with another. What distinguishes this model from a stand-alone one is that the transformational system begins as one type (e.g. an expert system) and ends up as the other (e.g. a neural network). Knowledge from the expert system is used to set the initial conditions, the training data and input variables for the neural network, which evolves from there.

In this research a transformational model in combining the expert system and neural network is used as starting point. Where the interaction of the systems in this specific case can be described in the following steps:

1. Based on its rules, the ES extracts instances of objects (patterns) and their associated attributes (input variables), from a database.
2. The data is intermediately stored in a depository (database or file), where pre-processing of the data is done if necessary.
3. The ANN reads the data from the depository (instances of objects and associated attributes) and commences training.
4. The integrated hybrid system produces an output at its designated task.



**Figure 8: A model of an ES & ANN hybrid system.**

## 2.4. Literature Review

This literature review summarizes important research in the area of neural networks applied in finance. Ten articles are briefly described, the methods used as well as achieved performance is highlighted.

### 2.4.1. Applications of Neural Networks in finance

Yoon and Swales (1996) compared the predictive ability of a neural network to that of multiple discriminant analysis (MDA), a statistical analysis method. As input variables 9 fundamental factors were used. The objective was to classify a company as either a well or poor performing firm. Using the given input variables the ANN outperformed MDA at classifying the firms. The ANN achieved a mean of 77.5% correct classifications while MDA got 65%.

In the work by Asakawa, Kimoto, Takeoka and Yoda (1990) a prediction system consisting of several ANN's was constructed to time whether to buy or sell the TOPIX (Index at Tokyo Stock Exchange) one month in advance. The ANN's average prediction results for each month were the base for deciding whether to buy or sell that particular month. The different networks were trained on different input variables. Input variables such as historical price, technical analysis indicators and economic indexes were used. The system was tested on historical data and the actual performance showed a 98 % increase in funds compared to 67% of a buy and hold strategy, from January 1987 to September 1989.

Bergeson and Wunsch II (1996) constructed a commodity trading model based on an ANN and an expert system, where a human expert defined buy and sell patterns using technical indicators such as moving averages, and the ANN was trained using these examples of buy and sell patterns. The human expert choose patterns that he “felt” indicated good buy or sell signals as well as false signals. This method of providing input data showed to be extremely labor intensive, in contrast to simply provide the network with prior values of price and indicator values. The system was trained on data from 1980 to 1988 and tested from January 4, 1989 to January 25, 1991 and showed a profit growth of 660%. Unfortunately the actual patterns used or the rules applied were left out, making it impossible to validate this work.

Kryzianowski, Galler and Wright (1993) used a neural network to automate stock picking, using data from the annual income statement and balance sheets as inputs of 120 publicly traded firms. The objective was to classify a company as positive or negative, regarding the stocks percentage performance of the following 1 year period. The result gave a 66.7 % total accuracy rate (accurate calls/numbers tested) on the positive cases and a 66.4 % total accuracy rate on the negative cases. The author’s way of measuring performance in this case, differs from the performance measurement used in this research. In real numbers, the accurate positive calls were 16 and the inaccurate negative calls were 34, resulting in a total of 50 calls classified as positive, of which only 16 actually were positive. Thus the accuracy rate of the calls classified as positive (the sensitivity) was 32 %. The rate of positive classes/total classes in the test set was 16% (24/149).

A specialized technique called template matching was used by (Leigh, Paz and Purvis, 2002) to create a neural network that used a pattern, common in technical analysis called “bull flag”, to time market entries. The 2 dimensional image of the pattern was transferred to a 10x10 template consisting of 60 trading days. This template was the slid over the time series in order to find areas within the time series graph that fitted into the template. The objective was to select days to buy using the template and neural network, which led to a 5-day price increase. The network showed an ability to correctly classify 680 good buys of 1017 indicated buys (66%) and showed an average 5 day price change of 0.005, compared to 2259 good buys of 3750 possible (60%) by using a “buy and hold for 5 days strategy” the average price change was 0.003 for this strategy.

The research by (Leigh, Purvis and Ragusa, 2002) used the same pattern matching technique and the “bull flag” pattern as the research by Leigh, Paz and Purvis (2002), the difference was that instead of selecting those days that led to a 5 day price increase a 20-day horizon was used. Some fine-tuning of the selection procedure was also conducted. In this case a statistical significant increase in average profit of the system during the testing period was found, compared to the over all 20 day price increase (2.67% compared to 1.08%), 839 days resulted in a purchasing recommendation by the system.

Lin, Ming and Phua (2001) built a genetically evolved neural network to forecast the direction of the STI index at the stock exchange of Singapore. Daily open, high, low and close as well as indices at major stock markets as NASDAQ and Nikkei225 were used as inputs. The genetic algorithms used optimized settings such as architecture and input data window. The network showed an ability to predict the following day’s market direction of 81%. A major flaw according to the authors was the neural networks inability to recognize major turning points, a flaw not uncommon according to the authors when using ANN’s as regression machines.

The NORN (Neural Oscillatory Based Recurrent network) predictor, presented by Lee & Liu (2001) was developed to find long term technical formations in stock price data. Templates of patterns (figure 8) stored as elastic graphs were matched against a two dimensional graph of the price data to find the patterns, the neural network then compared the price graph to the template in order to find eminent patterns. The system showed an overall ability of finding the predefined patterns of 91%, 2517 found patterns of 2766 possible. The 2766 total cases was identified by human experts.

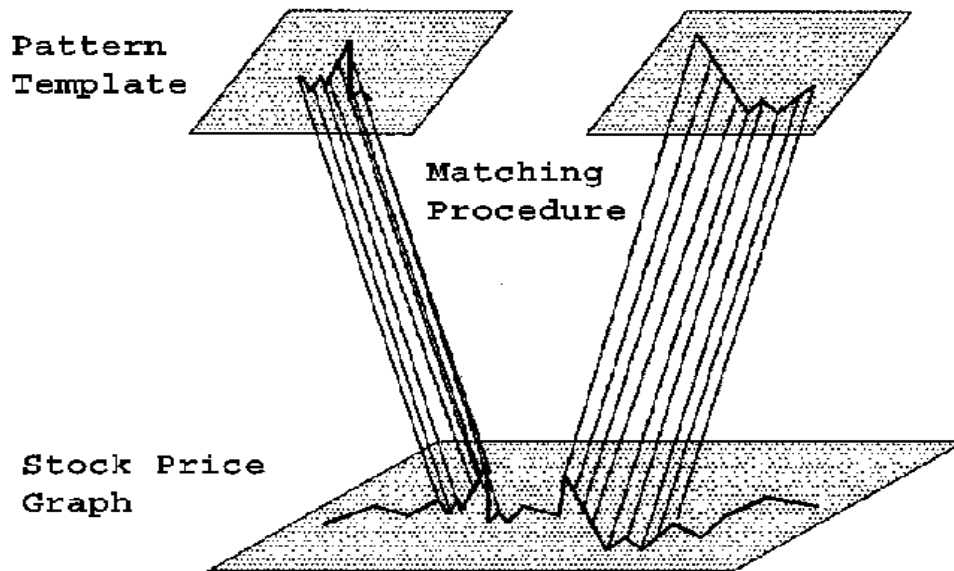


Figure 9: An example of a pattern template matching procedure (Lee & Liu, 2001).

