



SUSTAINABLE ECONOMY INSPIRED LARGE-SCALE FEED-FORWARD PORTFOLIO CONSTRUCTION

Šarūnas RAUDYS, Aistis RAUDYS, Židrina PABARŠKAITĖ

*Faculty of Mathematics and Informatics, Vilnius University,
Naugarduko g. 24, 03225 Vilnius, Lithuania*

Received 03 January 2013; accepted 31 May 2013

Abstract. To understand large-scale portfolio construction tasks we analyse sustainable economy problems by splitting up large tasks into smaller ones and offer an evolutionary feed-forward system-based approach. The theoretical justification for our solution is based on multivariate statistical analysis of multidimensional investment tasks, particularly on relations between data size, algorithm complexity and portfolio efficacy. To reduce the dimensionality/sample size problem, a larger task is broken down into smaller parts by means of item similarity – clustering. Similar problems are given to smaller groups to solve. Groups, however, vary in many aspects. Pseudo randomly-formed groups compose a large number of modules of feed-forward decision-making systems. The evolution mechanism forms collections of the best modules for each single short time period. Final solutions are carried forward to the global scale where a collection of the best modules is chosen using a multiclass cost-sensitive perceptron. Collected modules are combined in a final solution in an equally weighted approach (1/N Portfolio). The efficacy of the novel decision-making approach was demonstrated through a financial portfolio optimization problem, which yielded adequate amounts of real world data. For portfolio construction, we used 11,730 simulated trading robot performances. The dataset covered the period from 2003 to 2012 when environmental changes were frequent and largely unpredictable. Walk-forward and out-of-sample experiments show that an approach based on sustainable economy principles outperforms benchmark methods and that shorter agent training history demonstrates better results in periods of a changing environment.

Keywords: portfolio, optimization, clustering, simulation, algorithmic trading, automated trading, Markowitz, portfolio optimization, mean-variance, regularization.

Reference to this paper should be made as follows: Raudys, Š.; Raudys, A.; Pabarškaitė, Ž. 2014. Sustainable economy inspired large-scale feed-forward portfolio construction, *Technological and Economic Development of Economy* 20(1): 79–96.

JEL Classification: C2, G11.

Corresponding author Šarūnas Raudys
E-mail: sarunas.raudys@mif.vu.lt

Introduction

Sustainable economy. In general sense, sustainability is the competence to support, maintain or endure; sustainability can help a community ensure that its social, economic and environmental systems are well integrated and will endure. One of the main sustainability dogmas requires the enhancement of local economic vitality. One must consider this in every step of the process, regardless of whether the process originates in manufacturing or teaching process.

Large-scale processes are often difficult to optimize and raise sustainability issues. This has been noted in several research studies, including algae-based bio fuel production (Board on ... 2012). Therefore, the natural way to combat large-scale optimization problems is to break the problem down into smaller ones and address them locally, later joining the results into one large global solution. Small-scale solutions are easier to make sustainable, as illustrated by the fishery industry (Cochrane *et al.* 2011). We follow this principle by creating and analysing a multilevel feed-forward automated trading system portfolio.

Portfolio construction. Financial problems play one of leading roles in the evolution of modern society. The diversity of assets or financial trading participants is of great importance in this respect. Diversification is also highly important for the sustainable development of society (Jeucken 2001; McCormick 2012). Portfolio construction is a field in which a large quantity of data is available and it is easy to perform computer simulations. In this paper, we use sustainable economy principles in large-scale portfolio construction problems. Here, we may have thousands of potential portfolio members. At the same time, the task of portfolio management (PM) is one of the principal research topics in financial markets. Here we have clearly expressed performance criteria. To create the portfolio, using the formula:

$$x_{Pi} = \sum_{j=1}^N w_{ij}^S x_{rji} = \mathbf{x}_{ri} \mathbf{w}_{rs}^T = q_s, \quad (1)$$

where we form a multidimensional weight vector $\mathbf{w}_r = (w_{r1}, w_{r2}, \dots, w_{rN})$ that determines “optimal” investment proportions for N trading robots. In the analysis in this paper, we suppose that coefficients w_{rj} fulfil certain requirements:

$$w_{rj} \geq 0, \quad \sum_{j=1}^N w_{rj} = 1. \quad (2)$$

To find the weights, one uses L days’ investment profit, i.e. so many days of investment profit and history, x_{ji} ($i = 1, 2, \dots, L; j = 1, 2, \dots, N$).

Automated trading systems. During the last decade, automated trading systems (ATS) and especially high frequency trading (Chan 2008; Aldridge 2010) have become very popular. They potentially promise good returns with relatively small risk (Bookstaber 2009) as positions are held for a short time (from a few seconds to a few hours or a day) and thus losses are also small; this approach minimizes the risk of large losses. A profit is possible due to the very short term market inefficiencies. Usually, the number of ATS is larger than the number of available assets as it is possible to create a number of different types of ATS trading one asset. However, having a high number of ATS, we face large-scale portfolio construction problems.

Large-scale portfolio problems. Large-scale portfolio construction becomes a significant research problem when the number of trading strategies, N , is larger than the number of data points, L . In such cases, small sample size problems arise in portfolio optimization. We have a huge number of inputs and we know little about returns of these time series and correlations between them.

Portfolio of assets vs. trading systems. Historically, the majority of efforts have been aimed at creating portfolios for a set of assets (stocks, bonds, etc.) where we have profit or loss every day. In contrast, we use ATS instead of the assets where we may have profit or loss or zero (zero if ATS is not trading). To date, there have been very few attempts to relate analyses of “economic crises” with portfolio management problems using artificial and swarm intelligence (Cura 2009) approaches.

Portfolio management approaches comprise two different lines of attack: the first is based on the assets and the second on the successes of automated trading systems. ATS trade infrequently and therefore the data matrix is sparse. The data consist of circa 70% of zeros. Distribution of the data becomes bimodal.

Specificity of this paper. In this research, we aim to consider large-scale sustainable economy problems inspired by ATS portfolio construction. Here, a variety of diverse human factors is incorporated in the trading software and plays a prominent role. In our analysis, portfolio creation is not based on assets. This aspect distinguishes it from previous research. The method of portfolio construction implemented uses information on the successes and losses of the ATS that trade the assets. In addition, the underlying trading systems are short term/high frequency, so aggregated returns are not correlated to any great extent to variations in the underlying instruments.

