

Prediction of composite indicators using combined method of extreme learning machine and locally weighted regression

Jurga Rukšėnaitė^a, Pranas Vaitkus^b

^aVilnius Gediminas Technical University
Saulėtekio ave. 11, LT-10223 Vilnius, Lithuania
jurga.ruksenaite@vgtu.lt

^bVilnius University
Naugarduko str. 24, LT-03225 Vilnius, Lithuania
vaitkuspranas@gmail.com

Received: 8 May 2011 / **Revised:** 17 February 2012 / **Published online:** 21 May 2012

Abstract. In this paper, a method of artificial neural networks (NN) is proposed as an alternative tool for the one-step-ahead prediction of composite indicators (CIs) of Lithuania's economy. CI is composed of widely used social and economic indicators. The NN is applied for forecasting CI during the financial crisis and later periods (2008–2010) on the basis of data of earlier years (1998–2007). In this work, the Extreme Learning Machine (ELM) algorithm is combined with locally weighted regression. The analysis shows that the prediction error of a testing sample is statistically smaller compared to Levenberg–Marquardt or ELM methods.

Keywords: composite indicators, neural networks, ELM, locally weighted regression.

1 Introduction

Recently more and more artificial indicators have been established in global academic and research fields which reflect the development of a country's economy in various aspects. Composite indicators (CIs), which compare country performance, are a useful tool in setting policy priorities. Together with the widely used statistical indicators such as gross domestic product (GDP), price indices, the unemployment rate, and others, various artificial indicators are presented, e.g. Global Competitiveness Index (World Economic Forum), Index of Economic Freedom (Heritage Foundation and Wall Street Journal), Summary Innovation Index (European Commission), and others. The use of CI's around the world is growing year after year [1]. This may be influenced by the fact that more and more researches appear which sustain a proposition that GDP does not always reflect the real development of a country, quality of life, development of technologies, and other [2]. Also, when economic conditions are unstable, there is a need to evaluate and analyze additional indexes as separate statistical indicators do not always show the real economic

situation. A comprehensive analysis of artificial indices and statistical data may give a more general view on the current situation.

This research work is the further analysis of the construction of the CIs of Lithuania's economic development [3]. In the previous paper, a CI which shows the trends in Lithuania's economy was constructed. Different socioeconomic indicators, which are widely used and notably correlate with GDP, were used for the composition of CIs. The dynamics of CIs was compared to the trends in the main economic indicator – GDP. The analysis showed that the CI might be used as an additional tool for the investigation of the country's economic development. The impact of weights on the final results of CIs was studied.

Short-term forecasting of macroeconomic time series is one of the main stages of the analysis of economic trends in the country. Hence, the forecast of such an index (CI) allows us to make an additional economic analysis and earlier to detect possible changes in GDP trends.

Several institutions are publishing official forecasts of various macroeconomic indicators in Lithuania. E.g. at Statistics Lithuania, traditional econometric models are used for the short-term forecasting of Lithuania's GDP. These methods are based on regression analysis, where additional economic indicators – regressors – are used for the prediction of the single value added of a particular activity [4]. The Bank of Lithuania regularly does research related to the macroeconomic analysis and forecasts using the structured macroeconomic model. The model-based projections are then updated taking into account expert judgement with respect to structural changes and the latest available information about the forthcoming economic shocks [5]. The recently proposed methods for the forecasting of GDP are bridge models (monthly data are used) and dynamic factor models [6]. The authors [6] noted that the models that are appropriate for the euro area countries are not always suitable for the new Member States. For Lithuania, for instance, the results obtained in [6] are difficult to interpret.

It should be noted that often researches are modeling economic indicators using linear models [7] which are based on parametric techniques and rather strict conditions on the data distribution. If the conditions are fulfilled, the respective model may be used; otherwise, we cannot be sure of the quality of the statistical results obtained. One can apply the standard time series models to the process if it can be expressed by means of stationary sequences [7]. This assumption is likely to be violated during crisis periods.

In the last decades, artificial neural networks (NN) have become one of the most powerful tools which enable the analysis, simulation and prediction trends in large systems in various fields [8]. The NN theory is based on Takens' theorem [9], which claims that it is possible to rebuild the dynamics of the process using correct time lags. The main difference, compared to the methods discussed, is that the NN models are non-linear models which are, by definition, more powerful since they give more possibilities in the choice of the input-output relation [10]. The NN have the universal approximation property: under mild conditions on the data, they can fit any data set with an arbitrary high precision, provided that there is a sufficient number of parameters in the model. However, when there are too many parameters (compared to the number of the data available), the overfitting phenomenon appears [10].

