Chapter 1

DATA MINING FOR FINANCIAL APPLICATIONS

Boris Kovalerchuk
Central Washington University, USA

Evgenii Vityaev
Institute of Mathematics, Russian Academy of Sciences, Russia

Abstract This chapter describes data mining in finance by discussing financial tasks, specifics of methodologies and techniques in this data mining area. It includes time dependence, data selection, forecast horizon, measures of success, quality of patterns, hypothesis evaluation, problem ID, method profile, attribute-based and relational methodologies. The second part of the chapter discusses data mining models and practice in finance. It covers use of neural networks in portfolio management, design of interpretable trading rules and discovering money laundering schemes using decision rules and relational data mining methodology.

Keywords: finance time series, relational data mining, decision tree, neural network, success measure, portfolio management, stock market, trading rules.

> October. This is one of the peculiarly dangerous months to speculate in stocks in. The others are July, January, September, April, November, May, March, June, December, August and February. Mark Twain, 1894

1. INTRODUCTION: FINANCIAL TASKS

Forecasting stock market, currency exchange rate, bank bankruptcies, understanding and managing financial risk, trading futures, credit rating, loan management, bank customer profiling, and money laundering analyses are core financial tasks for data mining (Nakhaeizadeh *et. al.*, 2002). Some of these tasks such as bank customer profiling (Berka, 2002) have many similarities with data mining for customer profiling in other fields.

Stock market forecasting includes uncovering market trends, planning investment strategies, identifying the best time to purchase the stocks and what

stocks to purchase. Financial institutions produce huge datasets that build a foundation for approaching these enormously complex and dynamic problems with data mining tools. Potential significant benefits of solving these problems motivated extensive research for years.

Almost every computational method has been explored and used for *financial modeling*. We will name just a few recent studies: Monte-Carlo simulation of option pricing, finite-difference approach to interest rate derivatives, and fast Fourier transform for derivative pricing (Huang *et. al*., 2004; Zenios, 1999; Thulasiram and Thulasiraman, 2003). New developments augment traditional technical analysis of stock market curves (Murphy, 1999) that has been used extensively by financial institutions. Such stock charting helps to identify buy/sell signals (timing "flags") using graphical patterns.

Data mining as a process of *discovering useful patterns, correlations* has its own niche in financial modeling. Similarly to other computational methods almost every data mining method and technique has been used in financial modeling. An incomplete list includes a variety of linear and non-linear models, multi-layer neural networks (Kingdon, 1997; Walczak, 2001; Thulasiram *et. al*., 2002; Huang *et. al*., 2004), k-means and hierarchical clustering; k-nearest neighbors, decision tree analysis, regression (logistic regression; general multiple regression), ARIMA, principal component analysis, and Bayesian learning.

Less traditional methods used include rough sets (Shen and Loh, 2004), relational data mining methods (deterministic inductive logic programming and newer probabilistic methods (Muggleton, 2002; Lachiche and Flach, 2002; Kovalerchuk and Vityaev, 2000), support vector machine, independent component analysis, Markov models and hidden Markov models.

Bootstrapping and other evaluation techniques have been extensively used for improving data mining results. Specifics of financial time series analyses with ARIMA, neural networks, relational methods, support vector machines and traditional technical analysis is discussed in (Back and Weigend, 1998; Kovalerchuk and Vityaev, 2000; Muller *et. al*., 1997; Murphy, 1999; Tsay, 2002).

The naïve approach to data mining in finance assumes that somebody can provide a cookbook instruction on “how to achieve the best result”. Some publications continue to foster this unjustified belief. In fact, the only realistic approach proven to be successful is providing comparisons between different methods showing their strengths and weaknesses relative to problem characteristics (problem ID) conceptually and leaving for user the selection of the method that likely fits the specific user problem circumstances. In essence this means clear understanding that data mining in general, and in finance specifically, is still more art than hard science.

Fortunately now there is growing number of books that discuss issues of matching tasks and methods in a regular way (Dhar and Stein ,1997; Kovaler-

chuk and Vityaev, 2000; Wang, 2003). For instance, understanding the power of first-order If-Then rules over the decision trees can significantly change and improve data mining design. User's actual experiments with data provide a real judgment of data mining success in finance. In comparison with other fields such as geology or medicine, where test of the forecast is expensive, difficult, and even dangerous, a trading forecast can be tested next day in essence without cost and capital risk involved in real trading.

Attribute-based learning methods such as neural networks, the nearest neighbors method, and decision trees dominate in financial applications of data mining. These methods are relatively simple, efficient, and can handle noisy data. However, these methods have two serious drawbacks: a limited ability to represent background knowledge and the lack of complex relations. *Relational data mining* techniques that include Inductive Logic Programming (ILP) (Muggleton, 1999; Džeroski, 2002) intend to overcome these limitations.

Previously these methods have been relatively computationally inefficient (Thulasiram, 1999) and had rather limited facilities for handling numerical data (Bratko and Muggleton, 1995). Currently these methods are enhanced in both aspects (Kovalerchuk and Vityaev, 2000)and are especially actively used in bioinformatics(Turcotte *et. al*., 2001; Vityaev *et. al*., 2002). We believe that now is the time for applying these methods to financial analyses more intensively especially to those analyses that deal with probabilistic relational reasoning.

Various publications have estimated the use of data mining methods like hybrid architectures of neural networks with genetic algorithms, chaos theory, and fuzzy logic in finance. "Conservative estimates place about $5 billion to $10 billion under the direct management of neural network trading models. This amount is growing steadily as more firms experiment with and gain confidence with neural networks techniques and methods" (Loofbourrow and Loofbourrow, 1995). Many other proprietary financial applications of data mining exist, but are not reported publicly as was stated in (Von Altrock, 1997; Groth, 1998).