An additional idiosyncrasy of our analysis is the extremely large number (11,730) of potentially useful trading robots and relatively small (2,398) sample size (number of days) used to design portfolio weights in situations where data structure is changing constantly. The literature on this topic is very sparse. Research studies have tended to focus on the trading systems or portfolio construction methods or multi-agent systems (Smeureanu *et al.* 2012) separately. Some have optimized trading system portfolios (Moodyand, Lizhong 1997), (Dempster, Jones 2001) but very few have addressed large-scale trading system portfolio optimization problems (Perold 1984). Multi-agent systems are rarely used in trading, although occasional examples can be found (Araújo, de Castro 2011). In the past, only the authors of this paper have looked into large-scale feed-forward ATS portfolio construction (Raudys, S., Raudys, A. 2011, 2012) and multivariate statistical analysis of multidimensional investment tasks (Raudys 2013) as a whole.

The paper is organized as follows. In Section 1 we describe the data. In Sections 2 and 3 we itemize the portfolio management tasks and describe the novel decision-making methodology. In the last two sections, we present the discussion and conclusions.

1. Data description

Certain economic simulations are difficult to perform due to the lack of data, poor data quality or insufficient data. Thus, in some cases it is useful to choose some related problem that has sufficient data. This is especially acute for large-scale sustainability problems where the number

of factors is huge. Sustainable environment analysis is dispersed in a large number of diverse papers. Today it is impossible to collect large-scale data which has huge amount of factors affecting sustainability with lengthy histories. In analysis of ATS, however, we can analyse thousands of them and generate lengthy data histories. ATSs are also mutually dependent.

1.1. Data source description

Systematic trading firm that trades futures in the global markets provided us with 3 sets of data: $p = 3,133$, $p = 7,708$ and $p = 11,730$. We named them A, B and C accordingly. Here, time series data $X = (x_1, x_2, x_3, \dots, x_p)$. Each time series is created by simulating automated trading systems and by recording daily profit and loss. Time series are very sparse and consist of circa 70 percent of zeros. It means that ATS refuse to trade 7 out of 10 times on average. This is quite a typical behaviour as for example, trend following ATS will not trade until it detects a trend. One typical example is presented in Figure 1. We have daily profit and loss of the system in the top graph and the assets that is being traded in the bottom graph where ATS has occasional zero profit/non-trading periods.

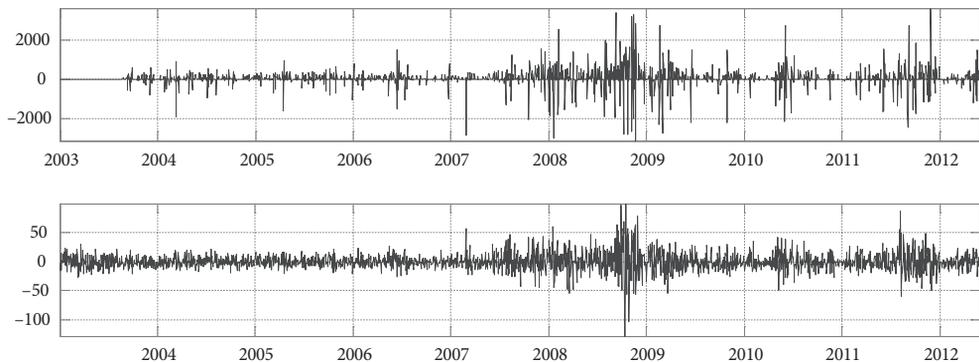


Fig. 1. Typical ATS (top) time series in our $p = 3,133$ sized dataset and the asset (E-mini S&P 500 futures) being traded (bottom) by that ATS

1.2. Summary of datasets

A – Variety of ATS systems including trend following, momentum, mean reversion and seasonality systems. The set was composed over a long period adding the best performing strategies.

B – One specific type of mean reversion system (MRS) was optimised on approximately 30 most liquid futures and from each instrument we selected a set of the best solutions. The MRS typically buys if market is falling and sells if market is rallying in anticipation of a reversal.

C – This dataset is similar to B, but RMS logic is slightly different. In selection procedure we selected bigger set of systems, hence dataset is bigger.

Table 1 below demonstrates characteristics of the data.

Table 1. In this table we present detailed information about datasets

Name	L (days)	p (robots)	% of zeros	From	To
A	2,581	3,133	68.65%	11 Mar 2002	04 Dec 2012
B	2,517	7,708	71.84%	10 Jan 2003	03 Sep 2012
C	2,398	11,730	64.44%	01 Jan 2002	10 Mar 2011

1.3. Individual dataset description

The firm is trading most liquid US and European futures in global exchanges (CME, CBOT, NYBOT, ICE, COMEX, EUREX): stock indexes, energies, metals, commodities, interest rate products, foreign exchange products (E-mini S&P 500, E-mini S&P MidCap 400, E-mini Russell 2000, E-mini, ASDAQ-100, E-mini DOW (\$5), Canadian Dollar, Swiss Franc, Japanese Yen, Australian Dollar, Euro FX, British Pound, Sugar No. 11, Coffee, Soybeans, Gold, Silver, Copper, DAX, EURO STOXX 50, Natural Gas, Crude Oil, 2 Year U.S. Treasury Notes, 5 Yr U.S. Treasury Notes, 10 Yr U.S. Treasury Notes, 30 Yr U.S. Treasury Bonds and others). The set ranges from high frequency and short term automated trading systems running on minute data to systems having trade duration of 5 days. Majority of the systems exit their position within 24 h. All ATS have realistic transaction costs and slippage included. Firm's objective was to create a portfolio that will have robust Sharpe ratio on out of sample data.

Some ATS are very similar and trade almost identically. Correlation coefficients of A ($p = 3,133$) dataset grouped by similar strategies can be viewed in Figure 2. We can see groups of similar strategies (yellow/light squares), caroty areas represent uncorrelated systems and red/darker squares represent negatively correlated systems (very few).