Often researches choose the NN tool as an alternative method for the analysis and forecasting of the main economic indicators such as GDP or inflation. In this paper, we refer only to the authors who obtained significant results using the NN in economic fields.

Tkacz (Bank of Canada) proposed the NN which enables forecasting the growth of Canadian GDP. The analysis showed that the NN yield statistically lower errors of forecasts for the year-over-year growth rate of real GDP relative to linear and univariate models [11]. McNelis and McAdams (European Central Bank) conducted an investigation into the forecasts of inflation using the NN for the USA, Japan and the euro area. The authors showed that, in particular cases, the errors of forecasts are significantly smaller in comparison with the linear models [12].

Hence, in this paper, the NN methods were chosen for the prediction of CIs according to the theoretical framework and statistically significant results of foreign researches which forecasted different macroeconomic indicators using this approach. The first research concerning the prediction of the growth of Lithuania's economy using the NN was proposed by Jakaitienė and Tamošiūnaitė [13]. They analyzed data of quarterly periodicity 1996–2002. The authors suggested to use the NN as an additional tool for the forecasting of GDP growth. Still they recommended that higher accuracy of prediction may be obtained with a larger set of observations (longer time series).

Practically, there are no comprehensive researches concerning the prediction (using the NN) of trends in Lithuania's economy after financial-economic crisis. This can be explained by the fact that most of the official socioeconomic macro data are short time series starting from 1998 [4]. For example, if data is of quarterly periodicity and the period of 1998–2010 is analyzed, there are only 52 observations. This fact may complicate the implementation of the NN architecture which usually requires large data sets for fitting.

On the other hand, there are papers published where the NN are used for the analysis of short time series. Zhang and Kline constructed the NN and analyzed the characteristics of various economic time series of quarterly periodicity, where the size of the smallest samples varied from 16 to 28 observations [14].

The objective of the paper is to analyze the one-step-ahead predictions of the CI of Lithuania's economy using NN methods. The essential questions were highlighted:

- (i) To analyze the accuracy of predictions using different data transformation methods at the initial stage and the NN architecture (the number of neurons of a hidden layer).
- (ii) To analyze errors using different methods of training and prediction of the NN.
- (iii) To verify whether the combined NN method proposed increases the CI forecasting accuracy.

The novelty of the research is the proposed NN method: we put together the algorithm of the ELM (for the training of NN) and the locally weighted regression (for prediction). To our knowledge there is no comprehensive studies analyzing the prediction accuracy of macroeconomic indicators by using this combined algorithm. Also in this paper we discuss the question concerning the small samples problem as NN are often used for the large samples.

The structure of the paper is as follows. The methodology of the construction of the CI is described in Section 2. Section 3 presents methods chosen for the training and testing of

NN. Section 4 gives the process of the implementation and characteristics of the models of NN. A practical case concerning the prediction of CI using the NN is described in Section 5. Section 6 gives concluding remarks.

2 Construction of the CI

In this section, the process of construction of the CI is described: the selection of preliminary data, data disaggregation, the selection of weights, and the aggregation of the CI.

We have defined the CI as an additional tool for country's economic analysis. The CI is defined as follows

$$CI(t, \varphi) = \sum_i X_i(t) \varphi_i, \quad i = 1, \dots, m, \quad (1)$$

where $t = 1, \dots, T$ is time, $\mathbf{X}(t) = (X_1(t), \dots, X_m(t))' \in \mathbf{R}^m$ is a vector of certain economic indicators, e.g. various statistics of business, technologies, and other. $\varphi \in \mathbf{R}^m$ is a vector of weights of the different indicators.

In this research we have used the methodology of construction the CI of Lithuania's economy proposed in [3]. We will shortly describe the main features of the methodology in order to present the theoretical background. In this paper, we have extended the period of analysis (1998–2010) and constructed the artificial index with higher frequency, that is, monthly periodicity (the previous one was of quarterly periodicity, the period of 1998–2009).

We have selected the data set \mathbf{X} for the construction of the CI of monthly periodicity. Statistical indicators (the same variables as in the mentioned paper, $m = 28$) are from the following fields/subfields: population and social statistics (2 indicators), industry (4 ind.), construction (4 ind.), domestic trade (6 ind.), foreign trade (4 ind.), services (3 ind.), price indexes (5 ind.).

The process of the construction of the CI can be described in the following steps.