2. SPECIFICS OF DATA MINING IN FINANCE

Specifics of data mining in finance are coming from the need to:

- forecast *multidimensional time series* with high level of *noise;*
- accommodate specific efficiency criteria (e.g., the maximum of *trading profit*) in addition to prediction accuracy such as R^2;
- make coordinated *multiresolution forecast* (minutes, days, weeks, months, and years);
- incorporate a *stream of text signals* as input data for forecasting models (e.g., Enron case, September 11 and others);

- be able to *explain the forecast* and the *forecasting model* ("black box" models have limited interest and future for significant investment decisions);
- be able to benefit from very *subtle patterns* with a *short life time;* and
- incorporate the impact of market players on market regularities.

The current *efficient market theory/hypothesis* discourages attempt to discover long-term stable trading rules/regularities with significant profit. This theory is based on the idea that if such regularities exist they would be discovered and used by the majority of the market players. This would make rules less profitable and eventfully useless or even damaging.

Greenstone and Oyer (2000) examine the month by month measures of return for the computer software and computer systems stock indexes to determine whether these indexes' price movements reflect genuine deviations from random chance using the standard t-test. They concluded that although Wall Street analysts recommended to use the "summer swoon" rule (sell computer stocks in May and buy them at the end of summer) this rule is not statistically significant. However they were able to confirm several previously known 'calendar effects" such as "January effect" noting meanwhile that they are not the first to warn of the dangers of easy data mining and unjustified claims of market inefficiency.

The market efficiency theory does not exclude that hidden *short-term local conditional regularities* may exist. These regularities can not work "forever," they should be corrected *frequently*.

It has been shown that the financial data are not random and that the efficient market hypothesis is merely a subset of a larger *chaotic market hypothesis* (Drake and Kim, 1997). This hypothesis does not exclude successful short term forecasting models for prediction of chaotic time series (Casdagli and Eubank, 1992).

Data mining does not try to accept or reject the efficient market theory. Data mining creates *tools* which can be useful for discovering subtle short-term conditional patterns and trends in wide range of financial data. This means that retraining should be a permanent part of data mining in finance and any claim that a silver bullet trading has been found should be treated similarly to claims that a perpetuum mobile has been discovered.

The impact of market players on market regularities stimulated a surge of attempts to use ideas of *statistical physics* in finance (Bouchaud and Potters, 2000). If an observer is a large marketplace player then such observer can potentially change regularities of the marketplace dynamically. Attempts to forecast in such dynamic environment with thousands active agents leads to much more complex models than traditional data mining models designed for. This is one of the major reasons that such interactions are modeled using ideas from statistical physics rather than from statistical data mining. The *physics*

approach in finance (Voit, 2003; Ilinski, 2001; Mantegna and Stanley, 2000; Mandelbrot, 1997) is also known as "econophysic" and "physics of finance". The major difference from data mining approach is coming from the fact that in essence the data mining approach is not about developing specific methods for financial tasks, but the physics approach is. It is deeper integrated into the finance subject mater. For instance, Mandelbrot (1997) (known for his famous work on fractals) worked also on proving that the price movement's distribution is scaling invariant.

Data mining approach covers empirical models and regularities derived directly from data and almost only from data with little domain knowledge explicitly involved. Historically, in many domains, deep field-specific theories emerge after the field accumulates enough empirical regularities. We see that the future of data mining in finance would be to generate more empirical regularities and combine them with domain knowledge via generic analytical data mining approach (Mitchell, 1997). First attempts in this direction are presented in (Kovalerchuk and Vityaev, 2000) that exploit power of relational data mining as a mechanism that permits to encode domain knowledge in the first order logic language.

2.1 Time series analysis

A temporal dataset ***T*** called a *time series* is modeled in attempt to discover its main components such as *Long term trend, L(**T**)*, *Cyclic variation, C(**T**)*, *Seasonal variation, S(**T**)* and *Irregular movements, I(**T**)*.Assume that ***T*** is a time series such as daily closing price of a share, or SP500 index from moment $0$to current moment k, then the next value of the time series $T(k+n)$ is modeled by formula 1.1:

$$T(k+n) = L(T) + C(T) + S(T) + I(T) \tag{1.1}$$

Traditionally classical ARIMA models occupy this area for finding parameters of functions used in formula 1.1. ARIMA models are well developed but are difficult to use for highly non-stationary stochastic processes.

Potentially data mining methods can be used to build such models to overcome ARIMA limitations. The advantage of this four-component model in comparison with "black box" models such as neural networks is that components in formula 1.1 have an interpretation.

2.2 Data selection and forecast horizon

Data mining in finance has the same challenge as general data mining in data selection for building models. In finance, this question is tightly connected to the selection of the target variable. There are several options for target variable

y: y=T(k+1), y=T(k+2),. . . ,y=T(k+n), where *y=T(k+1)* represents forecast for the next time moment, and *y=T(k+n)* represents forecast for $n$moments ahead. Selection of dataset $\boldsymbol{T}$ and its size for a specific desired forecast horizon n is a significant challenge.

For stationary stochastic processes the answer is well-known a better model can be built for longer training duration. For financial time series such as SP500 index this is not the case (Mehta and Bhattacharyya, 2004). Longer training duration may produce many and contradictory profit patterns that reflect bear and bull market periods. Models built using too short durations may suffer from overfitting and hardly applicable to the situations where market is moving from the bull period to the bear period. Also in finance the long-horizon returns could be forecast better than short-horizon returns depending on the training data used and model parameters (Krolzig *et. al*., 2004).

In standard data mining it is typically assumed that the quality of the model does not depend on *frequency* of its use. In financial application the frequency of trading is one of the parameters that impact a quality of the model. This happens because in finance the *criterion of the model quality* is not limited by the accuracy of prediction, but is driven by profitability of the model. It is obvious that frequency of trading impacts the profit as well as the trading rules and strategy.