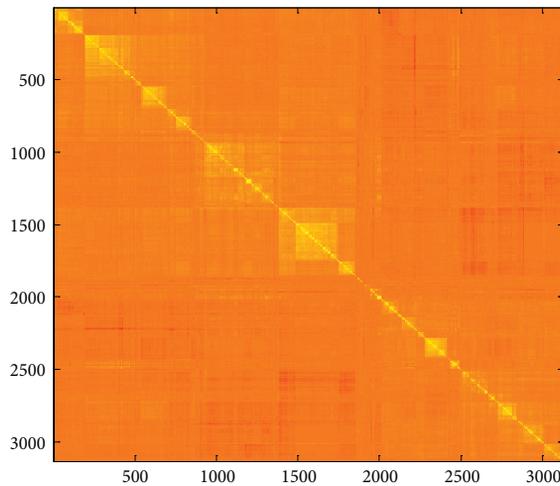


Fig. 2. Correlation matrix of $p = 3,133$ ATS dataset. Yellow/light areas of the figure correspond to highly correlated ATS, red correspond to negatively correlated ATS and caroty areas correspond to uncorrelated ATS

1.4. Typical strategy logic description

Some ATS (or trading strategies or simply strategies) are very different from each other and profit from completely different market inefficiencies. Some strategies work on a high-frequency market data and some work on a daily or even weekly data. At any point, strategy can be long, short or flat, so profits can be generated in rising and falling markets. Some strategies never hold position during the night, some do. Strategies use only technical analysis indicators and pay no attention to the fundamental data.

Trend following systems tend to follow the market direction. If market is rising they buy and if market is falling – sell. Momentum strategies behave summarily but use momentum as a market direction indicator.

Mean reversion systems (also known as contra-trend strategies) tend to take an opposite direction than trending ones. The rationale of this behaviour is that market is moving in cycle/waves. If the market rises significantly and systems see trend developing it will take opposite direction with the anticipation of a correction.

Interestingly, both types of systems can be profitable on the same financial instrument. The difference is the time frame. I.e. market may be trending upwards but on the way it waves a bit where mean reversion strategies can make a profit.

Seasonality type of strategies uses rationale that market repeats itself on specific time frame. Seasonality may not necessarily be yearly, it can be on a monthly, weekly or daily basis. If strategy spots such behaviour, next time it will take the position with anticipation of the same market movement.

1.5. Trading strategy optimization

After creation of ATS, it can be calibrated/optimised to the specific time frame, market or market conditions. The process involves changing strategy parameters (i.e. moving average periods, take profit or stop loss levels, etc.) and calculating simulated profit, Sharpe ratio or other performance measures. Systems with the best results are selected to the next level – inclusion into the portfolio. During ATS optimization procedure a lot of similar solutions (with small differences in parameters and trading patterns) can be produced. Optimization procedure typically is performed using brute force or genetic optimizations methods. This procedure produces a lot of similar systems and typically, a number of systems exceeds the number of data points. Therefore, a robust sustainable portfolio creation procedure is required to select only optimal set of ATS for the portfolio.

2. Problem description

In this section, we will familiarize with portfolio management problem in more detail. In particular, we will pay more attention in sustainable portfolio construction, mean variance portfolio optimization and proposed feed-forward portfolio construction.

2.1. Mean variance portfolio optimization

In the mean-variance framework (Markowitz 1952) one maximizes the sample mean (mean) and standard deviation (std) ratio (a modified Sharpe ratio with no risk free rate) for a selected number (say k values) of *a priori* selected return (profit or loss) values:

$$Sh = \text{mean}(x_{pi}) / \text{std}(x_{pi}), \tag{3}$$

where: $\mathbf{x}_i = (x_{r1i}, x_{r2i}, \dots, x_{rNi})$, “ T ” denotes transpose operation, ($s = 1, 2, \dots, k$), $\text{mean}(x_{pi}) = \bar{\mathbf{x}}_r \mathbf{w}_r^T$, $\text{std}(x_{pi}) = \mathbf{w}_r \mathbf{S}_r \mathbf{w}_r^T$, $\bar{\mathbf{x}}_r$ is N -dimensional estimate of mean vector, $\boldsymbol{\mu}$, of returns and $\mathbf{S}_r = \frac{1}{T-1} \sum_{i=1}^T (\mathbf{x}_{ri} - \bar{\mathbf{x}}_r)^T (\mathbf{x}_{ri} - \bar{\mathbf{x}}_r)$ is an estimate of $N \times N$ dimensional covariance matrix $\boldsymbol{\Sigma}$.

In practice instead of Eq. (3), a scaled ratio $Sh = \text{mean}(x_{pi}) / \text{std}(x_{pi}) \times 15.8745$ is used ($15.8745 = 252 / \sqrt{252}$ – is annualised Sharpe ratio modifier). In Equation (3) $\text{mean}(x_{pi}) = \bar{\mathbf{x}}_r \mathbf{w}_r^T$, $\text{std}(x_{pi}) = \mathbf{w}_r \mathbf{S}_r \mathbf{w}_r^T$, $\bar{\mathbf{x}}_r$ is N -dimensional estimate of mean vector, $\boldsymbol{\mu}$, of returns and $\mathbf{S}_r = \frac{1}{T-1} \sum_{i=1}^T (\mathbf{x}_{ri} - \bar{\mathbf{x}}_r)^T (\mathbf{x}_{ri} - \bar{\mathbf{x}}_r)$ is an estimate of $N \times N$ dimensional covariance matrix $\boldsymbol{\Sigma}$.

To speed up calculations, instead of exact solution, minimization of standard deviations by using the Lagrange multiplier, we used approximate analytic solution. In our research, we minimize cost function, where constraints (2) are incorporated inside:

$$\text{Cost}(q_t) = \mathbf{w}_t \mathbf{S} \mathbf{w}_t^T + \lambda_q (\mathbf{w}_t \bar{\mathbf{X}}^T - q_t)^2 + \lambda_1 (\mathbf{1} \mathbf{w}_t^T - 1)^2.$$

In the above equation, $\mathbf{1} = [1 \ 1 \ \dots \ 1]$ stands for a row vector composed of N “ones”. Scalars λ_q and λ_1 control the constraints violations. Values of λ_q , λ_1 should be sufficiently large to ensure terms, $(\mathbf{w}_t \bar{\mathbf{X}}^T - q_t)^2$ and $(\mathbf{1} \mathbf{w}_t^T - 1)^2$, converge to 0. We presumed $\lambda_q = \lambda_1 = \lambda = 10^8$.

Then optimal weights can be expressed in a straightforward way:

$$\mathbf{w}_t = (q_t \bar{\mathbf{X}} + \mathbf{1})(S / \lambda + \bar{\mathbf{X}}^T \bar{\mathbf{X}} + \mathbf{1}^T \mathbf{1})^{-1}, \tag{4}$$

where: λ plays role of optimization accuracy constant; and $q_t = \mathbf{w}_1 \bar{\mathbf{X}}^T$ is one of F return values used to calculate an efficient frontier (Markowitz 1952).