Data disaggregation

Some statistical indicators X_i , such as export or industrial production, are officially published on a monthly and quarterly basis. But we also use time series (construction, services data) which do not have statistically fixed information of monthly periodicity (available only on a quarterly basis). Here we have to solve a mathematical problem concerning the data disaggregation from low frequency (LF) (with quarterly periodicity), to higher frequency (HF) (with monthly periodicity). Two methods of time series disaggregation from LF to HF were chosen:

- (i) The method of Barbone, Bodo and Visco [15] was used when there was a variable which strongly correlated with the series analyzed. The residuals of the regression model are described using an autoregressive model AR(1), and the parameters are evaluated using the minimization of the sum of squared residuals.
- (ii) Denton method [16] was used when there was no variable with similar trends and structure. It disaggregates the time series using only its characteristics.

Next, we will consider the data set $X_i(t)$ of monthly periodicity, $t = 1, \dots, T$, $i = 1, \dots, m$.

Preliminary analysis of the data

Economic time series are often affected by seasonal fluctuations, which may be observed each year at roughly the same time (in our case, this is the same month). Such statistical data may give misleading information on the real dynamics of the time series. Therefore seasonal adjustment (TRAMO/SEATS method [17]) was applied to the all time series.

The data corrected for outliers (additive outliers, level shifts and transitory changes): the number of outliers of every time series should not exceed 5 per cent. Then the all selected indicators X_i were standardized (mean $\mu_i = 0$ and variance $\sigma_i^2 = 1$). Exactly these indicators are used in (1).

Selection of weights and aggregation

In the previous work indicators' weights φ were evaluated using factor analysis methods [18]. We chose the same set of φ that had been evaluated using the factor rotation method Varimax [19] and improved using the Nikoletti [20] method. In general, this method gives the highest weights to those indicators that strongly correlate with the corresponding factor. It is supposed that the chosen weights φ in (1) do not depend on the time t .

3 Methods of training and testing of the NN

In this section, we describe the methods that are used for training and testing the NN and the one-step-ahead prediction of the CI: Levenberg–Marquardt, Extreme Learning Machine (ELM) and the proposed method combined of ELM and locally weighted regression.

Let us describe the approximation problem of the single hidden layer of the feed-forward NN (SLFNs). For T arbitrary distinct samples $(\mathbf{z}_t, \mathbf{y}_t)$, $t = 1, \dots, T$, where $\mathbf{z}_t = [z_{t1}, \dots, z_{tn}]' \in \mathbf{R}^n$ and $\mathbf{y}_t = [y_{t1}, \dots, y_{tk}]' \in \mathbf{R}^k$, standard SLFNs with N hidden neurons and activation functions

$$\mathbf{g}_\ell(\mathbf{z}) = \mathbf{g}^\circ(\mathbf{w}_\ell \cdot \mathbf{z} + b_\ell), \quad \ell = 1, \dots, N, \quad (2)$$

are mathematically modeled as

$$\tilde{\mathbf{y}}_t := \sum_{\ell=1}^N \alpha_\ell \cdot \mathbf{g}_\ell(\mathbf{z}_t) \approx \mathbf{y}_t, \quad t = 1, \dots, T, \quad (3)$$

where $\mathbf{w}_\ell = [w_{\ell 1}, \dots, w_{\ell n}]'$ is the vector of weights connecting the ℓ th hidden neuron and the input neurons, $\alpha_\ell = [\alpha_{\ell 1}, \dots, \alpha_{\ell k}]'$ is the vector of weights connecting the ℓ th hidden neuron and the output neurons, and b_ℓ is the threshold of the ℓ th hidden neuron. T is the number of training samples and $\mathbf{g}^\circ : \mathbf{R} \rightarrow \mathbf{R}^k$ is a basic activation function. Symbol \prime denotes the matrix transposition.

Classical training method for backpropagation NN

For the implementation of the SLFNs, standard methods were chosen. For the training of SLFNs we chose Levenberg–Marquardt [21] method. It is based on gradient methods and has some disadvantages, e.g. the procedure is time consuming, there exists problem of local minimum, and other [22]. The weights and the value of the bias of the layer were calculated using Nguyen–Widrow initialization method [23]. The chosen restriction on the training process was the number of epochs equal to 300.

Extreme learning machine

ELM is a progressive learning algorithm of the SLFNs, which randomly chooses the input weights and analytically determines the output weights of SLFNs [22, 24]. Differently from the traditional learning algorithms, this algorithm tends to reach not only the smallest training error but also the smallest norm of weights (among all the least-squares solutions). Bartlett’s theory [25] on the generalization performance of the feedforward NN states that “the size of the weights is more important than the size of the network”. In theory, this algorithm tends to provide the best generalization performance and minimum training error at an extremely fast learning speed.