Chenoweth, Obradovic and Stephen-Lee (1996) investigated in their research whether the use of technical indicators applied as a filter to the output of a backpropagation neural network, would improve the models performance. When adding technical indicators the system showed a return of 15.99% on the test set, without, a return of 11.03%, a buy and hold would have resulted in a return of 11,05%. The systems using indicators also resulted in fewer trades, 54 versus 152.

Poh, Tan and Yao (1999) performed a study on the Kuala Lumpur Composite Index, investigating how a neural network using indicators common in technical analysis as input, would perform compared to other investing strategies. The ANN was trained to predict the daily stock index price, and then a heuristic rule was applied to the difference between predictions to indicate a buy or sell. The combination of the neural network as a regression tool and the heuristic buy/sell rule yielded an annual return of 26% during the testing period of 303 days in 1990/1991. A buy and hold strategy would have yielded a loss of 14.98%, comparison was also made to an Autoregressive Integrated Moving Average (Arima) model, a statistical technique, which showed a profit of 19.11 % during the same period.

To summarize we can conclude that past work using ANN as a forecasting tool to predict stock and indices movements has shown promising results from an academic perspective. Neural networks have outperformed standard statistical techniques in prediction and forecasting of economic time series (Yao et al, 1999), and that by combining technical analysis indicators and neural networks an ANN model can be improved (Chenoweth et al, 1996).



Much of the research has focused on using ANN to predict time series values various steps ahead, with relatively promising results. A major drawback of this approach is that it performs badly in recognizing major turning points, resulting in good performance mainly in strongly trending markets. Using another approach earlier research have shown that ANN's have the ability to find patterns in price data when using an ideal template of a pattern to compare with the price data (figure 8). And that neural networks using the "bull-flag" as a pattern template can outperform a buy and hold strategy during a specific period (Leigh, Purvis & Ragusa, 2002).

In this research the occurrence of three specific patterns along with specific input data is extracted from the selected time series of the OMX-index. This is feed to a neural network, in order to train the network on finding certain properties in the data that constitutes a pattern with a predefined outcome. Thus the focus is on finding those particular patterns that yield a short term profit.

## 3. Method

This research use quantitative methods for gathering, processing and analyzing the data. The gathering of empirical data is conducted by running standardized experiments. Processing of the acquired data is done by codifying the outcomes of the experiment into easily understood and separable groups, which are then analyzed by standard statistical techniques and decision trees.

Inference will be drawn from this information by deduction. The research hypothesis which is based on theoretical assumptions regarding the feasibility of technical analysis and neural networks will be tested against the actual outcomes of the experiment, resulting in a hypothetic-deductive work.

### 3.1. Research type

There are according to Alvager and Beach (1992) two main methods in which information can be obtained from an investigated system, in scientific and technological research, the observational method and the experimental method. The observational method involves taking records in a passive way. The researcher's study the phenomena as it is presented, taking notes and trying to formulate laws from the observed facts. In the experimental method, the researchers can create new situations and study the results without relying on conditions given by nature. An experiment can be defined as, the acquisition of data to measure the performance of the solution under controlled conditions in a laboratory.

The research type selected is the experimental method, which is suitable since we want to gather empirical data regarding the selected phenomena under controlled conditions.

First the various standpoints taken when setting up the experiment are described and explained. Next, a description of the setup of the artificial neural network and the various parameters associated with this are explained. And last the procedure of executing the experiment is described.

#### 3.1.1. Experiment Setup

According to Medsker (1994) it is far more difficult to construct a hybrid system than two separate systems. I should also be mentioned that the difficulty in using neural networks for classification tasks is also greater than using them as regression tools, which is the most common method in using neural networks applications for financial prediction.

Most of the time and effort of this research has been devoted to the iterative tasks of selecting input variables, defining the targets for the patterns, selecting the parameters of the neural network, training and testing. It must be stressed that this is an extremely time consuming endeavor, several neural network architectures need to be evaluated each associated with at least a dozen of parameters that need to be fine tuned.

Approximately 30 different input variables were tested (excluding recalculations of these such as lag or difference), several different definitions of targets such as 3, 5, 8 or 10 percent movements within 2, 3, 5, 8, or 10 days were tested, when setting up the parameters of the neural network five different activation functions were tested in different combinations at the two points they are used in the network, five different learning rules for backpropagation training were tested. Three different types of stopping criterions were tested and two types of weight updates were evaluated.

The learning rate parameter and momentum constant, the number of PE's in the hidden layer, which input variables to use and the slope parameter of the activation function, were optimized by genetic algorithms. Genetic algorithms are heuristic search procedures used to find a nearly optimum value instead of simply trying all possible combinations, which would be computationally demanding. Genetic algorithms can achieve an often good enough result in much more computationally and time efficient manner, than trying all possible combinations.

Before conducting the actual experiment a number of factors need to be addressed regarding how to select the datasets for training, how to assign the input samples into the classes, how to improve generalization, which inputs and recalculation of these to use and what to do if the classes are un-proportionally distributed. This is described below.

The method in which the training set is chosen can influence the generalization capability of the network (Cordella, De Stefano, Sansone, Tortorella and Vento, 1998). One way to construct training sets used with successful results in medical image recognition is to hand pick "problematic" cases as training samples (Sun and Nekovei, 1998). Another way is to select the samples in a random way. In this research the test cases are selected randomly, by using the random number generator in MS Excel and the ordering the observations. After randomization the top most 60 % of the observations were selected as training data, the next 20 % as cross validation data and the last 20% as testing data.

The way in which the neural network assigns the input samples to the respective classes is also important. If the cost of misclassification is regarded as high, for example in classifying medical images in the framework of cancer patients, a high reject rate is acceptable provided that the misclassification rate is kept low. The input sample in this research is attributed to a class according to the winner-takes-all rule, meaning that the input sample is attributed to class 0 or 1 depending on whose output neuron has the highest or lowest value, resulting in every case being classified as either 0 or 1, for easy statistical comparisons of the results.

A commonly encountered problem in MLP classification occur when the frequency of classes in the training set vary significantly (Euliano et al, 2000). If the numbers of training examples for every class vary significantly the neural network may "follow the line of least resistance" and always predict the most common class. In this research the classes do vary significantly and to avoid that the neural network follows the line of least resistance, each class will be weighted proportionately according to the number of samples of that class present in the training dataset.