Note, the optimization accuracy constant λ controls weights magnitude and stands as additional regularisation constant (Brodie *et al.* 2009; Raudys, S., Raudys, A. 2011; Zafeiriou *et al.* 2012; Stuhlsatz *et al.* 2012). After calculating vector \mathbf{w}_s , to satisfy the constraints (2) we brought negative weights to naught, if $w_j \leq 0$ and normalize \mathbf{w}_s to meet constraint (2). Analytical solution was roughly 30 times faster as traditional Lagrange multiplier based optimization procedure realized in Matlab *frontcon* code. Simple analytical expression and high calculation speed is very important when we have to generate a large number of agents differing in subsets of trading robots and training parameters, and training sequence.

2.2. Benefits of portfolio management model for sustainability analysis

The mean/standard deviation ratio used in portfolio construction has two important advantageous features:

1. Provided the number of inputs (assets, or trading robots) is large, *distribution of the weighted sum* (2) can become close to Gaussian (*central limit theorem*).
2. In case of Gaussian returns, maximization of ratio (3) means that mean value of returns is maximized provided probability $Prob(x_{p_i} < P_{\max})$ is fixed *a priori*, where P_{\max} is a freely chosen risk level. Positive feature of the mean and standard deviation criterion is that result does not depend on the choice of P_{\max} .

Another encouraging feature also lies in a fact that, in many practical tasks, good Gaussian fit can be obtained when number of inputs, N , exceeds one hundred even in the case of correlated and obviously non-Gaussian univariate distributions of x_{a2i} . In this way, the mean-variance framework does not require normality of distribution for each single input. *One needs that distribution of the weighted output, $x_{p_i} = \mathbf{x}_{r_i} \mathbf{w}_r^T$, would be close to Gaussian.*

The presence of extremely large number of trading robots is a characteristic peculiarity of automatic asset trading. If a sample size, L , is small and the number of inputs N , is large, instabilities and diminution in the portfolio performance arise. The instabilities can be reduced if regularizing constraints or penalty terms are incorporated in the optimization procedure (Kan, Zhou 2007; DeMiguel et al. 2007). Use of assumptions about a block diagonal structure of matrix Σ , can dramatically reduce the number of parameters to be estimated. In pattern classification, this model has been used for four decades (see e.g. review (Raudys, Young 2004)), now it is successfully applied for portfolio management (Hung et al. 2000; Raudys, Zliobaite 2005). Authors in DeMiguel et al. (2007), Kan, Zhou (2007) divided the assets into three groups. As a result they used specific “block type” assumptions and obtained a noticeable gain. Complexity reduction is also obtained while using a tree type dependence between variables $x_{r1}, x_{r2}, \dots, x_{rN}$ (Bai et al. 2009; Raudys, Saudargiene 2001; Raudys 2001).

3. Multilayer feed-forward system for large-scale portfolio design

Feed-forward large-scale portfolio construction system is composed of several parts. The steps are as follows:

- In order to obtain diversifications and reduce a number of inputs we form trading groups that calculate averages of a large number of similar trading robots. For this purpose, for each single time intervals (say 100 or 400 days time interval) we cluster ATSS’ by the correlation of their simulated trading track record – profit and loss series. We form a number of clusters in such a way that the correlations inside the clusters are high, however, correlations between the clusters are much smaller.
- Then we can join trading systems inside each single cluster in order to make a trading agent. For this purpose, the $1/N$ portfolio is close to optimal (Raudys 2013).
- After reduction of the number of factors (inputs) we can use mean-variance principle based portfolio design, i.e. perform classical Markowitz rule between the outputs of the first order agents.

- To increase the diversity we: 1) use distinct regularization values; 2) carry out a several clustering procedures based of time series of different length. In such a way we obtain a great number of trading modules (see schema in Fig. 3).
- To select the best modules for final portfolio weights calculation we use specially developed *cost sensitive multilayer perceptron*.
- We use selected agents to produce future portfolio as an average of modules' weights recognized as belonging to “the elite (the fourth) class”.

In the next section we will explain the steps in more detail.

3.1. History length selection

The data used for analysis cover prior-to-crisis and crisis time period, characterized by multitude of sudden changes. In preceding analysis (Raudys, S., Raudys, A. 2011) it was found that 300–600 days of data history produce the highest Sharpe ratio. Calculations show that a small (below 100) number of inputs should be used in such situations (Raudys 2013). Consequently, we have to use either non-trainable 1/N Portfolio rule, or reduce the number of inputs severely and use the mean-variance approach. Both ways have their positive and negative aspects. Thus, we need to find a compromise solution.

At first, we considered conditions where 1/N portfolio can be optimal and found that if

- the sum of the automated trading robot outputs are normally distributed;
- outputs of N_b robots are equally correlated, i.e. $\rho_{ij} = \rho$;
- the mean returns of the robots are equal,

then the benchmark 1/ N_b portfolio rule is the optimal solution. This conclusion follows from a simple matrix algebra analysis of covariance matrix of the form

$$\mathbf{K} = \mathbf{I} \times (1 - \rho) + \mathbf{1} \mathbf{1}^T \times \rho,$$

when matrix \mathbf{K} is inserted instead of \mathbf{S} into Equation (4).

Modern world is affected by the fast technological, political and economical changes, especially in prior-to-crisis and crisis period. Thus, financial situations very often are varying and unpredictable. Therefore, the lengthy time series are often unsuitable for precise portfolio calculation. One needs to employ shorter training histories (Raudys, Zliobaite 2005; Raudys, Mitasiunas 2007; Raudys 2013). In our analysis of diverse trading strategies we face data where in $\frac{3}{4}$ of days the trading robots refuse to operate. Therefore, actual (effective) lengths of time series are much shorter and training sample size becomes a very important issue.

3.2. Clustering of the trading robots

One of the principal ideas proposed in this paper is that having a large number of ATS (trading robots) allows us to group/cluster most similar ATS together and create almost uncorrelated groups. We group PnL series into clusters by their correlation using k-means algorithm and perform 1/N portfolios in each of them. The k-means algorithm is known and popular algorithm for constructing groups of similar items described in many books and realized in

many data mining software. In our portfolio design schema, we perform cluster analysis of $N \times L$ data according to correlations between the N time series of length L . Here we use absolute values $|1 - \text{correlation}_{ij}|$ as the similarity measure between i -th and j -th time series. Assuming that the robots inside single group are similar we can use mean values of their outputs in such a way realizing $1/N$ B portfolio rules serving as R first order (expert) agents. Having a small number of expert agents we can use mean-variance approach to design more complex agent (module) where R weights would be calculated according to Equation (4). To improve small sample properties of the covariance matrix we use regularized covariance matrix:

$$S_{\text{reg1}} = S \times (1 - \tau) + D \times \tau, \tag{5}$$

where: D is a diagonal $N \times N$ matrix found from matrix S ; and τ is regularization parameter ($0 \leq \tau \leq 1$).