Combined method of ELM and LW regression

The locally weighted (LW) regression is described in [26–29]. In this paper we propose the method that puts together the ELM method and LW regression.

This type of NN, when it is used together with the LW regression, is known as locally weighted NN. The advantage of the locally weighted NN is that closer attention is paid to the regression curve rather than to coefficients, i.e. whether the regression curve accurately replicates the real data. The weakest place of the weighed NN is large time needed to calculate the optimal parameters. However, this is compensated by the accuracy of the method, compared with standard NN.

The algorithm of the proposed method can be divided into several stages. First, the sample is divided into training and testing samples: $\Omega_1 = \{(\mathbf{z}_t, \mathbf{y}_t), t = 1, \dots, T_1\}$, $T_1 < T$, is the training sample and $\Omega_2 = \{(\mathbf{z}_t, \mathbf{y}_t), t = T_1 + 1, \dots, T\}$ is the testing sample.

Stage 1. For the given T_1 and the samples $(\mathbf{z}_t, \mathbf{y}_t)$, $t = 1, \dots, T_1$, the training of NN is performed using the ELM and estimates of $(\widehat{\mathbf{w}}_\ell, \widehat{b}_\ell)$ (2) and the output weights $\widehat{\alpha}_\ell$ (3) are obtained ($\ell = 1, \dots, N$).

Stage 2. The estimates $(\widehat{\mathbf{w}}_\ell, \widehat{b}_\ell)$, $\ell = 1, \dots, N$ of the coefficients $(\mathbf{w}_\ell, b_\ell)$, $\ell = 1, \dots, N$ of the neurons of the hidden layer (evaluated at the first stage) are kept fixed and, for each new observation (the observation which does not belong to the training samples), only the weights α_ℓ of the neurons of output layer are recalculated ($\ell = 1, \dots, N$). Thus, during the process, only local information is used.

More precisely, for given query data \mathbf{z}_τ , $T_1 + 1 \leq \tau \leq T$, (τ is the time of the query) the new estimates $\hat{\alpha}_\ell$ are found using the LW regression:

$$\sum_{t=\tau-L}^{\tau-1} \left(\mathbf{y}_t - \sum_{\ell=1}^N \alpha_\ell \cdot \hat{\mathbf{g}}_\ell(\mathbf{z}_t) \right)^2 \exp \left\{ -\frac{1}{2\sigma^2} \|\mathbf{z}_t - \mathbf{z}_\tau\|^2 \right\} \rightarrow \min_{\{\alpha_\ell\}}. \quad (4)$$

Here

$$\hat{\mathbf{g}}_\ell(\mathbf{z}) := \mathbf{g}^\circ(\hat{\mathbf{w}}_\ell \cdot \mathbf{z} + \hat{b}_\ell), \quad \ell = 1, \dots, N,$$

σ is a bandwidth and $L < T_1$ is a size of sliding window.

Stage 3. The new estimates $\hat{\alpha}_\ell$, $\ell = 1, \dots, N$, obtained in the previous stage by solving (4) are used in (5) for the prediction of \mathbf{y}_q :

$$\hat{\mathbf{y}}_q = \sum_{\ell=1}^N \hat{\alpha}_\ell \cdot \hat{\mathbf{g}}_\ell(\mathbf{z}_q). \quad (5)$$

4 Process of the NN implementation

In this section, we will describe the process of the NN implementation and the main its characteristics: the transformation of data, the architecture of NN, measures of the prediction of accuracy.

According to [12], there is no a priori way to determine which scaling function is the most suitable for the NN. This depends on the characteristics of the data. A reasonable strategy is to estimate the model with different types of scaling functions and then to find out which one gives the best performance. Two linear scaling transformations, f_v , $v = 1, 2$, are considered.

Define $y_t := f_v(\text{CI}(t))$ (here $k = 1$), $\mathbf{z}_t := f_v(X(t-1))$ (here $n = m$), where f_1 (f_2) means the coordinatewise scaling transformation to the interval $[0, 1]$ (respectively, the symmetric interval $[-1, 1]$).

All models which were applied to the prediction of CI, have the same architecture of NN with the following features. The SLFNs model has one hidden layer. The number of nodes N of the hidden layer was chosen 5, 10, 15, 20. The number of inputs was chosen $m = 28$. The sigmoid (logistic) function was employed as the basic activation function \mathbf{g}° , ($\mathbf{g}^\circ(u) = 1/(1 + \exp(-u))$).