2.3 Measures of success

Traditionally the quality of financial data mining forecasting models is measured by the standard deviation between forecast and actual values on training and testing data. This approach works well in many domains, but this assumption should be revisited for trading tasks. Two models can have the same standard deviation but may provide very different trading return. The small R^2 is not sufficient to judge that the forecasting model will correctly forecast stock change direction (sign and magnitude). For more detail see (Kovalerchuk and Vityaev, 2000). More appropriate measures of success in financial data mining are measures such as Average Monthly Excess Return (AMER) and Potential trading profits (PTP) (Greenstone and Oyer, 2000):

$$AMER_j = R_{ij} - \beta_i R_{500j} - (\sum_{j=1}^{12} (R_{ij} - \beta_i R_{500j})/12)$$

where R_{ij} is the average return for the S&P500 index in industry i and month j and R_{500j} is the average return of the S&P 500 in month j. The β_i values adjust the AMER for the index's sensitivity to the overall market. A second measure of return is Potential Trading Profits (PTP):

$$PTP_{ij} =_{ij} -R_{500j}$$

PTP shows investor's trading profit versus the alternative investment based on the broader S&P 500 index.

2.4 QUALITY OF PATTERNS AND HYPOTHESIS EVALUATION

An important issue in data mining in general and in finance in particular is the evaluation of quality of discovered pattern P measured by its statistical significance. A typical approach assumes the testing of the null hypothesis H that pattern P is not statistically significant at level α. A meaningful statistical test requires that pattern parameters such as the month(s) of the year and the relevant sectoral index in a trading rule pattern P have been chosen *randomly* (Greenstone and Oyer, 2000). In many tasks this is not the case.

Greenstone and Oyer argue that in the summer "summer swoon" trading rule mentioned above, the parameters are not selected randomly, but are produced by data snooping – checking combination of industry sectors and months of return and then reporting only a few "significant" combinations. This means that rigorous test would require to test a different null hypothesis not only about one "significant" combination, but also about the *"family" of combinations.* Each combination is about an individual industry sector by month's return. In this setting the return for the "family" is tested versus the overall market return.

Several testing options are available. Sullivan *et. al.* (1998, 1999) use a bootstrapping method to evaluate statistical significance of such hypotheses adjusted for the effects of data snooping in "trading rules" and calendar anomalies. Greenstone and Oyer (2000) suggest a simple computational method – combining individual *t-test* results by using the Bonferroni inequality that given any set of events A_1, A_2, . . ., A_n, the probability of their union is smaller than or equal to the sum of their probabilities:

$$\mathrm{P}(\mathrm{A}_1 \,\&\, \mathrm{A}_2 \&. . . \&\, \mathrm{A}_k) \quad \leq \quad \Sigma_{i=1:k} \mathrm{P}(\mathrm{A}_i)$$

Where A_i denotes the false rejection of statement i, from a given family with k statements. One of the techniques to keep the family-wide error rate at reasonable levels is "Bonferroni correction" that sets a significance level of α/k for each of the k statements.

Another option would be to test whether the statements are jointly true using the traditional *F-test.* However if the null hypothesis about a joint statement is *rejected* it does not identify the profitable trading strategies (Greenstone and Oyer, 2000).

The sequential semantic probabilistic reasoning that uses F-test addresses this issue (Kovalerchuk and Vityaev, 2000). We were able to identify profitable and statistically significant patterns for SP500 index using this method. Informally

the idea of semantic probabilistic reasoning is coming from the principle of *Occam's razor* (a law of simplicity) in science and philosophy. Informally for trading it was written as follows:

- When you have two competing trading theories which make exactly *the same predictions*, the one that is simpler is the better & more profitable one.
- If you have two trading/investing theories which *both explain* the observed facts then you should use the simplest one until more evidence comes along.
- The *simplest explanation* for a commodity or stock price movement phenomenon is more likely to be accurate than more complicated explanations.
- If you have two *equally likely solutions* to a trading or day trading problem, pick the simplest.
- The price movement explanation requiring the *fewest assumptions* is most likely to be correct.

3. ASPECTS OF DATA MINING METHODOLOGY IN FINANCE

Data mining in finance typically follows a set of general for any data mining task steps such as problem understanding, data collection and refining, building a model, model evaluation and deployment (Klösgen and Zytkow, 2002). Some specifics of these steps for trading tasks are presented in (Zemke, 2002; Zemke, 2002) such as data enhancing techniques, predictability tests, performance improvements, and pitfalls to avoid.

Another important step in this process is adding expert-based rules in data mining loop when dealing with absent or insufficient data. "Expert mining" is a valuable additional source of regularities. However in finance, expert-based learning systems respond slowly to the to market changes (Cowan, 2002). A technique for efficiently mining regularities from an *expert's perspective* has been offered (Kovalerchuk and Vityaev, 2000). Such techniques need to be integrated into financial data mining loop similar to what was done for medical data mining applications (Kovalerchuk *et. al.*, 2001).

3.1 Attribute-based and relational methodologies

Several parameters characterize data mining methodologies for financial forecasting. Data categories and mathematical algorithms are most important among them. The first data type is represented by *attributes* of objects,

that is each object x is given by a set of values $A_1(x)$, $A_2(x)$,. . . ,$A_n(x)$. The common data mining methodology assume this type of data and it is known as an *attribute-based* or *attribute-value methodology*. It covers a wide range of statistical and connectionist (neural network) methods.

The *relational data type* is a second type, where objects are represented by their relations with other objects, for instance, x>y, y<z, x>z. In this example we may not know that x=3, y=1 and z=2. Thus attributes of objects are not known, but their relations are known. Objects may have different attributes (e.g., x=5, y=2, and z= 4), but still have the same relations. Less traditional *relational methodology* is based on the relational data type.

Another data characteristic important for financial modeling methodology is an actual *set of attributes* involved. A fundamental analysis approach incorporates all available attributes, but technical analysis approach is based only on a time series such as stock price and parameters derived from it. Most popular time series are index value at open, index value at close, highest index value, lowest index value and trading volume and lagged returns from the time series of interest. Fundamental factors include the price of gold, retail sales index, industrial production indices, and foreign currency exchange rates. Technical factors include variables that are derived from time series such as moving averages.