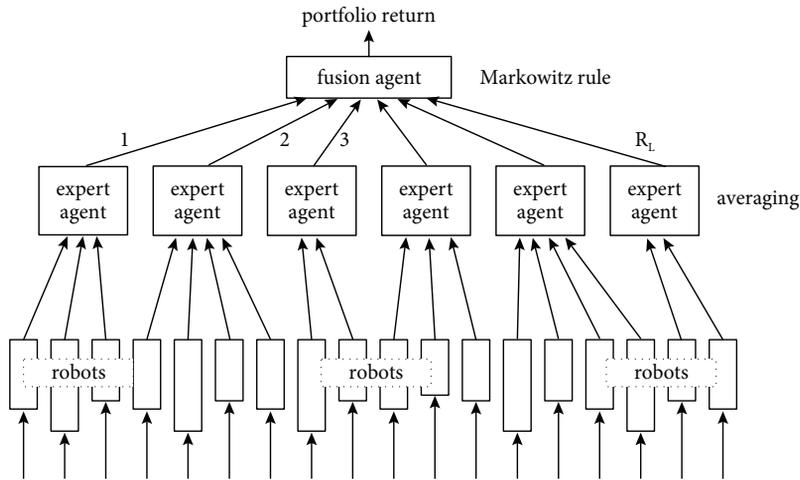


Fig. 3. Feed-forward flow of information in single trading module of decision-making system

Two levels decision-making schema or R non-trainable (expert) and one trainable agent we call *feed-forward trading module* (FFM), see Figure 3.

3.3. Final feed-forward system design

A novelty of the present paper is an introduction of a vast number of feed-forward modules depicted in Figure 3. The modules differ in learning set sizes, L , used to perform clustering of the data, regularization parameter, t , and randomly selection of subsets of expert agents. In our experiments reported below, in each of a *walk-forward step* we considered four learning set sizes $L_1 = 100$, $L_2 = 200$, 300 and 400 days prior to 100 days validation period and clustered ATS into $R = 25$ groups. After averaging, we formed 100 diverse expert agents. To increase a diversity, from 100 agents we formed 80 semi-randomly selected groups of $R_L = 60$ expert agents to be joined into 120 types of higher order agents. Each of the latter agents were made

by the mean-variance approach using one of four *a priori* selected values of covariance matrix regularization ($t = 0.4, 0.6, 0.7$ or 0.8). Altogether, we used 480 diverse feed-forward modules in the experiments.

3.4. Selection of “elite modules” for the final portfolio calculation

Each single module depicted in Figure 3 gives a set of portfolio weights. Some modules perform better in some time interval, other modules are preferable in another time interval. To adapt to changes constantly, every 10 days we performed module selection for the final portfolio design. Each time we divided 480 modules into $K = 4$ equal groups (120 modules in each) according to their mean return values during the last 20 days. Then we used specific parameters of the modules: 1) a number of times when each of 100 first order agent was employed in the module; 2) regularization parameter; and 3) clustering data interval size.

For classification we used four class multilayer perceptron training pair-wise misclassification cost specific loss function, where in the output layer instead of minimizing the class specific weights w_1, w_2, \dots, w_K , one minimizes differences of two weight vectors, $w_j - w_h$ (Raudys, S., Raudys, A. 2010):

$$Loss_{\text{novel}} = \sum_{h=1}^K q_h / N_h \sum_{j=1}^K C_{hj} \sum_{s=1}^{N_h} (f(w_j - w_h)^T y_s^{(h)})^2. \quad (6)$$

We use a $K \times K$ – dimensional misclassification cost matrix $C_{\text{cost}} = ((C_{hj}))$ to control pair-wise misclassification cost.

4. Simulation study

In the simulation study we performed a set of experiments with three different datasets. The proposed feed-forward methodology showed better results than benchmark method. Below we will describe experiments in more detail. The model testing can be done in 3 ways. Firstly, one creates a model, calibrates its parameters and tests performance/quality on the same data set. This type of testing is widely criticized because complex models can adapt to the training data and generate optimistically biased results. Secondly, one calibrates the model on one set and validates it on the unseen data. This method is better but also unreliable as having one satisfactory out of sample result which can be a matter of luck. This approach is popular but limits us to a small amount of unseen data. The third alternative: one is repeating the experiment with multitude of pairs of the data subsets. One subset is used for model calibration and another one is used for performance evaluation. In time series analysis it is called a walk-forward approach. Here, we train the model on one set and test it on the small period of future data. Next, we shift the training period by period x and we shift the testing period by the same time x into the future. This procedure is repeated until there is no data to shift our training and testing data periods. Walk-forward analysis is gaining popularity. It is a time consuming process but allows viewing potential results in out of sample with longer time periods.

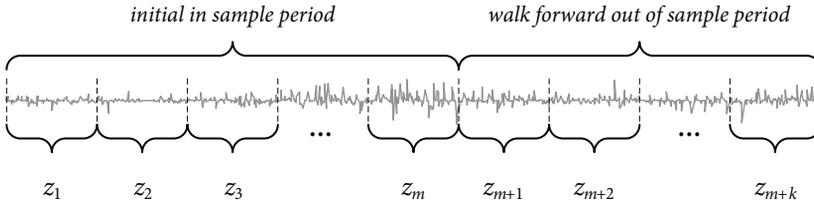


Fig. 4. Walk-forward testing, z_{m+1} is the first out-of-sample period and z_{m+k} and is the last

In this study, we organised data into $k = 16$ time intervals z_i of 100 working days. Initially, we create a portfolio using $z_1 \dots z_m$ intervals and test it on z_{m+1} interval. In the next step, we create a portfolio using $z_1 \dots z_{m+1}$ range and test on z_{m+2} . The process is repeated until we reach z_{m+k} . This is illustrated in Figure 4. So we have totally k out of sample periods that we can concatenate and get one long out of sample period. In total we have $m + k$ periods. Thus, in experiments reported below, for testing we used $16 \times 100 = 1600$ days data.

4.1. Benchmark methods

In small scale problem we could use classic Markowitz approach for portfolio construction. Because of the size of the problem we cannot use classic Markowitz as a benchmark method. Markowitz is not capable of handling large-scale, high-dimensionality data. We are forced to use simple $1/N$ rule as a benchmark instead. Here, every possible portfolio member is included into account.