It is crucial to avoid overtraining when training neural networks, when the network is trained too much, the network 'memorizes' the training patterns and does not generalize well. This has been avoided by using a cross validation set. Cross validation computes the error in a test set at the same time that the network is being trained with the training set. The error can keep decreasing in the training set, but may start to increase in the test set. This happens when the network starts "memorizing" the training patterns, when this happens the training is stopped.

The selection of inputs that have a predictive relationship to the target is extremely important, if the inputs don't have any or very a weak relationship to the target being classified the results will be poor. According to Gately (1996) and Azoff (1994), the use of lagging values and differences of current and some periods back of the indicators, can help the network

during training. Therefore lags and differences of the indicators are provided as well as the current values.

The inputs used are the same for every pattern to be classified, they are:

<b>Current Value</b>	<b>Five periods back value</b>	<b>Current/Five per. back val.</b>
Close	Close	Close
High		
Low		
RSI	RSI	RSI
MA 50 period	MA 50 period	MA 50 period
MA 21 period	MA 21 period	MA 21 period
Bollinger Bands on MA 21	Bollinger Bands on MA 21	Bollinger Bands on MA 21
Bollinger Bands Difference MA 13, +/- 3 StDev	Bollinger Bands Difference MA 13, +/- 3 StDev	Bollinger Bands Difference MA 13, +/- 3 StDev

The selection of target or what is to be classified is a very important factor of the performance of the system. Gately (1996) state that if poor performance is achieved when using neural networks to predict time series, the first thing to revisit should be the target. If a network shows poor performance at predicting an absolute price level, better results may arise if the target is changed to predict i.e. the change in price. In this research the target to be predicted is whether a particular instance of a pattern is good or bad. Hence the definition of a good and bad pattern, what constitutes the target, is crucial. If the neural network shows poor performance at its designated task, the target is one of the factors that should be experimented with.

### **3.1.2. Neural Network Setup**

The objective is to build a neural network with a good ability to generalize that is, good neural network model fitness in the out-of-sample data. There are several parameters that need to be addressed that determine the models performance, such as the choice of activation function, network architecture (number of hidden nodes/layers), learning times (number of epochs) and type of training algorithm and the various parameters associated with the training algorithm.

In this research an object oriented simulation environment called Neuro Solutions 4.2, is used to build the neural networks. A multi layer perceptron network with one hidden layer and back propagation learning using a tanh (hyperbolic tangent) activation function is the base configuration created for every trading signal. The only difference between a tanh activation function and a sigmoid activation function (figure 3) is that the tanh function assumes a value between  $-1$  and  $1$ , not  $0$  and  $1$  as in the case of the sigmoid.

Apart from this base configuration, several parameters need to be set with values such as the momentum constant and step size (formula 3) of the learning algorithm, the slope parameter of the activation function (formula 1), the number of neurons in the hidden layer and if some inputs are to be excluded based on poor predictive relationship. These parameters have been the subject of optimization by the use of genetic algorithms. 2500 different combinations in parameter settings of the base configuration for every pattern, was evaluated using the genetic algorithm, the combination that achieved the lowest MSE was selected as the final neural network model. This process of training 2500 neural networks took between 3 and 8 hours on

an 800Mhz desktop PC with 512 MB Ram, depending on the selection of the genetic search procedure.

### **3.1.3. Experiment Execution**

The data used, is provided from Borsdata AB ([www.borsdata.se](http://www.borsdata.se)) and loaded into SuperCharts, a technical stock analysis software. The various values for the indicators used as input to the neural network are calculated in SuperCharts. All experiments have been conducted on the same security, the Swedish OMX index which is a weighted index of 30 most traded stocks on the Stockholm stock exchange. Daily data from 1987 10 08 to 2003 03 18 was used. No thorough examination of the correctness of the data from Borsdata has been done, only that a value exists for every period.

The data was exported into an Excel spreadsheet from the SuperCharts application and all occurrences of the respective forms of patterns were extracted along with the selected input variables using a programmed macro (this constituted the ES module), giving three data sets, one for each pattern form, the input data was stored in columns and each occurrence of a pattern consisted of the entire row. In the respective data sets the patterns were coded into good and bad occurrences, based on the respective goal. Were 1 denoted a good occurrence and 0 a bad. Thus each data set consisted of a number of observed patterns, along with the selected input, and a classification of the patterns outcome (1 or 0). Each row which denoted an occurrence of a specific pattern was assigned a random float number ranging from 0 to 1, and the rows were then sorted, thereby randomizing the entire data set (this is the storage component).

From the respective randomized sets a portion was assigned as training data (60%), cross validation (20%) and (20%) as testing data. This data was feed to an artificial neural network (the ANN module) created with Neuro Solutions. After completed training the networks performance was tested on the test set.

## **3.2. Data collection and Data analysis**

Thurén (1991) states that there are two main ways of gathering and making sense of data used to derive knowledge in research, qualitative and quantitative methods.

Qualitative methods have a low degree of formalization. These methods are primarily used as a mean to provide understanding of a specific phenomenon. The purpose is not necessarily to test whether the information holds in general. The central theme is to get a deeper comprehension of the problem context of study.

Quantitative methods are more formalized and structured. These methods are to a larger extent characterized by control from the researcher. Design and planning using these methods are characterized by selectivity and distance to the information source. This is necessary to conduct a formalized analysis and comparison, and to test whether the result achieved hold in a more general sense. Statistical measurement methods play a central role during the analysis of quantitative information.

This research is based on quantitative methods for data gathering and data analysis, thus quantitative data is provided as input to the experiments. The empirical results from the experiments are analyzed in a quantitative manner; standardized statistical tests are conducted on this data, to provide an answer to the research question. The empirical data gathered from

the experiments are transformed into specialized quantitative performance metrics that enable a more valid analysis of the outcomes. These performance metrics also enable statistical measurements to prove the research question. The performance metrics used are explained and defined in the following section, thereafter the statistical tests are explained and defined.

### 3.2.1. Performance metrics

When the interclass prior probabilities of classes vary significantly, as is the case for all the pattern forms in this research, the overall classification error may not be most appropriate performance criterion.

For example, suppose you have a two-class problem where 99% of your samples are of the Class1 type and 1% is of the Class2 type. The network will likely follow the path of least resistance and arrive at a model that classifies all of your data as Class1. Overall, the network will be 99% accurate, but this is probably not what you want since it will be 0% correct for Class2. Or a model may show a probability of correctly classifying class1 at 80% and 90% respectively for class2. But if the prior probabilities, the number of class1 relative to the total observations, vary significantly i.e. 20 instances of class1 and 200 instances of class2, the model correctly classifies 16 ( $0.8 \cdot 20$ ) instances of class1 and incorrectly classifies 20 ( $200 - (0.9 \cdot 200)$ ) instances of class2 as class1, resulting in a  $(16/36)$  44 % probability of an observation classified as class1, actually belonging to class1.