The following methods for training the SLFNs in sample Ω_1 and prediction for sample Ω_2 were used:

M_1 : The Levenberg–Marquardt method was applied to training; the evaluated characteristics were used for prediction.

M_2 : The ELM method was applied to training; the evaluated characteristics were used for prediction.

M_3 : The combined method: the ELM method was used for generating the NN coefficients; the values of predictions for the each query \mathbf{z}_τ were evaluated online using the LW regression (4). The size of sliding window was $L = 35$, the bandwidth $\sigma = 1$.

The *experiment* can be represented by

$$\widehat{\text{CI}} = h(\mathbf{X}, f_v, M), \quad v = 1, 2, \quad M \in \{M_1, M_2, M_3\}, \quad (6)$$

where $\widehat{\text{CI}}$ is the predicted CI, \mathbf{X} is the selected data set.

The technique of ensemble averaging [8] was applied when a number of different NN share a common input, and their individual outputs are combined to produce the overall output value.

This technique is used when there are K different models of NN which produce a similar error, and in this case it is difficult to choose the best model. The output value of the ensemble (of the t th data point) is obtained by using a simple average of all individual outputs. The mean square error of the ensemble (of the testing sample) is not greater than the error of the separate model of the NN [8].

In our case, the prediction of the ensemble yields a better solution compared to the predictions of separate NN. We chose the size of the ensemble $K = 50$: the experiment was performed 50 times using the different methods ($M \in \{M_1, M_2, M_3\}$) and the different transformations ($v = 1, 2$).

The following measures of the NN prediction accuracy were used:

- (i) The root mean square error:

$$\text{RMSE} = \left(\frac{1}{T} \sum_{t=1}^T (\text{CI}(t) - \widehat{\text{CI}}(t))^2 \right)^{1/2},$$

- (ii) The mean absolute percentage error:

$$\text{MAPE} = \frac{1}{T} \sum_{t=1}^T \left| \frac{\text{CI}(t) - \widehat{\text{CI}}(t)}{\text{CI}(t)} \right| \cdot 100,$$

- (iii) The positive predictive value, in per cent:

$$\text{PPV} = \frac{1}{T} \sum_{t=1}^T \mathbf{1}((\widehat{\text{CI}}(t+1) - \widehat{\text{CI}}(t))(\text{CI}(t+1) - \text{CI}(t)) > 0),$$

- (iv) The estimates of mean $E(\varepsilon)$ and variance $\text{Var}(\varepsilon)$ of the errors $\varepsilon(t) = \text{CI}(t) - \widehat{\text{CI}}(t)$, $t = T_1 + 1, \dots, T$.

The accuracy of training was tested using only RMSE.

Beyond the analysis of the errors, we investigated the phenomenon of the random walk (RW). In the case of RW, it is possible to use simpler methods than the models of NN for the forecasting of the series. There are papers that show that the one-step-ahead forecasts $\widehat{y}(t)$ (obtained using the NN) of the economic series behave like a RW process (see, e.g. [30]). It means that $\widehat{y}(t+1) = \widehat{y}(t) + \varepsilon(t)$ where t is time, $\{\varepsilon(t)\}$ are uncorrelated and homoscedastic random increments with zero mean. R. Site and J. Site [30] analyzed

Standard & Poor 500 financial time series using the Time Delay NN [31, 32] and Elman recurrent NN [33]. The authors compared RW prediction errors, NN prediction errors, and other. They found that the NN predicts like an RW process for all parameters they tried.

We applied the standard Student's *t*-test to test whether the predicted process is the RW. The following increments were considered:

- (i) the estimated increments by applying the NN: $\widehat{CI}(t+1) - CI(t)$,
- (ii) the actual increments: $CI(t+1) - CI(t)$.

5 Analysis of the results

The periods of training and prediction were 1998–2007 ($T_1 = 120$) and 2008–2010 ($T - T_1 = 36$), respectively. Figure 1 shows that the dynamics of training data totally differs from the trends in the data during then financial crisis (the period from the second half of 2008 until the start of 2010).

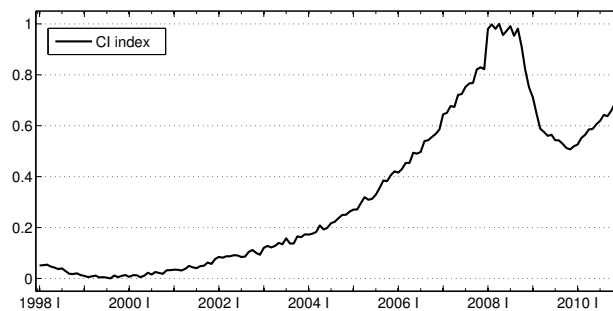


Fig. 1. The transformed CI index

The main attention was given to the measures of prediction accuracy for the period 2008–2010. The crucial question is how efficiently the NN can be applied in crisis periods to predict an unexpected decline in the economy and its recovery signs. From the huge number of the results obtained from different models (6), we will present only the main and the most important ones (Tables 1, 2).