The next characteristic of a specific data mining methodology is a form of the relationship between objects. Many data mining methods assume a *functional form* of the relationship. For instance, the linear discriminant analysis assumes linearity of the border that discriminates between two classes in the space of attributes. Often it is hard to justify such functional form in advance. Relational data mining methodology in finance does not assume a functional form for the relationship. Its intention is *learning symbolic relations* on numerical data of financial time series.

3.2 Attribute-based relational methodologies

In this section we discuss a combination of both attribute-based and relational methodologies that permit to mitigate their difficulties. In most of the publications relational data mining was associated with *Inductive Logic Programming* (ILP) which is a deterministic technique in its purest form. The typical claim about relational data miming is that it can not handle large data sets (Thulasiram, 1999). This statement is based on the assumption that initial data are provided in the form of relations. For instance, to mine in a training data with $m$attributes for n data objects we need to store and operate with $n \times m$ data elements, but for m simplest binary relations (used to represent graphs) we need to store and operate with $n^2 \times m$ elements. This number is $n$times larger and for large training datasets the difference can be very significant. The *attribute-*

based relational data mining does not need to store and operate with $n^2 \times m$ elements. It computes relations from attribute-based data set on demand. For instance, to explore a relation, *Stock(t)>Stock(t+k)* for k days ahead we do not need to store this relation. It can be computed from for every pair of stock data as needed to build a graph of stock relations. In finance with predominantly numeric input data, a dataset that should be represented in a relational form from the beginning can be relatively small.

We share Thuraisingham's (1999) vision that relational data mining is most suitable for applications where *structure can be extracted from the instances.* We also agree with her statement that data mining is now very much an art and to make it into a science, we need more work in areas like ILP that is a part of relational learning that includes probabilistic learning.

3.3 Problem ID and method profile

Selection of a method for discovering regularities in financial time series is a very complex task. Uncertainty of problem descriptions and method capabilities are among the most obvious difficulties in this process. Dhar and Stein (1997) introduced and applied a unified vocabulary for business computational intelligence problems and methods that provide a framework for matching problems and methods.

A problem is described using a set of *desirable values* (problem ID profile) and a method is described using its *capabilities* in the same terms. Use of unified terms (*dimensions*) for problems and methods enhances capabilities of comparing alternative methods. Introducing dimensions also accelerates their clarification. Next, users should not be forced to spend time determining a method's capabilities (values of dimensions for the method). This is a task for developers, but users should be able to identify desirable values of dimensions using natural language terms as suggested by (Dhar and Stein ,1997).

Along these lines Table 1.1 indicates three shortcomings of neural networks for stock price forecasting related to explainability, usage of logical relations and tolerance for sparse data.

The strength of neural networks is also indicated by lines where requested capabilities are satisfied by neural networks. The advantages of using neural network models include the ability to model highly complex functions and to use a high number of variables including both fundamental and technical factors.

3.4 Relational data mining in finance

Decision tree methods are very popular in data mining applications in general and in finance specifically. They provide a set of human readable, consistent rules, but discovering small trees for complex problems can be a significant challenge in finance (Kovalerchuk and Vityaev, 2000). In addition, rules ex-

Table 1.1. Comparison of model quality and resources

Dimension	*Desirable value for stock price forecast problem*	*Capability of a neural network method*
Accuracy	Moderate	High
Explainability	Moderate to High	Low to Moderate
Response speed	Moderate	High
Ease to use logical relations	High	Low
Ease to use numerical attributes	High	High
Tolerance for *noise* in data	High	Moderate to high
Tolerance for *sparse data*	High	Low
Tolerance for complexity	High	High
Independence from experts	Moderate	High

tracted from decision trees fail to compare two attribute values as it is possible with relational methods.

It seems that relational data mining methods also known as relational knowledge discovery methods are gaining momentum in different fields (Muggleton, 2002; Džeroski, 2002; Thulasiram, 1999; Neville and Jensen, 2002; Vityaev *et. al*., 2002).

Data mining in finance not only follows this trend but also leads the application of relational data mining for multidimensional time series such as stock market time series. A. Cowan, a senior financial economist from US Department of the Treasury noticed that examples and arguments available in (Kovalerchuk and Vityaev, 2000) for the application of relational data mining to financial problems produce expectations of great advancements in this field in the near future for financial applications (Cowan, 2002).

It was strengthened in several publications that relational data mining area is moving toward probabilistic first-order rules to avoid the limitations of deterministic systems, e.g., (Muggleton, 2002). Relational methods in finance such as Machine Method for Discovering Regularities (MMDR) (Kovalerchuk and Vityaev, 2000) are equipped with probabilistic mechanism that is necessary for time series with high level of noise.

MMDR is well suited to financial applications given its ability to handle numerical data with high levels of noise (Cowan, 2002). In computational experiments, trading strategies developed based on MMDR consistently outperform trading strategies developed based on other data-mining methods and buy and hold strategy.

4. DATA MINING MODELS AND PRACTICE IN FINANCE

Prediction tasks in finance typically are posed in one of two forms: (1) straight prediction of the market numeric characteristic, e.g., stock return or exchange rate, and (2) the prediction whether the market characteristic will increase or decrease. Having in mind that we need to take into account the trading cost and significance of the trading return in the second case we need to forecast whether the market characteristic will increase or decrease no less than some threshold. Thus, the difference between data mining methods for (1) or (2) can be less obvious, because (2) may require some kind of numeric forecast.

Another type of task is presented in (Becerra-Fernandez *et. al*., 2002). This task is assessment of investing risk. It uses a decision tree technique C5.0 (Quinlan, 1993) and neural networks to a dataset of 52 countries whose investing risk category was assessed in a Wall Street Journal survey of international experts. The dataset included 27 variables (economic, stock market performance/risk and regulatory efficiencies).