According to Back et al (1998) statistics such as the sensitivity, positive predictivity and false positive rate, can provide more meaningful results regarding the performance when using neural networks for classification.

The sensitivity of a class is defined as the proportion of events labeled as that class which are correctly detected. In the confusion matrix below rows correspond to the predicted classes and columns to the desired (real) classes. Making the sensitivity of class1 equal to  $A / A + B$ , and  $D / C + D$ , for class 2.

**Confusion Matrix**

Output Desired /	<i>Class1</i>	<i>Class2</i>
<i>Class1</i>	A	B
<i>Class2</i>	C	D

**Table 4: Explanation of the confusion matrix.**

The positive predictivity of a class is the proportion of events which were predicted to be the class and were labeled as that class. Making it  $A / A + C$ .

The false positive rate of a class is the proportion of all patterns for other classes which were incorrectly classified as that class,  $C / A + C$ .

Some research have only focused on measuring the neural networks ability to correctly classify the respective classes not mentioning their distribution relative to each other, or the rate of misclassification meaning the number of wrong classifications relative to the correct ones, making it difficult to draw any clear conclusions regarding the practical performance.

Apart from the above performance measurements the final networks mean square error and correlation coefficient are calculated. The calculation of the mean square error (MSE) which

is the networks error criterion has been described in formula 2. The size of the mean square error can be used to determine how well the network output fits the desired output, but it does not necessarily reflect whether the two sets of data (networks output and desired output) move in the same direction. For instance, by simply scaling the network output, we can change the MSE without changing the directionality of the data. The correlation coefficient (r) solves this problem. By definition, the correlation coefficient between a network output x and a desired output d is:

$$r = \frac{\sum_i (x_i - \bar{x})(d_i - \bar{d})}{N \sqrt{\frac{\sum_i (d_i - \bar{d})^2}{N} \frac{\sum_i (x_i - \bar{x})^2}{N}}}$$

The correlation coefficient is confined to the range [-1, 1]. When r =1 there is a perfect positive linear correlation between x and d, that is, they co vary, which means that they vary by the same amount. When r = - 1, there is a perfectly linear negative correlation between x and d, that is, they vary in opposite ways (when x increases, d decreases by the same amount). When r = 0 there is no correlation between x and d, i.e. the variables are called uncorrelated. Intermediate values describe partial correlation, for example a correlation coefficient of 0.88 means that the fit of the model to the data is reasonably good.

Finally, a decision tree is constructed to evaluate the profitability of the patterns. A decision tree consists of a set of nodes and branches. At a decision node, the decision-maker takes an action; the action is the choice of a branch to be followed. The branch leads to a chance node, where chance determines the outcome; that is, chance chooses the branch to be followed. Then either the final outcome is reached or (the branch ends) or the decision-maker gets to take another action, and so on. A decision node is marked by a square and a chance node by a circle. The actions of chance are governed by a probability, the probability in this case is obtained by calculating the sensitivity of the good outcomes of the patterns achieved by the sole pattern heuristic and the hybrid system respectively. The final outcomes are the reward or loss achieved by the selected course of action; that is, the reward would be the profit of a good trade and the loss the loss of a bad trade. The value of the final outcome is calculated according to the worst case average principle. This means that if a pattern has not meet its designated goal within the specific time, we assume that the position is terminated at a loss of the average percentage difference between the entry level and the stop loss level.

Figure 10 exemplifies this, two decision trees are shown, the topmost denotes the probabilities associated with the recommendation obtained by the hybrid system for the inside day pattern calculated on the test set, the one at the bottom shows the same for the sole use of the pattern. Thus the expected payoff of taking a trade when the hybrid system recommends it, is in the case of the ID pattern a profit of 4.5 points. The sole use of the pattern shows an expected payoff of -0.9 points, making it unprofitable.

The average maximum payoff is calculated by first setting the OMX value to 500, the profit of a good outcome is according to the goal 2%, making the average profit 10 points (500\*0.02). The average maximum loss is obtained by, for the bad patterns calculating the percentage difference between the entry (High or Low) and the stop loss point (opposite of the entry), which in the ID pattern case is 0.0103 when calculating High-Low/High and 0.0109

when calculating High-Low/Low. In the case when an ID pattern fails the average loss is 1.03% for buys and 1.09% for sells. For simplicity we use the value 1.1% (worst case average loss) as the average loss for a failed ID pattern, which is 5.5 points ( $500 \times 0.011$ ). The average maximum payoff is calculated in the same manner for the other patterns.

This of course also assume that that the price does not gap open over the entry in the case of a buy or gap below in a sell, the day after the occurrence of a pattern. An important fact to note is that in this case we assume that when the patterns goal is not met, we take a loss calculated as the difference between the entry and the stop loss level. This constitutes the maximum average loss.

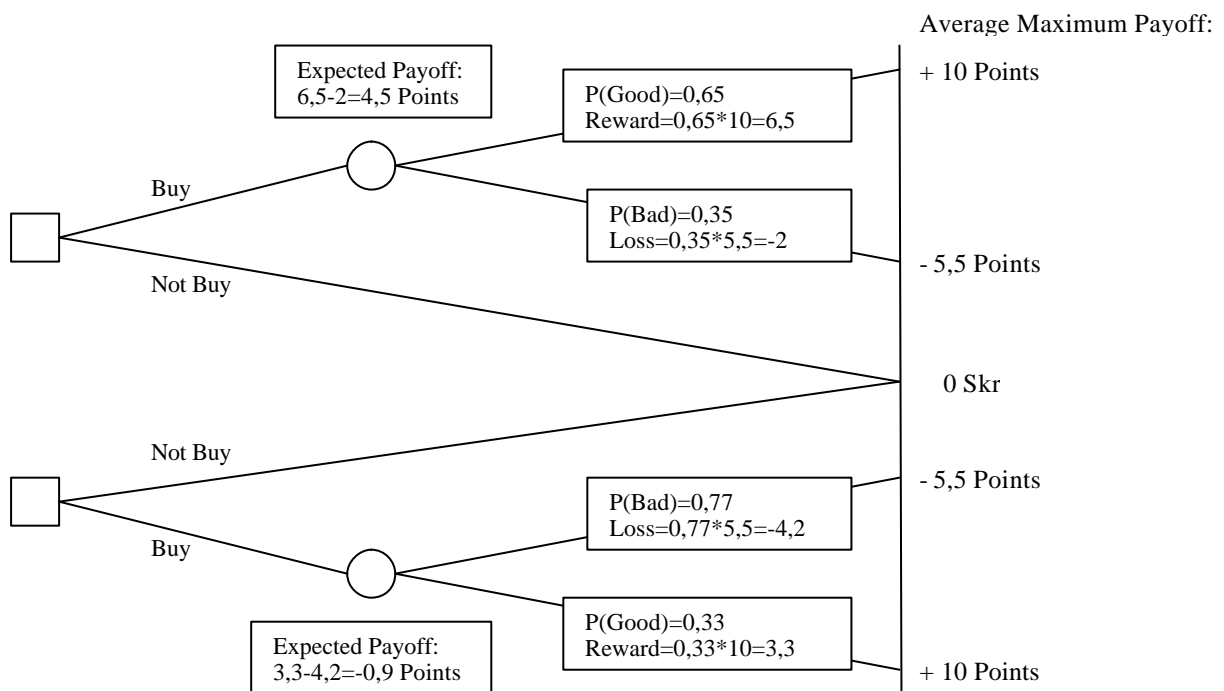


Figure 10: A decision tree for the inside day pattern.