This rule is sometimes regarded as equally weighted rule. In some research works it was noted that this rule can be rather good in many practical portfolios (DeMiguel *et al.* 2009). We considered two versions of $1/N$ rule. In the first one, all ATs are used. In its modification, only agents with average above zero profits during the latest 400 trading days were selected.

In spite of the known ways to reduce complexity “there are still many miles to go” before the gains promised by optimal portfolio choice can actually be realized out of sample (Kan, Zhou 2007; DeMiguel *et al.* 2007). One of the reasons of a mild success relies on the incorrect normality assumption inherently incorporated into the standard mean-variance framework. This factor is often observed if a small number of assets and records are measured. Therefore, simple fixed (non-trainable) benchmark portfolio trading rule ($1/N$ – equally weighted portfolio) with ($w_{r1} = w_{r2} = \dots = w_{rN} = 1/N$) is suggested as a benchmark method (DeMiguel *et al.* 2007).

4.2. Portfolio construction experiments

In an attempt to choose the best model for the final portfolio calculation, we split 480 systems into four equal (120 models in each of them) *pattern classes* according to the last period’s 10 days mean return values. The most successful models (the 4th class) were used in the final Portfolio weights calculation for *subsequent 10 validation days*. To “recognize” the 4th class models we used the misclassification cost-sensitive multilayer perceptron with 20 inputs,

two hidden units and four outputs. Such perceptron allowed realizing *non-linear decision boundary*. Attributes for recognition of the best models were: a) 100 quantities of expert agents in each model; b) a generalized parameter that characterized learning set sizes of the given model; and c) regularization parameter, τ .

Totally, we had 102 attributes. Learning set size used to train perceptrons was 480. To improve small sample properties of classification rule we performed a singular value decomposition of 480×102 dimensional data and used the first 20 principal components for classification. The multilayer perceptron was trained starting from very small random initial weights using a cost sensitive algorithm aimed to minimize the sum pair-wise costs of misclassifications (Raudys, S., Raudys, A. 2010). In some of experiments the pair-wise misclassification costs were calculated according to values of differences between the average returns in the 4 pattern classes. The best results, however, were obtained when our costs matrix prevented allocation of vectors of the first class (most unsuccessful) to the fourth class. In Table 2 we present misclassification cost matrix used in the experiments. Here, allocation of the worst models (the 1st class) to the 4th class is predominantly penalized.

Table 2. The 4×4 dimensional matrix, C_{cost} , of pair-wise misclassification costs

Class	1	2	3	4
1	0	2	4	20
2	1	0	1	1
3	1	1	0	1
4	1	1	1	1

Table 3. A number of allocations of 120 vectors (modules) matrix in a single 20 days period training session. Diagonal values represent correct classifications and other – a number of misclassifications (on the right)

Class	1	2	3	4
1	82	37	1	0
2	21	44	47	8
3	17	15	69	19
4	4	5	15	96

In Table 3 and Figure 5 we can see that the 1st and 4th classes are separated pretty well. None from the first class was assigned to the class 4. This is what we tried to achieve using cost matrix and indicating very high cost for such assignment. This result was achieved using perceptron with 2 hidden layers that takes into account cost matrix during training. Figure 5 visualises 480 agents in 2D two hidden unit space.

We can see in the figure that class one (black dots) is not mixed with class 4 (green dots). In 16 walk-forward shifts of the training and the test data sets we designed 480 trading modules 16 times. Selection of the elite (the 4th class) modules were performed after each 10 trading days. So, the evaluation of portfolio performance was executed 160 times.

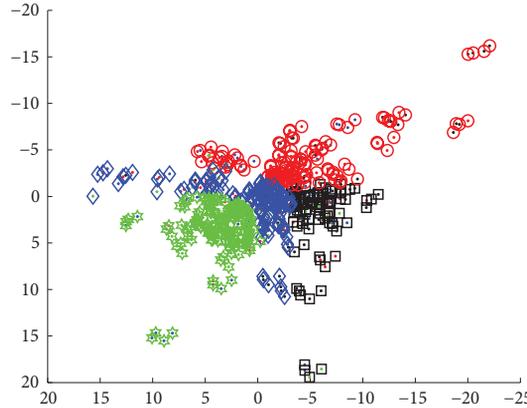


Fig. 5. Classification results into 4 pattern classes (first class is black; green – the “elite” class)

To avoid over-adaptation to experimental material, the parameters of multistage portfolio weights calculation schema were found while experimenting with the 11,730 trading agent data recorded in the period from 2002 to 2010. The evaluation of the method’s performance was executed with another two data sets formed from diverse trading robots collections selected from 2002 to June 2012 and from 2003 to December 2012 data archives. Below we present out-of-sample results, variation of the Sharpe ratio during the last 6 years.

By employing proposed approach where agents differ in the training history length, allowed us to improve portfolio Sharpe ratio from equally weighted 5.23 to a new system’s 7.59. Both benchmark methods were approximately equally effective (Fig. 6. left panel). The 1.5 times improvement is very stable and statistically significant and in almost all out of sample experiments produced better Sharpe ratio (Fig. 6).

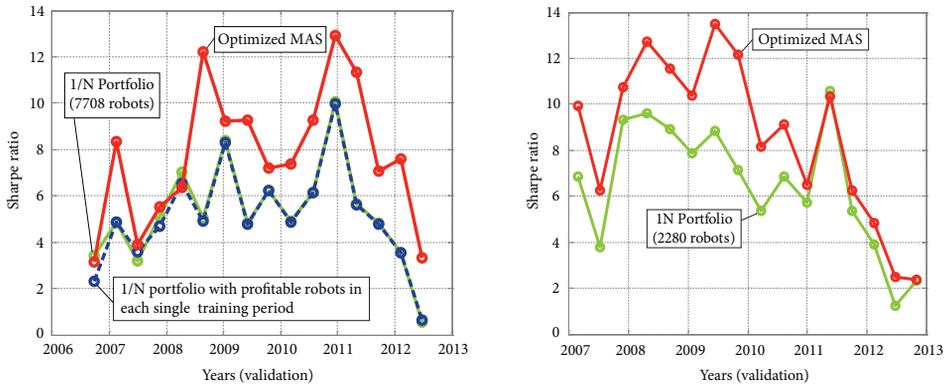


Fig. 6. Variation of out-of-sample Sharpe ratio evaluated in six years period with two diverse data sets of trading robots (left panel – $p = 7,708$ dataset, right panel – $p = 3,133$ dataset)

Conclusions

This paper presents a portfolio construction method inspired by sustainable economy principles, so that large tasks are divided into smaller ones, solved and later composed into a final solution. The theoretical justification for this novel solution is based on multivariate statistical analysis of multidimensional investment tasks, particularly on relations between data size, algorithm complexity and portfolio efficacy. Validation of the feed-forward decision-making system was performed on large-scale financial data sets, taking into consideration thousands of ATS during the last ten years.