4.1 Portfolio management and neural networks

The neural network most commonly used by financial institutions is a multilayer perceptron (MLP) with a single hidden layer of nodes for time series prediction. The peak of research activities in finance based on neural networks was in mid 1990s (Trippi and Turban, 1996; Freedman *et. al*., 1995; Azoff, 1994) that covered MLP and recurrent NN (Refenes, 1995). Other neural networks used in prediction are time delay networks, Elman networks, Jordan networks, GMDH, milti-recurrent networks (Giles *et. al*., 1997).

Below we present typical steps of *portfolio management* using the neural network forecast of return values.

1 Collect 30- 40 historical fundamental and technical factors for stock S_1, say for 10-20 years.

2 Build a neural network NN_1 for predicting the return values for stock S_1.

3 Repeat steps 1 and 2 for every stock S_i,that is monitored by the investor. Say 3000 stocks are monitored and 3000 networks, NN_i are generated.

4 Forecast stock return $S_i(t+k)$for each stock *i and k* days ahead (say a week, seven days) by computing $NN_i(S_i(t))=S(t+k)$.

5 Select n highest $S_i(t+k)$values of predicted stock return.

6 Compute a total forecasted return of selected stocks, T and compute $S_i(t+k)/T$. Invest to each stock proportionally to $S_i(t+k)/T$.

7 Recompute NN_i model for each stock i every k days adding new arrived data to the training set. Repeat all steps for the next portfolio adjustment.

These steps show why neural networks became so popular in finance. Potentially all steps above can be done automatically including actual investment. Even institutional investors may have no resources to manually analyze 3000 stocks and their 3000 neural networks every week. If investment decisions are made more often, say every day, then the motivation to use neural networks with their high adaptability is even more evident.

This consideration also shows current challenges of data mining in finance – the need to build models that can be very quickly evaluated in both accuracy and interpretability. Because NN are difficult to interpret even without time limitation recently steps 1-6 have been adjusted by adding more steps after step 3 that include *extracting interpretable rules* from the trained neural networks and improving prediction accuracy using rules, e.g., (Giles *et. al.*, 1997).

It is likely that extracting rules from the neural network is a *temporary solution*. It would be better to extract rules directly from data without introducing neural network artifacts to rules and potentially overlooking some better rules because of this. It is clear that it can happen from mathematical considerations. There is also a growing number of computational experiments that support this claim, e.g., see (Kovalerchuk and Vityaev, 2000) on experiments with SP500, where first order rules built directly from data outperformed backpropagation neural networks that are most common in financial applications. (Moody and Saffell, 2001) discuss advantages of incremental portfolio optimization and building trading models.

The logic of using data mining in trading futures is similar to portfolio management. The most significant difference is that it is possible to substitute numeric forecast of actual return to less difficult categorical forecast, will it be profitable buy or sell the stock at price S(t) on date t. This is corresponding to long and short terms used in stock market, where *Long* stands for buying the stock and *Short* stands for sell the stock on date t.

4.2 Interpretable trading rules and relational data mining

The logic of portfolio management based on discovering interpretable trading rules is the same as for neural networks with the substitution of NN for rule discovering techniques. Depending on the rule discovering techniques produced rules can be quite different. Below we present categories of rules that can be discovered.

Categorical rules predict a categorical attribute, such as increase/decrease, buy/sell. A typical example of a *monadic categorical rule* is the following rule:

If $S_i(t) <$Value1 and $S_i(t-2) <$Value2 then $S_i(t+1)$ will increase.

In this example, $S_i(t)$is a continuous variable, e.g., stock price at the moment t. If S_i*(t) is* a discrete variable that Value1 and Value2 are taken from m discrete values. This rule is called monadic because it compared a *single attribute* value with a *constant*. Such rules can be discovered from a trained decision trees by tracing its branches to the terminal nodes. Unfortunately decision trees produce only such rules.

The following technical analysis rule is a *relational categorical rule*, because to derive a conclusion it compares values of two attributes such as 5 and 15 day moving averages (ME5 and ME15) and derivatives of moving averages for 10 and 30 days (DerivativeME10, DerivativeME30) :

If ME5(t)=ME15(t) & DerivativeME10(t)>0 DerivativeME30(t)>0 then Buy stock at moment (t+1).

This rule can be read as" If moving averages for 5 and 15 days are equal and derivatives for moving averages for 10 and 30 days are positive then buy stock on the next day. The statement ME5(t)=ME15(t) compares two attribute values. Thus, in this sense classical for stock market technical analysis is superior to decision trees. The presented rule is written in a first order logic form. Note that typically technical analysis rules are not discovered in this form, but relational data mining technique does.

Classical categorical rules assume crisp relations such as $S_i(t)$ <Value1 and ME5(t)=ME15(t). More realistic would be to assume that ME5(t) and ME15(t) are equal only approximately and Value1 is not exact. Fuzzy logic and rough sets rules are used in finance to work with *"soft" relations* (Von Altrock, 1997; Kovalerchuk and Vityaev, 2000; Shen and Loh, 2004). The logic of using *"soft" trading rules* in finance includes the conversion of time series to soft objects, discovering temporal "soft" rule from stock market data, discovering temporal "soft" rule from experts ("
textitexpert mining"), testing consistency of expert rules and rules extracted from data, and finally using rules for forecasting and trading.

4.3 Discovering money laundering and attribute-based relational data mining

4.3.1 Problem statement . Forensic accounting is a field that deals with possible illegal and fraudulent financial transactions. One current focus in this field is the analysis of funding mechanisms for terrorism (Prentice, 2002) where *clean money (*$e.g.$*, charity money) and laundered money are both used* for a variety of activities including acquisition and production of weapons and

their precursors. In contrast, traditional illegal businesses and drug trafficking *make dirty money appear clean*.

The specific tasks in automated forensic accounting related to data mining are the identification of suspicious and unusual electronic transactions and the reduction in the number of 'false positive' suspicious transactions. Currently inexpensive, simple rule-based systems, customer profiling, statistical techniques, neural networks, fuzzy logic and genetic algorithms are considered as appropriate tools (Prentice, 2002).