### 3.2.2. Statistical tests

In order to make a test of whether the ability to predict good occurrences of an observed pattern form is higher by the use of the hybrid system than the sole use of the pattern heuristic, a one tailed test for the difference between two population proportions (Aczel, 1996), sensitivity, is conducted. This test assumes that the observations are normally distributed. According to Aczel (1996) the sampling distribution of the sampling proportion is considered normally distributed when both  $n \cdot p$  and  $n \cdot (1-p)$  are larger than 5. If this is not the case the t distribution with  $n-1$  degrees of freedoms will be used. The selected significance level is 0.05.

Thus the sensitivity achieved in the test set by the sole use of the pattern heuristic is compared to the sensitivity achieved by the neural network in its respective set. The sensitivity of the test set, is calculated by first determining the number of observed patterns that are to be “acted” upon (classified as buys), and then calculating the percentage of good occurrences of



the total. In the case of the sole use of the pattern, this means that all observations of a pattern are regarded as a buy.

The null hypothesis and the alternate hypothesis are formulated as follows:

$H_0$ : The difference in sensitivity between the two populations is zero.

$H_1$ : The sensitivity achieved in population 1 is larger than the one achieved in population 2.

Or in mathematical form:

$H_0: P_1 - P_2 = 0$

$H_1: P_1 - P_2 > 0$

Where  $P_1$  is the sensitivity of the good outcomes achieved in the population by to the hybrid system, and  $P_2$  is the sensitivity of the good outcomes achieved in the population of by the sole pattern heuristic.

The equation for this test is:

$$z = \frac{p_1 - p_2}{\sqrt{p(1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

**Formula 4: The formula for the statistic comparison test.**

Where  $p$  stands for the combined sample proportion in both samples, considered as a single sample. That is:

$$p = \frac{x_1 + x_2}{n_1 + n_2}$$

This test is designed to answer the question whether the sensitivity regarding the good outcomes achieved by the neural network is higher than the one achieved by the sole pattern heuristic, and if it is statistically significant at the confidence level of  $\sigma = 0, 05$ . This test does not state how a large improvement is achieved.

For those patterns where an improvement is found statistically significant, a confidence interval for the difference between the two population proportions is calculated. The purpose of this test is to give an approximation of how a large improvement in sensitivity that the hybrid model can yield over the sole use of the pattern heuristic. A confidence level of 90% will be used, so  $z$  assumes the value 1.64.

$$(p_1 - p_2) \pm Z \sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}$$

**Formula 5. The formula for the confidence interval calculation.**

### 3.3. Inference

When conducting inference it is usually separated between induction and deduction. Alvesson and Sköldbberg (1994) define the different approaches in the following way: an inductive approach uses a number of separate cases to formulate a general truth from any observed relationships. While deduction starts with a general rule or theory and state that this rule only explains a certain case of interest.

Induction can result in that underlying structures or situations are left out of the picture. Since one from a collection of separate instances formulate a truth. The deductive approach can in turn be blamed for not explaining anything, rather avoiding explanation. Deduction is based on theory while induction is based on empiric findings. In reality there is a presence of a mix between two approaches, called adduction. Adduction is based on empirical facts without rejecting theoretical assumptions.

The classical way in which artificial neural networks base its inference is purely inductive. No theoretical assumptions regarding the relationship of the input variables to the desired output are needed. The ANN solely draws its conclusion by examination of the data set provided as training.

An expert system on the other hand is based on some sort of knowledge or theory regarding the specific domain and is therefore deductive in its inference.

These two methods are thus combined to some extent in the model of the hybrid system, the expert system that pre selects the desired patterns performs inference by pre established theory regarding which data to select for training and which input to provide to the ANN. The ANN on the other hand does not use any assumptions regarding the properties of the data it is served, and performs inference by induction. We can therefore state that the way in which the hybrid system draws conclusion is or performs inference is adductive, meaning that a combination of deductive and inductive inference is used in the hybrid systems inference.

But the research question stated in this research is based on prior theory from the domain of technical stock analysis, and the feasibility of artificial neural networks as a classification tool. Thus based on prior theory a hypothesis is stated, which is to be tested. This hypothesis is based on the following assumptions:

1. Patterns derived from technical stock analysis can be used to some extent to predict future price behavior.
2. Artificial neural networks can predict the outcome of these patterns.

So the way in which we conclude whether the research question is true or false is by deduction.

To summarize we conclude that we have used theoretical assumptions to formulate a hypothesis, which is to be accepted or rejected based on empirical findings from a series of experiments. The way in which inference is drawn during the experiments is adductive, the way in which inference is drawn to answer to research question is deductive.

### **3.4. Reliability and Validity**

When conducting quantitative research two important things is the reliability and validity of the work (Thurén, 1991). According to Andersen (1994) the importance of an exact formulated research problem and that the definitions used within are free from ambiguity, precise and adequate, can not be over state, since the evaluation of reliability and validity relies upon this.

Reliability means that the measurements are correctly done. If calculated correctly and the element of randomness is eliminated several researchers using the same method should come up with similar results. The principle is that the same method is supposed to yield the same results irrespective of the person conducting the research. Validity according to Thurén (1991) means that the research really has investigated what it was supposed to and nothing else. To clarify this, any conclusions regarding the outcome must be based on well-founded assumptions regarding the empirical data's relationship to the problem or hypothesis. It is often a question of judgment and the goal of the research whether the validity of the work is high or not.

The research question stated in this research and the definitions used within have been carefully formulated regarding precision and adequacy, without becoming stiff or difficult to comprehend. At least this was the author's intent.

The methods used in this research and the calculations associated with the analysis are explained to the extent that the results should be replicable by other researchers. In order to provide a high degree of validity of this research the definition of what is to be investigated and the description of the methods to do this are thoroughly explained.