A short explanation of the content of the tables is as follows. The M_1 , M_2 and M_3 denote the method of training and testing, the numbers in the brackets refer to the number of neurons of the hidden layer. The RMSE of training data is marked “_train” otherwise the accuracy measures are applied to the predicted data. In the tables descriptive statistics (average; maximum and minimum denoted, respectively, by “_max” and “_min”) of 50 repetitions of the each model in (6) are presented. The RMSE_2008 is the error of the prediction for the 2008.

When comparing the transformation f_1 versus f_2 , the former has smaller RMSE for both training and testing sets. The opposite conclusion holds only for the PPV of the classical model ($M_1(5)$).

The classical method M_1 gives the largest errors of RMSE; the MAPE is greater than 20 per cent, while $M_3(20)$ is only 3.66 per cent (Table 1). Concerning the methods of training and testing, the combined method of ELM and LW regression (M_3) distinguishes for its smallest RMSE (both of training and prediction) and MAPE values. The PPV of the combined method is similar to that of the ELM (M_2). The RMSE of model $M_3(20)$ is 0.053 in the year 2008 and 0.027 in the next year when the decline in the economy has slowed down (Table 1). This can be explained by the fact that during earlier periods we cannot identify such decline in the economy; therefore, it is impossible to train the NN to determine a sudden shift.

Table 1. Measures of prediction accuracy for the linear scale transformation f_1 .

	$M_1(5)$	$M_2(15)$	$M_2(20)$	$M_3(15)$	$M_3(20)$
RMSE_train	0.013	0.018	0.014	0.018	0.014
RMSE	0.154	0.116	0.124	0.039	0.036
RMSE_max	0.258	0.233	0.241	0.046	0.044
RMSE_min	0.091	0.052	0.046	0.034	0.033
RMSE_2008	0.115	0.097	0.100	0.056	0.053
RMSE_2009	0.128	0.108	0.145	0.031	0.027
RMSE_2010	0.179	0.118	0.103	0.022	0.020
MAPE	20.41	14.90	16.37	4.11	3.66
MAPE_max	33.15	31.84	35.33	5.29	4.68
MAPE_min	11.45	5.42	5.32	3.09	2.99
PPV	64.41	68.00	71.59	69.24	73.53
PPV_min	32.35	47.06	52.94	55.88	61.76
PPV_max	88.24	91.18	91.18	85.29	85.29
$E(\varepsilon)$	-0.006	0.007	-0.056	0.020	0.019
$\widehat{\text{Var}}(\varepsilon)$	0.010	0.009	0.009	0.001	0.001

Table 2. Measures of prediction accuracy for the linear scale transformation f_2 .

	$M_1(5)$	$M_2(15)$	$M_2(20)$	$M_3(15)$	$M_3(20)$
RMSE_train	0.028	0.035	0.029	0.035	0.028
RMSE	0.380	0.281	0.363	0.120	0.125
RMSE_2008	0.263	0.214	0.264	0.192	0.204
RMSE_2009	0.382	0.270	0.359	0.070	0.058
RMSE_2010	0.415	0.309	0.389	0.055	0.058
PPV	68.29	63.71	65.35	62.59	64.35

The RMSE of two models $M_3(15)$ and $M_3(20)$ are shown on Fig. 2 (the number of the experiment is n^*). It can be seen that only two values of the model $M_3(20)$ (the numbers of experiments are $n^* = 1, 9$) are larger than 0.04. The mean values of errors ε for models $M_1(5)$, $M_2(20)$ and $M_3(20)$ are presented in Fig. 3. Table 1 gives the values of mean estimates. Models M_1 and M_2 produce a similar mean value of errors though the estimated variance $\widehat{\text{Var}}(\varepsilon)$ is larger compared for the model M_3 ($\widehat{\text{Var}}(\varepsilon) = 0.001$).