There are many indicators of possible suspicious (abnormal) transactions in traditional illegal business. These include (1) the use of several related and/or unrelated accounts before money is moved offshore, (2) a lack of account holder concern with commissions and fees (Vangel and James, 2002), (3) correspondent banking transactions to offshore shell banks (Vangel and James, 2002), (4) transferor insolvency after the transfer or insolvency at the time of transfer, (5) wire transfers to new places (Chabrow, 2002), (6) transactions without identifiable business purposes, and (7) transfers for less than reasonably equivalent value.

Some of these indicators can be easily implemented as simple flags in software. However, indicators such as wire transfers to new places produce a large number of 'false positive' suspicious transactions. Thus, the goal is to develop more sophisticated mechanisms based on interrelations among many indicators. To meet these challenges link analysis software for forensic accountants, attorneys and fraud examiners such as NetMap, Analyst's Notebook and others (Chabrow, 2002; i2; Evett *et. al*., 2000) have been and are being developed.

Data mining can assist in discovering patterns of fraudulent activities that are closely related to terrorism such as transactions without identifiable business purposes. The problem is that often an individual transaction does not reveal that it has no identifiable business purpose or that it was done for no reasonably equivalent value. Thus, data mining techniques can search for suspicious patterns in the form of more complex combinations of transactions and other evidence using background knowledge. This means that the training data are formed not by transactions themselves but combination of two, three or more transactions. This implies that the number of training objects exploded. The percentage of suspicion records in the set of all transactions is very small, but the percentage of suspicious combinations in the set of combinations is minuscule. This is a typical task of *discovering rare patterns*. Traditional data mining methods and approaches are ill-equipped to deal this such problems. Relational data mining methods open new opportunities for solving these tasks by discovering "negated patters" described below.

4.3.2 Approach and method. Consider a transactions dataset with attributes such as seller, buyer, item sold, item type, amount, cost, date, company

name, type, company type. We will denote each record in this dataset as (<S>, <B>, <I>), where <S>, <B>, and <I> are sets of attributes about the seller, buyer, and item, respectively.

We may have two linked records R1=(<S1>, <B1>, <I1>) and R2=(<S2>, <B2>, <I2>), such that the first buyer B1 is also a setter S2, B1=S2. It is also possible that the item sold in both records is the same I1=I2. We create a new dataset of pairs of linked records {<R1,R2>}. Data mining methods will work in this dataset to discover suspicious records if samples or definitions of normal and suspicious patterns provided. Below we list such patterns:

- a normal pattern (NP) – a Manufacturer Buys a Precursor & Sells the Result of manufacturing (MBPSR);
- a suspicious (abnormal) pattern (SP) – a Manufacturer Buys a Precursor & Sells the same Precursor (MBPSP);
- a suspicious pattern (SP) – a Trading Co. Buys a Precursor and Sells the same Precursor Cheaper (TBPSPC);
- a normal pattern (NP) – a Conglomerate Buys a Precursor & Sells the Result of manufacturing (CBPSR).

A data mining algorithm A analyzes pairs of records {<R1,R2>} with say 18 attributes total and can match a pair (#5,#6) with a normal pattern MBPSR, A(#5,#6)= MBPSR, while another pair (#1,#3) can be matched with a suspicious pattern, A(#1,#3)= MBPSP.

If definitions of suspicious patterns are given then finding suspicious records is a matter of computationally efficient search is a database that can be distributed. This is not the major challenge. The *automatic generation of patterns/hypotheses descriptions* is a major challenge. One can ask: "Why do we need to discover these definitions (rules) automatically?" A manual way can work if the number of types of suspicious patterns is small and an expert is available. For multistage money-laundering transactions, this is difficult to accomplish manually. Creative criminals and terrorists permanently invent new and more sophisticated money laundering schemes. There is no statistics for such new schemes to learn as it is done in traditional data mining approaches.

An approach based on the idea of "negated patters" can uncover such unique schemes. According to this approach *highly probable patterns* are discovered and then *negated.* It is assumed that a highly probable pattern should be *normal*. In more formal terms, the *main hypothesis (MH)* of this approach is:

If Q is a highly probable pattern (>0.9) then Q constitutes a normal pattern and not(Q) can constitute a suspicious (abnormal) pattern

Below we outline an algorithm based on this hypothesis to find suspicious patterns. Computational experiments with two synthesized databases and few suspicious transactions schemes permitted us to discover transactions. The actual relational data mining algorithm used was algorithm MMRD (Machine Method for Discovery Regularities). Previous research has shown that MMDR based on first-order logic and probabilistic semantic inference is computationally efficient and complete for statistically significant patterns (Kovalerchuk and Vityaev, 2000).

The algorithm finding suspicious patterns based on the main hypotheis (MH) consists of four steps:

1 *Discover* patterns, compute probability of each pattern, select patterns with probabilities above a threshold, say 0.9. To be able to compute conditional probabilities patterns should have a rule form: IF A then B. Such patterns can be extracted using decision tree methods for relatively simple rules and using relational data mining for discovering more complex rules. Neural Network (NN) and regression methods typically have no if-part. With additional effort rules can be extracted from NN and regression equations.

2 *Negate* patterns and compute *probability* of each negated pattern,

3 Find records database that satisfy negated patterns and analyze these records for possible *false alarm* (records maybe normal not suspicious).

4 Remove false alarm records and provide detailed analysis of suspicious records.

5. CONCLUSION

To be successful a data mining project should be driven by the application needs and results should be tested quickly. Financial applications provide a unique environment where efficiency of the methods can be tested instantly, not only by using traditional training and testing data but making real stock forecast and testing it the same day. This process can be repeated daily for several months collecting quality estimates.

This chapter highlighted problems of data mining in finance and specific requirements for data mining methods including in making interpretations, incorporating relations and probabilistic learning.

The relational data mining methods outlined in this chapter advances pattern discovery methods that deal with complex numeric and non-numeric data, involve structured objects, text and data in a variety of discrete and continuous scales (nominal, order, absolute and so on). The chapter shows benefits of using such methods for stock market forecast and forensic accounting that includes

uncovering money laundering schemes. The technique combines first-order logic and probabilistic semantic inference. The approach has been illustrated with an example of discovery of suspicious patterns in forensic accounting.