Many researchers have stated that designing neural networks is more of an art than a science, and that many pitfalls are associated with this endeavor. It is after many weeks of careful study and time consuming experimentation, that the author of this research expresses his personal belief that this statement is true. Since the in lack of expressed best practices regarding the design of a neural classifier for this type of problem, the methods and assumptions used could probably have been improved by an individual with deeper domain specific skills. The conclusions regarding the outcome of this research are drawn in the final fifth and final section.

## 4. Analysis of Results

The following performance metrics: sensitivity, positive predictivity, false positive rate, MSE and correlation achieved by the best performing neural networks for the respective pattern forms and class are depicted in table format along with a confusion matrix of the class distributions. The statistical test (formula 4) aimed at accepting or rejecting the null hypothesis stating that no difference in sensitivity between the hybrid system and the sole pattern heuristic is then calculated. If the outcome of this test finds a statistically significant improvement in sensitivity by the hybrid system, a confidence interval (formula 5) to approximate the extent of this improvement is calculated. Last the decision tree technique (figure 10) is used to evaluate the profitability of the hybrid system and the sole pattern respectively, on the observations in the test set.

It is important to note that in the case of the sole use of the pattern heuristic used by the expert system, all found observations of patterns are regarded as good.

### 4.1. The reversal day up pattern

After completed training the results achieved in the test set became:

<b>Performance</b>	<b>RD Up Good</b>	<b>RD Up Bad</b>
Sensitivity	0,4000	0,8913
Pos Predictivity	0,6154	0,7736
False Pos Rate	0,3846	0,2264
MSE	0,2432	0,2469
Correlation	0,3024	0,3024

The confusion matrix below shows the classifications made by the hybrid system, the top row shows the classifications labeled as good by the hybrid system. Of these 20 classifications labeled as good by the hybrid system, 8 were correctly classified.

<b>Output / Desired</b>	<b>RD Up Good</b>	<b>RD Up Bad</b>
<b>RD Up Good</b>	8	12
<b>RD Up Bad</b>	5	41

The sensitivity of the “good” reversal days is 40% in test set, compared to the a priori probability of a reversal day turning out as a good signal of 19.7% (13/66) in the test set. This means that when the neural network classifies an observed RD up as a good RD up, the probability of this classification being correct is 40 %. Thus an improvement of over 100% compared to the prior probability.

This increase of probability (40 % compared to 19.7%) in determining if a particular RD up will result in a price increase has occurred at the cost of missing 5 good buying opportunities or a loss of 38.5 % of the good opportunities.

The statistic test evaluating if the difference in sensitivity in the two populations is significant at a 0.05 significance level, is now calculated. Both  $n * p$  and  $n * (1-p)$  are larger than 5, so the population proportion is considered large, and the z distribution is used. The z critical value of this one sided test is 1.64, at the selected significance level.

The null and alternate hypotheses are expressed as:

$$H_0: 0.40 - 0.197 = 0$$

$$H_1: 0.40 - 0.197 > 0$$

The test is calculated according to formula 4 and is shown here for clarification:

$$z = \frac{0,40 - 0,197}{\sqrt{(0,244 * 0,756) * \left(\frac{1}{20} + \frac{1}{66}\right)}} = 1,85$$

$$p = \frac{8 + 13}{20 + 66} = 0,244$$

The z value obtained (1.85) is larger than the critical value (1.64), so the null hypothesis is rejected and the alternate hypothesis accepted. We can be 95% confident that the sensitivity achieved by the hybrid system regarding the good signals is higher than the sensitivity achieved by the sole use of the pattern.

The improvement is statistically significant at the 0.05 significance level. Thus the null hypothesis is rejected and the alternate hypothesis is accepted, there is a statistically significant increase in sensitivity.

We now calculate a confidence interval at a 90% confidence level for the difference in sensitivity regarding the good RD up pattern of the two population proportions (formula 5). We may be 90% confident that the difference between the both sensitivity proportions lies in this range.

$$(0,40 - 0,197) \pm 1,64 \sqrt{\frac{0,40 * 0,60}{20} + \frac{0,197 * 0,803}{66}} = 0,203 \pm 0,197$$

This means that we may be 90% confident that the difference in sensitivity between the two proportions of interest lies anywhere from **[0,006...0,400]**.

When using the decision tree technique (figure 10) for evaluating profitability. The result shows an expected payoff of **1.5** points  $((0.4 * 15) - (0.6 * 7.5))$  for when the hybrid system states a buy (OMX set to 500). The expected payoff is **-3** points  $((0.197 * 15) - (0.803 * 7.5))$  for the sole use of the pattern heuristic.

The average maximum loss in the cases of a bad pattern is 1.5 % and according to the definition of a good pattern the profit is 3%. No gaps (the following days low is > than the high of the RD up day) occurred in this data set.

## 4.2. The reversal day down pattern

After completed training the results achieved in the test set became:

<i>Performance</i>	<i>RD Down Good</i>	<i>RD Down Bad</i>
Sensitivity	0,3333	0,8406
Pos Predictivity	0,2667	0,8788
False Pos Rate	0,7333	0,1212
MSE	0,1983	0,2050
Correlation	0,1191	0,0736

The confusion matrix below shows the classifications made by the hybrid system, the top row shows the classifications labeled as good by the hybrid system. Of these 12 classifications labeled as good, only 4 were correctly classified.

<i>Output / Desired</i>	<i>RD Down Good</i>	<i>RD Down Bad</i>
<i>RD Down Good</i>	4	8
<i>RD Down Bad</i>	11	58

The sensitivity of the “good” reversal day down signals is approximately 33% in test set, compared to the a priori probability of a reversal day turning out as a good signal of 18.5% (15/81) in the test set. This means that when the neural network classifies an observed RD down as a good RD down, the probability of this classification being correct is approximately 33 %. Thus at first glance a large improvement compared to the prior probability.

This increase of probability (33 % compared to 18.5%) in determining if a particular RD down will result in a price increase has occurred at the cost of missing 11 good selling opportunities or a loss of 73,3 % of the good opportunities.

The statistic test evaluating if the difference in sensitivity in the two populations is significant at a 0.05 significance level, is now calculated. Both  $n * p$  and  $n*(1-p)$  are not larger than 5, so the population proportion is considered small, and the t distribution is used. The t critical value of this one sided test with  $n-1$  (11) degrees of freedom is 1.80, at the 0.05 significance level.

The null and alternate hypotheses are expressed as:

$$H_0: 0.33-0.185 = 0$$

$$H_1: 0.33-0.185 > 0$$

The statistical test for the difference between the sensitivity in the two populations is calculated using formula 4. The variable z assumes the value 1.16 which is lower than the critical value 1.80, so the null hypothesis is accepted. The conclusion is that the improvement in sensitivity achieved in this case is due to random chance at a 95% certainty. Since this experiment did not prove a statistically significant increase in sensitivity according to the confidence level of 0.05, no confidence interval is calculated.