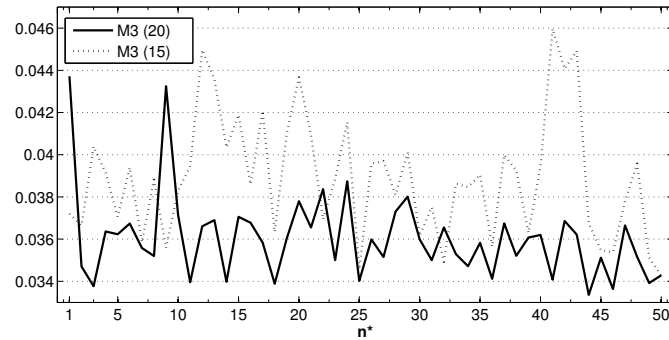


Fig. 2. RMSE of models $M_3(20)$ and $M_3(15)$, n^* is the number of the experiment.

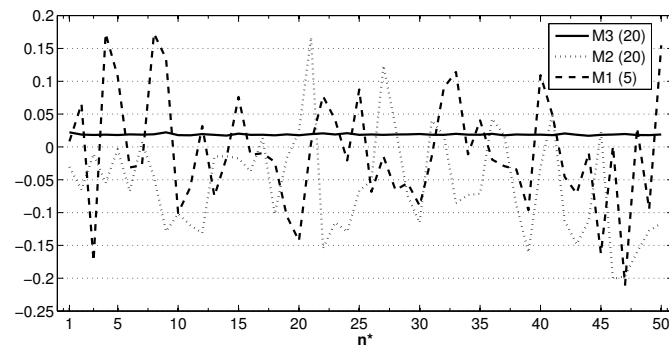


Fig. 3. The mean values of errors ε for models $M_1(5)$, $M_2(20)$ and $M_3(20)$, n^* is the number of the experiment.

In summary, the experiment showed that the proposed method of NN gives the smallest values of RMSE and MAPE and the largest value of PPV when the hidden layer has 20 nodes and the transformation f_1 is applied.

The analysis of the predictions of the model $M_3(20)$ concerning the RW process produced the following results (p -values of 2-tailed t -test): the sequence of predictions $\widehat{CI}(t)$, $t = 1, \dots, T$, is not an RW process (with p -value = 0.007; the actual CI (1998–2010) is not an RW process (p -value = 0.027), but some parts of the index are RW processes are; e.g. the process of the first 120 elements is not RW (p -value = 0.000), the process of the first 140 elements is RW (p -value = 0.104), the process of the last 120 elements is not RW (p -value = 0.022).

The accuracy of the NN predictions was compared with that of standard $AR(p)$ models [7]. The same set of the indicators was chosen with the linear scale transformation f_1 and applied the $AR(p)$ models to each of them. The values of \widehat{CI} were obtained by aggregating the series of the $AR(p)$ predictions and the same measures of accuracy were

evaluated (Table 1). The best value of the RMSE of $AR(p)$ was 0.091 when $p = 1$ and then increased with the order p . Therefore we restricted the further analysis to the cases $p \leq 4$. The PPV did not change much and was approximately equal to 64.7 per cent for $p = 1, 2, 3$ and 61.8 per cent for $p = 4$. When increasing the model order p the respective value of MAPE was not improved and was around 7 per cent. In all the considered cases the measures of accuracy for $AR(p)$ were worse compared to corresponding measures of the proposed method and similarly to the models M_1 or M_2 (Table 1).

6 Conclusions

In this paper, the problem of forecasting an artificial index CI of Lithuania's economy proposed in [3] during the crisis and later period 2008–2010 is investigated. The locally weighted NN, where the extreme learning machine (ELM) method is used for training and the locally weighted (LW) regression is used for the prediction, is proposed as a possible solution. This type of the NN has an advantage since more attention is paid to the linear parametric part of the NN thereby increasing the efficiency of the method.

The empirical analysis showed that the error of the CI predictions obtained by the proposed method is statistically smaller compared with the Levenberg–Marquardt and ELM methods as well as with standard $AR(p)$ models. It was found that for the considered data the best prediction accuracy yields the linear scaling transformation to the interval $[0, 1]$ and the NN with 20 hidden neurons.

Analysis of the results based on various accuracy measures (RMSE, MAPE and PPV, see Section 4) suggests that the proposed method may be used for data of rather small sample size and during periods when the dynamics of time series may have unexpected changes.

For forecasting economic changes and prediction of the Gross domestic product in periods of economic instability additional carefully validated synthetic indexes of economy are necessary.