Currently the success of data mining exercises has been reported in literature extensively. Typically it is done by comparing simulated trading and forecasting results with results of other methods and real gain/loss and stock. For instance, recently Huang *et. al*. (2003) claimed that data mining methods achieved better performance than traditional statistical methods in predicting credit ratings. Much less has been reported publicly on success of data mining in real trading by financial institutions. It seems that the market efficiency theory is applicable to reporting success. If real success is reported then competitors can apply the same methods and the leverage will disappear because in essence all fundamental data mining methods are not proprietary.

Next future direction is developing practical decision support software tools that make easier to operate in data mining environment specific for financial tasks, where hundreds and thousands of models such as neural networks, and decision trees need to be analyzed and adjusted every day with a new data stream coming every minute. E.g., Tsang, Yung, Li (2003) reported an architecture for learning from and monitoring the stock market.

Inside of the field of data mining in finance we expect an extensive growth of *hybrid methods* that combine different models and provide a better performance than can be achieved by individuals. In such integrative approach individual models are interpreted as *trained artificial "experts"*. Therefore their combinations can be organized similar to a consultation of real *human experts*. Moreover, these artificial experts can be effectively combined with real experts. It is expected that these artificial experts will be built as autonomous *intelligent software agents*. Thus "experts" to be combined can be data mining models, real financial experts, trader and virtual *experts* that runs trading rules extracted from real experts. A virtual expert is a software intelligent agent that is in essence an expert system. We coined a new term **"***expert mining***"** as an umbrella term for extracting knowledge from real human experts that is needed to populate virtual experts.

We expect that in coming years data mining in finance will be shaped as a distinct field that blends knowledge from finance and data mining, similar to what we see now in bioinformatics where integration of field specifics and data mining is close to maturity. We also expect that the blending with ideas from the theory of dynamic systems, chaos theory, and physics of finance will deepen.

References

Azoff, E., Neural networks time series forecasting of financial markets, Wiley, 1994.

Back, A., Weigend, A., A first application of independent component analysis to extracting structure from stock returns. Int. J. on Neural Systems, 8(4):473–484, 1998.

Becerra-Fernandez, I., Zanakis, S. Walczak,S., Knowledge discovery techniques for predicting country investment risk, Computers and Industrial Engineering Vol. 43 , Issue 4:787 – 800, 2002.

Berka, P. PKDD Discovery Challenge on Financial Data, In: Proceedings of the First International Workshop on Data Mining Lessons Learned, (DMLL-2002), 8-12 July 2002, Sydney, Australia.

Bouchaud, J., Potters,M., Theory of Financial Risks: From Statistical Physics to Risk Management, 2000, Cambridge Univ. Press, Cambridge, UK.

Bratko, I., Muggleton, S., Applications of Inductive Logic Programming. Communications of ACM, 38(11): 65-70, 1995.

Casdagli, M., Eubank S., (Eds). Nonlinear modeling and forecasting, Addison Wesley, 1992.

Chabrow, E. Tracking the terrorists, Information week, Jan. 14, 2002, http://www.tpirsrelief.com/forensic_accounting.htm

Cowan, A., Book review: Data Mining in Finance, International journal of forecasting, Vol.18, Issue 1, 155-156, Jan-March 2002.

Dhar, V., Stein,R., Intelligent decision support methods, Prentice Hall, 1997.

Džeroski S., Inductive Logic programming Approaches, In: Klösgen W., Zytkow J. Handbook of data mining and knowledge discovery, Oxford Univ. Press, 2002, 348-353.

Drake, K., Kim Y., Abductive information modeling applied to financial time series forecasting, In: Nonlinear financial forecasting, Finance and Technology, 1997, 95-109.

Evett, IW., Jackson, G. Lambert, JA , McCrossan, S. The impact of the principles of evidence interpretation on the structure and content of statements. Science and Justice, 40, 2000, 233–239.

Freedman R., Klein R., Lederman J., Artificial intelligence in the capital markets, Irwin, Chicago, 1995.

Giles, G., Lawrence S., Tshoi, A. Rule inference for financial prediction using recurrent neural networks, In: Proc. Of IEEE/IAAFE Conference on Computational Intelligence for financial Engineering, IEEE, NJ, 1997, 253-259.

Groth, R., Data Mining, Prentice Hall, 1998.

Greenstone, M., Oyer, P., Are There Sectoral Anomalies Too? The Pitfalls of Unreported Multiple Hypothesis Testing and a Simple Solution, Review of Quantitative Finance and Accounting, 15, 2000: 37-55, http://faculty-gsb.stanford.edu/oyer/wp/tech.pdf

Haugh, M., Lo, A., Computational Challenges in Portfolio Management, Tomorrow's Hardest Problems, IEEE Computing in Science and Engineering, May/June 2001, 54-59.

Huang, Z, Chen H, Hsu C.-J., Chen W.-H., Wu S., Credit rating analysis with support vector machines and neural networks: a market comparative study, Decision support systems, 2004, in press.

Ilinski, K., Physics of Finance: Gauge Modeling in Non-Equilibrium Pricing, Wiley, 2001

i2 Applications-Fraud Investigation Techniques, `http://www.i2.co.uk/Products/`

Kingdon, J., Intelligent systems and financial forecasting. Springer, 1997.

Klösgen W., Zytkow J. Handbook of data mining and knowledge discovery, Oxford Univ. Press, Oxford, 2002.

Kovalerchuk, B., Vityaev, E., Data Mining in Finance: Advances in Relational and Hybrid Methods, Kluwer, 2000.

Kovalerchuk, B., Vityaev E., Ruiz J.F., Consistent and Complete Data and "Expert Mining" in Medicine, In: Medical Data Mining and Knowledge Discovery, Springer, 2001, 238-280.