The result using the decision tree technique (figure 10) shows an expected payoff of **-1.6** points  $((0,33*15)-(0,77*8,5))$  for when the hybrid system states a buy (OMX set to 500). The expected payoff is **-4.2** points  $((0.185*15)-(0.815*8.5))$  for the sole use of the pattern

heuristic. But the expected payoff is achieved by the hybrid system is of course in this case not statistically significant.

The average maximum loss (High-Low/Low) in the cases of a bad pattern is 1.7 % and according to the definition of a good pattern the profit is 3%. 3 gaps (the following days high is < than the low of the RD down day) occurred in this data set, all three of them belonging to bad RD down patterns so they do not alter the average profit.

### 4.3. The inside day pattern

After completed training the results achieved in the test set became:

<b>Performance</b>	<b>ID Good</b>	<b>ID Bad</b>
Sensitivity	0,6500	0,7759
Pos Predictivity	0,5000	0,8654
False Pos Rate	0,5000	0,1346
MSE	0,1774	0,1797
Correlation	0,4913	0,4913

The confusion matrix below shows the classifications made by the hybrid system, the top row shows the classifications labeled as good by the hybrid system. Of these 20 classifications labeled as good, 13 were correctly classified.

<b>Output Desired</b>	<b>ID Good</b>	<b>ID Bad</b>
<b>ID Good</b>	13	7
<b>ID Bad</b>	13	45

The sensitivity of the “good” inside days is 65% in test set, compared to the a priori probability of any inside day turning out as a good signal of 33.3% (26/78) in the test set. This means that when the neural network classifies an observed ID as a good ID, the probability of this classification being correct is 40 %. Thus an improvement of approximately 98% compared to the prior probability.

This increase of probability (65 % compared to 33.3%) in determining if a particular ID will result in a good one, has occurred at the cost of missing 13 good buying opportunities or a loss of 50 % of the good opportunities.

The statistic test evaluating if the difference in sensitivity in the two populations is significant at a 0.05 significance level, is now calculated. Both  $n * p$  and  $n*(1-p)$  are larger than 5, so the population proportion is considered large, and the z distribution is used. The z critical value of this one sided test is 1.64, at the selected significance level.

The null and alternate hypotheses are expressed as:

$$H_0: 0.65-0.33 = 0$$

$$H_1: 0.65-0.33 > 0$$

The statistical test for the difference between the sensitivity in the two populations is calculated using formula 4. The variable z assumes the value 2.64 which is higher than the critical value 1.64, so the null hypothesis is rejected. Since this experiment proves a

statistically significant increase in sensitivity according to the confidence level of 0.05, a confidence interval is calculated.

We now calculate a confidence interval at a 90% confidence level for the difference in sensitivity regarding the good ID pattern of the two population proportions. Formula 5 is used, which results in  $0.32 \pm 0.198$

This means that we may be 90% confident that the difference in sensitivity between the two proportions of interest lies anywhere from **[0.122, 0.518]**.

The result from the decision tree in figure 10 shows an expected payoff of **4.5** points  $((0.65*10)-(0.33*5.5))$  for when the hybrid system states a buy (OMX set to 500). The expected payoff is **-0.9** points  $((0.33*10)-(0.77*5.5))$  for the sole use of the pattern.



## 5. Conclusion and Discussion

This final chapter concludes this research by a brief conclusion of the results achieved by the hybrid approach, followed by a more general discussion. Last, further research in this application domain is recommended.

### 5.1. Conclusion

Concluding this research is best done by answering the primary research question:

- Can an artificial neural network provide an enhancement to specific technical rule based trading patterns?

The major findings of this research are that; using the proposed method we were able to enhance some specific technical rule based trading patterns, occurring in the OMX-index during the specific period. There is support in the results to answer yes to the research question, since two of the specific patterns were significantly enhanced.

The results are promising, this research support the theory that there are factors present in the specific input variables influencing the phenomena of a good occurrence of a specific pattern. And that neural network can identify this phenomenon, in these two specific cases. But it is important to stress the fact that there is no evidence, that the proposed method would enhance other already profitable patterns in other stocks or indices.

### 5.2. Discussion

The results achieved by the hybrid system are in two cases better than in the case of the sole expert system. It is important to note that the neural networks also use additional input data to separate between good and bad patterns compared to the sole pattern heuristics. It could be argued that a human expert with domain specific knowledge in technical analysis could probably also achieve an improvement over the sole pattern heuristic with the use of these indicators as to filters. But it is interesting that a significant improvement can be achieved with neural networks *without* deep domain specific knowledge of technical analysis. The patterns and indicators used in this research are regarded as common knowledge since this information is available for the broad masses, which may contribute to the fact that all the patterns showed very poor results when applied in isolation. In two of these cases the hybrid system managed to turn the prior unprofitable pattern heuristics into a profitable ones, the results achieved by the hybrid system in classifying the inside day pattern and the reversal day pattern are indeed promising.

In this research the expert system was used to make a preliminary assessment, based on the assumption of the feasibility of the specific patterns, the neural networks was then used to refine this assessment further using its generalization and learning properties to optimize the strategy. In some cases the hybrid model showed the potential to achieve a higher prediction performance than the single ES model. Often the capabilities of a single technique are not enough to deal with the complexities of real world problems. When two or more techniques are integrated, their different strengths can be combined into what is termed a hybrid system. As in this case, an example of a hybrid system, where an expert system combined with a neural network performs a pattern classification with better efficiency resulting in improved decision support compared to the sole use of the expert system.

In a practical implementation of a hybrid system, utilizing techniques such as an expert system and artificial neural networks, for investment decision support the both methods could complement each other. The expert system is responsible for screening or searching for the most favorable investment opportunities, according to some predefined rules or knowledge. The neural networks are then trained to further refine the expert systems findings. The system can easily be integrated with a database containing global financial information and perform tasks such as automated arbitrage and stock picking.

Although not the within the scope of this research an interesting type of neural networks are so called *unsupervised neural networks*, such networks can be used to visualize and organize large data sets. One type of unsupervised network is the Self Organizing Map (SOM) these networks can be used to detect new knowledge and information from massive amounts of data, the SOM in contrary of the MLP do not use error correction learning, instead the SOM can search for hidden structures within the data and display hidden relationships without supervision (Deboeck & Kohonen, 1998). A SOM can be used for knowledge discovery, that is, the whole interactive process of discovery of novel patterns or structures in the data. This knowledge can then be extracted to an expert system which is responsible for finding other observations that display the selected characteristics, last, a neural network trained with error correction learning is used to further refine the expert system, thus these artificial intelligence methods can be used in the whole process from knowledge discovery or gathering to prediction or classification.

### **5.3. Further research**

Further research using the concept devised in this work could include:

- Evaluating other patterns or anomalies occurring in stocks or indices.
- Using other neural network types to conduct the classification.
- Applying the methods on other stocks and indices.