Having a large number of portfolio candidates, we move towards the Gaussian distribution of portfolio returns. In such a situation, one can use the mean-variance approach. In the case of thousands of ATS, however, the small sample size problem arises. In high dimensional situations the employment of typical solutions, such as the Markowitz optimization principle, becomes ineffective. Thus, it is necessary to develop additional tools for reducing dimensionality with minimal loss of useful information. Our novel multilevel feed-forward decision-making schema comprises the following procedures:

- 1) clustering is used for dimensionality reduction and generating the first order trading agents by using the non-trainable $1/N$ portfolio design rule;
- 2) mean/variance optimization employed to taking into account correlations between the outputs of the first order trading agents;
- 3) cost-sensitive multi-category classification applied to select the group of the best trading modules;
- 4) final decision making based on the non-trainable $1/N$ portfolio rule.

The above sequence of procedures is based on sound theoretical considerations and is explained in more detail in Raudys, S., Raudys, A. (2011) and Raudys (2013). In the first procedure, we obtain a gain due to the theoretically-based knowledge that for correlated agents with similar mean returns, the non-trainable $1/N$ portfolio rule is well founded. In the second procedure, we obtain a gain because we have a relatively small number of first order trading agents and regularize the covariance matrix while developing the trading modules. In the third procedure, we have a gain in view of the fact that we are selecting the most promising trading modules by means of a special multilayer perceptron capable of taking into account the pair-wise costs of misclassification. In the fourth procedure, we expect a gain due to employing the $1/N$ rule. The gain can result from the fact that after the use of randomization, the performances of all modules allocated to the 4th class and correlations between the module outputs should not differ notably. Theory shows that in such situations the non-trainable $1/N$ rule becomes close to optimal.

In our two large-scale empirical performance evaluations we demonstrated the superiority of the novel method over the benchmark methods in 16 out-of-sample periods. From theoretical and empirical analysis it was clear that sample size issues are of great importance in portfolio construction: shorter time series are beneficial to out-of-sample portfolios when environments are undergoing frequent change. This can be useful during a crisis period, in which the environment is changing more rapidly than would usually be the case. Therefore, for portfolio construction, shorter histories have to be used (Raudys, S., Raudys, A. 2011).

The new trading system portfolio methodology has a theoretical basis and has been verified empirically using the large financial data sets. It shows promising results, although, it can be improved undoubtedly. One possible way would be to start applying evolutionary and/or memetic algorithms (Krasnogor, Smith 2005) instead of agent/module selection. Recent approaches aimed at pre-processing truly high-dimensional input data to low-dimensional representations combined with regularization (Stuhlsatz *et al.* 2012; Zafeiriou *et al.* 2012) can facilitate enhanced trading agents and design of modules.

The proposed multi-layer feed-forward portfolio construction system with the selection of the best agents and modules for each time interval allowed reducing the number of incorrect decisions and increasing Sharpe ratio. We believe that the use of adaptive multistage feed-forward systems is suitable not only for financial portfolio modelling. In sustainable ecology, sustainable economy and sustainable society analysis tasks, a multitude of factors/agents (smaller elements of the large model) influence the final decision. The similarity between such tasks and the large-scale portfolio design strategy suggests that the newly developed methodology is worth applying to wider areas of research. We need to seek alternative problems that can provide sufficient data and which are similar in nature to the modelling problems discussed.

Acknowledgments

This work was supported by the Research Council of Lithuania under Grants MIP-043/2011 and MIP-018/2012.

References

- Aldridge, I. 2010. *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. Hoboken, New Jersey: John Wiley & Sons. 354 p.
- Araújo, C.; de Castro, P. 2011. Towards automated trading based on fundamentalist and technical data, *Advances in Artificial Intelligence – SBIA 2010*, 112–121.
- Bai, Z.; Liu, H.; Wong, W.-K. 2009. On the Markowitz mean-variance analysis of self-financing portfolios, *Risk and Decision Analysis* 1: 35–42.
- Board on Agriculture and Natural Resources. 2012. *Sustainable development of algal biofuels* [online], [cited 17 November 2012]. Available from Internet: http://www.nap.edu/catalog.php?record_id=13437. Washington DC, USA: The National Academic Press.
- Bookstaber, R. 2009. *Risk from high frequency and algorithmic trading not as big as many think* [online], [cited 30 August 2009]. Available from Internet: <http://seekingalpha.com/article/158962-risk-from-high-frequency-and-algorithmic-trading-not-as-big-as-many-think>
- Brodie, J.; Daubechies, I.; De Mol, C.; Giannone, D.; Loris, I. 2009. Sparse and stable Markowitz portfolios, *PNAS (Proceedings of the National Academy of Sciences of the United States of America)* 106(30): 12267–12272. <http://dx.doi.org/10.1073/pnas.0904287106>
- Chan, E. P. 2008. *Quantitative trading: how to build your own algorithmic trading business*. Hoboken, New Jersey: John Wiley & Sons. 204 p.
- Cochrane, K. L.; Andrew, N. L.; Parma, A. M. 2011. Primary fisheries management: a minimum requirement for provision of sustainable human benefits in small-scale fisheries, *Fish and Fisheries* 12(3): 275–288. <http://dx.doi.org/10.1111/j.1467-2979.2010.00392.x>