Krolzig, M., Toro, J., Multiperiod Forecasting in Stock Markets: A Paradox Solved, Decision Support Systems, 2004 (in print).

Lachiche, N., Flach, P.A True First-Order Bayesian Classifier. 12th International Conference, ILP 2002, Sydney, Australia, July 9-11, 2002. Lecture Notes in Computer Science 2583 Springer 2003,133-148.

Loofbourrow, J., Loofbourrow, T., What AI brings to trading and portfolio management, In: Freedman R., Klein R., Lederman J., Artificial intelligence in the capital markets, Irwin, Chicago, 1995, 3-28.

Mandelbrot, B., Fractals and scaling in finance, Springer, 1997

Mantegna, R., Stanley, H., An Introduction to Econophysics: Correlations and Complexity in Finance, Cambridge Univ. Press, Cambridge, UK, 2000

Mehta, K., Bhattacharyya S., Adequacy of Training Data for Evolutionary Mining of Trading Rules, Decision support systems, 2004 (in print).

Mitchell, T., Machine learning. 1997, McGraw Hill.

Moody, J. Saffell, M. Learning to trade via direct reinforcement, IEEE transactions on neural Networks, Vol. 12, No. 4, 2001, 875-889.

Muller, K.-R., Smola, A., Rtsch, G., Schlkopf, B., Kohlmorgen, J., & Vapnik, V., 1997. Using support vector machines for time series prediction, In: Advances in Kernel Methods – Support Vector Learning, MIT Press, 1997.

Murphy, J. Technical analysis of the financial markets: A comprehensive guide to trading methods and applications, Prentice Hall, 1999.

Muggleton, S., Learning Structure and Parameters of Stochastic Logic Programs, 12th International Conference, ILP 2002, Sydney, Australia, July 9-11, 2002. Lecture Notes in Computer Science 2583 Springer 2003, 198-206.

Muggleton S., Scientific Knowledge Discovery Using Inductive Logic Programming. Communications of ACM, 42(11), 1999, 42-46.

Nakhaeizadeh, G., Steurer, E., Bartmae, K., Banking and Finance, In: Klösgen W., Zytkow J. Handbook of data mining and knowledge discovery, Oxford Univ. Press, Oxford, 2002, 771-780.

Neville, J., Jensen, D. , Supporting relational knowledge discovery: Lessons in architecture and algorithm design, In: Proceedings of the First International Workshop on Data Mining Lessons Learned, (DMLL-2002), 8-12 July 2002, Sydney, Australia.

Prentice, M., Forensic Services-tracking terrorist networks,2002, Ernst & Young, UK.

Quinlan J.R., C4.5: programs for machine learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993.

Refenes A., (Ed.) Neural Networks in the Capital Markets, Wiley, 1995

Shen L., Loh, H., Applying rough sets to market timing decisions, Decision support systems, 2004, in press.

Sullivan, R., Timmermann, A., White, H., Dangers of Data-Driven Inference: The Case of Calendar Effects in Stock Returns. University of California. San Diego Department of Economics, Discussion Paper 98-16, 1998.

Sullivan, R., Timmermann, A., White, H., Data-Snooping, Technical Trading Rule Performance, and the Bootstrap. Journal of Finance 54, 1999, 1647-1691.

Thulasiram, R., Thulasiraman, P., Performance Evaluation of a Multithreaded Fast Fourier Transform Algorithm for Derivative Pricing, Journal of Supercomputing, Vol.26 No.1, 43-58, August 2003.

Thulasiram, R. Jayaraman, S. Sampath, S. Financial Forecasting using Neural Networks under Multithreaded Environment, IIIS Proc. of the 6th World Multiconference on Systems, Cybernetics and Informatics, SCI 2002 , Orlando, FL, USA, July 14-17, 2002, 147-152.

Thuraisingham, B, Data mining: technologies, techniques, tools and trends. CRC Press, 1999

Trippi, R., Turban, E., Neural networks in finance and investing, Irwin, Chicago, 1996,

Tsay, R. ,Analysis of financial time series. Wiley, 2002.

Turcotte, M., Muggleton, S., Sternberg, M., The Effect of Relational Background Knowledge on Learning of Protein Three-Dimensional Fold Signatures. Machine Learning, 43(1/2), 2001, 81-95.

Vangel, D., James A. Terrorist Financing: Cleaning Up a Dirty Business, the issue of Ernst & Young's financial services quarterly, Springer, 2002.

Vityaev E.E., Orlov Yu. L., Vishnevsky O.V., Kovalerchuk B.Ya., Belenok A.S., Podkolodnii N.L., Kolchanov N.A. Knowledge Discovery for Gene Regulatory Regions Analysis, In: Knowledge-Based Intelligent Information Engineering Systems and Allied Technologies, KES 2002. Eds. E. Damiani,

R. Howlett, L.Jain, N. Ichalkaranje, IOS Press, Amsterdam, 2002, part 1, 487-491.

Voit, J., The Statistical Mechanics of Financial Markets, Vol. 2, Springer, 2003.

Von Altrock C. , Fuzzy Logic and NeuroFuzzy Applications in Business and Finance, Prentice Hall, 1997.

Walczak, S., An empirical analysis of data requirements for financial forecasting with neural networks, Journal of Management Information Systems, 17(4), 2001, 203-222, 2001.

Wang, H., Weigend A. ,Data mining for financial decision making, Decision support systems, 2004, in press.

Wang J., Data Mining; opportunities and challenges, Idea Group, London, 2003

Zemke, S. On Developing a Financial Prediction System: Pitfalls and Possibilities, In: Proceedings of the First International Workshop on Data Mining Lessons Learned (DMLL-2002), 8-12 July 2002, Sydney, Australia.

Zemke, S. , Data Mining for Prediction. Financial Series Case, Doctoral Thesis, The Royal Institute of Technology, Department of Computer and Systems Sciences, Sweden, December 2003.

Zenios, S. High Performance Computing in Finance - Last Ten Years and Next, Parallel Computing, Dec. 1999, 2149-2175.