The combination of expert systems and artificial neural networks together with other artificial intelligence techniques such as genetic algorithms and fuzzy logic is an interesting area, the potential real world applications are numerous. Systems utilizing these techniques are able to combine the best characteristics of man and machine, the learning ability of the human brain and the massive processing capability of modern computers.

## References

- Aczel, A.D. (1996). *Complete Business Statistics* (3d ed). NY: McGraw-Hill.
- Alvager, T.K.E. & Beach, D.P. (1992). *Handbook for Scientific and Technical Research*. NJ: Prentice-Hall.
- Alvesson, M. & Sköldböck, K. (1994). *Tolkning och Reflektion Vetenskapsfilosofi och Kvalitativ Metod*. Lund: Studentlitteratur.
- Andersen, H. (Ed.). (1994). *Vetenskapsteori och Metodlära En Introduktion*. Lund: Studentlitteratur.
- Aronson, J. E., Turban, E. (2001). *Decision Support Systems And Intelligent Systems* (6 th ed). NJ: Prentice Hall.
- Asakawa, K., Kimoto, T., Takeoka, M., & Yoda, M. (1996). Stock Market Prediction System With Modular Neural Networks. In R.R. Trippi, & E. Turban (Eds.), *Neural Networks in Finance and Investing*. (pp. 497-510). Irwin.
- Azoff, E.M. (1994). *Neural Network Time Series Forecasting Of Financial Markets*. NY: Wiley.
- Back, A., Burns, I., Giles, L.C., Lawrence, S. & Tsoi, A.C. (1998). Neural Network Classification and Prior Class Probabilities. In G.B. Orr, & K.-R. Muller. (Eds.), *Neural Networks: Tricks of the Trade* (pp. 299-313). Berlin: Springer.
- Bergeson, K., & Wunsch II, D.C. (1996). A Commodity Trading Model Based On A Neural Network-Expert System Hybrid. In R.R. Trippi, & E. Turban (Eds.), *Neural Networks in Finance and Investing*. (pp. 581-587). Irwin.
- Bollinger, J. (2002). *Bollinger on Bollinger Bands*. NY: McGraw-Hill.
- Boussabaine, A. H., Wanous, M. (2000). A Neurofuzzy Model for Predicting Business Bankruptcy. In B. Edisbury, P.J.G. Lisboa, & A. Vellido (Eds.), *Business Applications of Neural Networks*. (pp. 55-72). Singapore: World Scientific.
- Chenoweth, T., Obradovic, Z. & Stepen-Lee, S. (1996). Embedding Technical Analysis Into Neural Network Based Trading Systems. *Applied Artificial Intelligence*, 10, (pp. 523-541).
- Chien-Hua, L., Fadlalla, A. (2001). An Analysis of the Applications of Neural Networks in Finance. *Interfaces*, 31: 4 July-August 2001, (pp. 112-122).
- Cordella, L.P., De Stefano, C., Sansone, C., Tortorella, F. & Vento, M. (1998). Neural Network Classification Reliability: Problems and Applications. In C.T. Leondes (Ed.), *Image Processing and Pattern Recognition*. London: Academic Press.
- Deboeck, G., & Kohonen, T. (Eds.). (1998). *Visual Explorations in Finance with Self-Organizing Maps*. Berlin: Springer.

- Edwards, R.D., & Magee, J. (1998). *Technical Analysis of Stock Trends* (7 th Ed). FL: CRC Press LLC.
- Euliano, N.R., Lefebvre C.W., & Principe J.C. (2000). *Neural and Adaptive Systems: Fundamentals Through Simulations*. Wiley.
- Farley, A.S. (2001). *THE MASTER SWING TRADER Tools and Techniques to Profit from Outstanding Short Term Trading Opportunities*. NY: McGraw-Hill.
- Gately, E. (1996). *Neural Networks for Financial Forecasting*. Wiley.
- Hassibi, K., (2000). Detecting Payment Card Fraud with Neural Networks. In B. Edisbury, P.J.G. Lisboa, & A. Vellido (Eds.), *Business Applications of Neural Networks*. (pp.141-157). Singapore: World Scientific.
- Haykin, S. (1999). *Neural Networks A Comprehensive Foundation* (2 nd Ed). NJ: Prentice Hall.
- Hawley, D.D., Johnson, J.D., & Raina, D. (1996). Artificial Neural Systems: A New Tool for Financial Decision Making. In R.R. Trippi, & E. Turban (Eds.), *Neural Networks in Finance and Investing*. (pp. 25-44). Irwin.
- Kryzanowski, L., Galler, M & Wright, D.W. (1996). Using Artificial Neural Networks to Pick Stocks. In R.R. Trippi, & E. Turban (Eds.), *Neural Networks in Finance and Investing*. (pp. 525-541). Irwin.
- Nilsson, P., & Torssell, J. (1998). *Boken Om Trading – Tillämpad teknisk analys*. Borsinsikt AB.
- Pring, M.J. (1993). *Martin Pring on Market Momentum*. USA: Shepherd.
- Schwager, J. (1992). *The New Market Wizards*. Wiley.
- Swales, G., & Yoon, Y. (1996). *Predicting Stock Price Performance: A Neural Network Approach*. In R.R. Trippi, & E. Turban (Eds.), *Neural Networks in Finance and Investing*. (pp. 483-465). Irwin.
- Lee, R.S.T. & Liu, J.N.K. (2001). NORN Predictor – Stock Prediction Using A Neural Oscillatory – Based Recurrent Network. *International Journal of Computational Intelligence and Applications*, 4, (pp. 439-451).
- Leigh, W., Paz, M. & Purvis, R. (2002). An analysis of a hybrid neural network and pattern recognition technique for predicting short-term increases in the NYSE composite index. *OMEGA*, 30, (pp. 69-76).
- Leigh, W., Purvis, R. & Ragusa J.M. (2002). Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network and genetic algorithm: a case study in romantic decision support. *Decision Support Systems*, 32, (pp. 361.377).

Lin, W., Ming, D. & Phua, P.K.H. (2001). Neural Network With Genetically Evolved Algorithms For Stocks Prediction. *Asia-Pacific Journal of Operational Research*, 18, (pp. 103-107).

Medsker, L.R. (1994). *Hybrid Neural Network And Expert Systems*. Dordrecht: Kluwier.

Poh, H-L., Tan, C.L. & Yao, J. (1999). Neural Networks For Technical Analysis: A Study On KLCI. *International Journal of Theoretical and Applied Finance*, 2, (pp. 221-241).

Sun, Y. & Nekovei, R. (1998). Medical Imaging. In C.T. Leondes (Ed.), *Image Processing and Pattern Recognition*. London: Academic Press.

Thurén, T. (1991). *Vetenskapsteori för nybörjare*. Stockholm: Liber.