- Cura, T. 2009. Particle swarm optimization approach to portfolio optimization, *Nonlinear Analysis: Real World Applications* 10(4): 2396–2406. <http://dx.doi.org/10.1016/j.nonrwa.2008.04.023>
- DeMiguel, V.; Garlappi, L.; Uppal, R. 2007. Optimal versus naive diversification: how inefficient is the 1/N portfolio strategy?, *Review of Financial Studies* 22(5): 1915–1953. <http://dx.doi.org/10.1093/rfs/hhm075>
- DeMiguel, V.; Garlappi, L.; Nogales, F. J.; Uppal, R. 2009. A generalized approach to portfolio optimization: improving performance by constraining portfolio norms, *Management Science* 55(5): 798–812. <http://dx.doi.org/10.1287/mnsc.1080.0986>
- Dempster, M. A. H.; Jones, C. M. 2001. A real-time adaptive trading system using genetic programming, *Quantitative Finance* 1(4): 397–413. <http://dx.doi.org/10.1088/1469-7688/1/4/301>
- Hung, K. K.; Cheung, C. C.; Xu, L. 2000. New Sharpe-ratio-related methods for portfolio selection, in *Proc. of the IEEE/IAFE/INFROMS Conference on Computational Intelligence for Financial Engineering*, 26–28 March, 2000, New York, 34–37.
- Jeucken, M. 2001. *Sustainable finance and banking: the financial sector and the future of the planet: people-planet-profit in the Financial Sector*. Guilford: Routledge. 320 p.
- Kan, R.; Zhou, G. 2007. Optimal portfolio choice with parameter uncertainty, *Journal of Financial and Quantitative Analysis* 42(3): 621–656. <http://dx.doi.org/10.1017/S0022109000004129>
- Krasnogor, N.; Smith, J. 2005. A tutorial for competent memetic algorithms: model, taxonomy, and design issues, *IEEE Transactions on Evolutionary Computation* 9(5): 474–488. <http://dx.doi.org/10.1109/TEVC.2005.850260>
- Markowitz, H. 1952. Portfolio selection, *The Journal of Finance* 7(1): 77–91.
- McCormick, R. 2012. Towards a more sustainable finance system, part 2: creating an effective civil society response to the crisis, *Law and Financial Markets Review* 6(3): 200–207. <http://dx.doi.org/10.5235/175214412800650527>
- Moodyand, J.; Lizhong, W. 1997. Optimization of trading systems and portfolios, in *Proceedings of the IEEE/IAFE Computational Intelligence for Financial Engineering (CIFEr)*, 23–25 March, 1997, New York, 300–307.
- Perold, A. 1984. Large-scale portfolio optimization, *Management Science* 30(10): 1143–1160. <http://dx.doi.org/10.1287/mnsc.30.10.1143>
- Raudys, S. 2001. *Statistical and neural classifiers: an integrated approach to design*. New York: Springer. 289 p. <http://dx.doi.org/10.1007/978-1-4471-0359-2>
- Raudys, S. 2013. Portfolio of automated trading systems: complexity and learning set size issues, *IEEE Transactions on Neural Networks and Learning Systems* 24(3): 448–459. <http://dx.doi.org/10.1109/TNNLS.2012.2230405>
- Raudys, S.; Young, A. 2004. Results in statistical discriminant analysis: a review of the former Soviet Union literature, *Journal of Multivariate Analysis* 89(1): 1–35. [http://dx.doi.org/10.1016/S0047-259X\(02\)00021-0](http://dx.doi.org/10.1016/S0047-259X(02)00021-0)
- Raudys, S.; Mitasiunas, A. 2007. Multi-agent system approach to react to sudden environmental changes, *Lecture Notes in Artificial Intelligence* 4571: 810–823.
- Raudys, S.; Raudys, A. 2010. Pair-wise costs in multi-class perceptrons, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32: 1324–1328. <http://dx.doi.org/10.1109/TPAMI.2010.72>
- Raudys, S.; Raudys, A. 2011. High frequency trading portfolio optimization: integration of financial and human factors, in *Proc. 11th International Conference on Intelligent Systems Design and Applications (ISDA)*, 22–24 November, 2011, Córdoba, Spain, 696–701.
- Raudys, S.; Raudys, A. 2012. Three decision making levels in portfolio management, in *IEEE Conference on Computational Intelligence for Financial Engineering and Economics*, 29–30 March, 2012, New York, 1–8.

- Raudys, S.; Saudargiene, A. 2001. First order tree-type dependence between variables and classification performance, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(2): 1324–1328. <http://dx.doi.org/10.1109/34.908975>
- Raudys, S.; Zliobaite, I. 2005. Prediction of commodity prices in rapidly changing environments, *Lecture Notes in Computer Science* 3686: 154–163.
- Smeureanu, I.; Ruxanda, G.; Diosteanu, A.; Delcea, C.; Cotfas, L. A. 2012. Intelligent agents and risk based model for supply chain management, *Technological and Economic Development of Economy* 18(3): 452–469. <http://dx.doi.org/10.3846/20294913.2012.702696>
- Stuhlsatz, A.; Lippel, J.; Zielke, T. 2012. Feature extraction with deep neural networks by a generalized discriminant analysis, *IEEE Transactions on Neural Networks and Learning Systems* 23(4): 596–608. <http://dx.doi.org/10.1109/TNNLS.2012.2183645>
- Zafeiriou, S.; Tzimiropoulos, G.; Petrou, M.; Stathaki, T. 2012. Regularized kernel discriminant analysis with a robust kernel for face recognition and verification, *IEEE Transactions on Neural Networks and Learning Systems* 23(3): 526–534. <http://dx.doi.org/10.1109/TNNLS.2011.2182058>

Šarūnas RAUDYS. Doctor Habil, Professor. He obtained the Master's degree and the PhD degrees in Computer Science from Kaunas University of Technology, and USSR Doctor of Science (Habil) degree from Riga Institute of Electronics and Computer Science in 1978. Presently, he is a Senior Researcher in Faculty of Mathematics and Informatics, Vilnius University. Research interests: multivariate analysis, statistical pattern recognition, data mining, artificial neural networks, deep learning, evolvable multi-agent systems, artificial economics, and artificial life.

Aistis RAUDYS received his PhD from the Institute of Mathematics and Informatics, Lithuania in a field of Feature Extraction from Multidimensional Data. Currently, he works as a Senior Research Fellow at Vilnius University Faculty of Mathematics and Informatics where he teaches Algorithmic Trading Technologies. Previously, he worked as a Researcher and also as a Software Developer in various software companies. He collaborated with a number of top tier banks including Deutsche Bank, Société Générale and BNP Paribas. He is the author of 21 publications and scientific works. His research interests are in machine learning for financial engineering and automated trading.

Židrina PABARŠKAITĖ obtained PhD from Vilnius Gediminas Technical University, Lithuania in 2009. She is the author of 7 research articles. She worked as a Lecturer and Data Analyst in the past. Currently, she is working as Postdoctoral Research Fellow at Kaunas University of Technology in the field of Multivariate Data Analysis. Her research object was web log mining process: enhancements of web log data preparation process, application of different methods and algorithms to the web log data analysis and results presentation.