References

1. R. Bandura, A survey of composite indices measuring country performance: 2008 update, A UNDP/ODS working paper, 2008.
2. J.E. Stiglitz, A. Sen, J.P. Fitoussi, Report by the Commission on the Measurement of Economic Performance and Social Progress, 2009, www.stiglitz-sen-fitoussi.fr.
3. J. Rukšėnaitė, Impact of factor rotation methods on simulation composite indicators, *Math. Model. Anal.*, **16**(3), pp. 418–431, 2011.
4. Statistics Lithuania, www.stat.gov.lt.
5. The Bank of Lithuania, www.lb.lt.
6. G. Rünstler, K. Barhoumi, S. Benk, R. Cristadoro, A. Den Reijer, A. Jakaitienė, P. Jelonek, A. Rua, K. Ruth, C. Van Nieuwenhuyze, Short-term forecasting of GDP using large monthly

- data sets: A pseudo real-time forecast evaluation exercise, Lietuvos bankas working paper, No. 1, 2008.
7. G.E.P. Box, G.M. Jenkins, G.C. Reinsel, *Time Series Analysis: Forecasting and Control*, Prentice Hall, Englewood Cliffs, NJ, 1994.
 8. S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd edition, Prentice Hall, New York, 1999.
 9. F. Takens, Detecting strange attractors in turbulence, in: *Dynamical Systems and Turbulence, Warwick 1980*, Lect Notes Math., Vol. 898, Springer, Heidelberg, 1981, pp. 366–381.
 10. A. Lendasse, E. De Bodt, V. Wertz, M. Verleysen, Non-linear financial time series forecasting – Application to the Bel 20 stock market index, *Eur. J. Econ. Soc. Syst.*, **14**(1), pp. 81–91, 2000.
 11. G. Tkacz, Neural network forecasting of Canadian GDP growth, *Int. J. Forecasting*, **17**, pp. 57–69, 2001.
 12. P. McNelis, P. McAdam, Forecasting inflation with thick models and neural networks, European Central Bank working paper, No. 352, 2004.
 13. A. Jakaitienė, M. Tamošiūnaitė, Lithuanian economic growth parameters prediction using neural networks, *Lith. Math. J.*, **43**, pp. 635–639, 2003.
 14. P. Zhang, D.M. Kline, Quarterly time-series forecasting with neural networks, *IEEE Trans. Neural Networks*, **18**, pp. 1800–1814, 2007.
 15. L. Barbone, G. Bodo, I. Visco, Costi e profitti in senso stretto: un’analisi su serie trimestrali, 1970–1980, *Bollettino della Banca d’Italia*, **36**, pp. 465–510, 1981.
 16. F.T. Denton, Adjustment of monthly or quarterly series to annual totals: An approach based on quadratic minimization, *J. Am. Stat. Assoc.*, **66**, pp. 99–102, 1971.
 17. R. Kaiser, A. Maravall, An application of TRAMO-SEATS: Changes in seasonality and current trend-cycle assessment: The German retail trade turnover series, Banco de Espana working paper, No. 0011, 2000.
 18. S.A. Aivazian, B.S. Mchitarian, *Applied Statistics and Base of Econometrics*, State University Higher School of Economics, Moscow, 1998 (in Russian).
 19. J.O. Kim, C.W. Muller et al., *Factor, Discriminant and Cluster Analysis*, Finansy i statistika, Moscow, 1989 (in Russian).
 20. G. Nicoletti, S. Scarpetta, O. Boylaud, Summary indicators of product market regulation with an extension to employment protection legislation, OECD Economics Department working papers, No. 226, 2000.
 21. M.T. Hagan, M. Menhaj, Training feed-forward networks with the Marquardt algorithm, *IEEE Trans. Neural Networks*, **5**, pp. 989–993, 1994.
 22. G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: Theory and applications, *Neurocomputing*, **70**, pp. 489–501, 2006.

23. D. Nguyen, B. Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights, in: *Proceedings of the International Joint Conference on Neural Networks*, Vol. 3, 1990, pp. 21–26.
24. G.-B. Huang, D. Wang, Y. Lana, Extreme learning machines: A survey, *Int. J. Mach. Learn. & Cybern.*, **2**, pp. 107–122, 2011.
25. P.L. Bartlett, The sample complexity of pattern classification with neural networks: The size of the weights is more important than the size of the network, *IEEE Trans. Inf. Theory*, **44**, pp. 525–536, 1998.
26. C.G. Atkeson, A.W. Moore, S. Schaal, Locally weighted learning, *Artif. Intell. Rev.*, **11**, pp. 11–73, 1997.
27. R. Baronas, F. Ivanauskas, R. Masloviskis, M. Radavičius, P. Vaitkus, Locally weighted neural networks for analysis of the biosensor response, *Kybernetika*, **43**, pp. 21–30, 2007.
28. A.W. Moore, J.G. Schneider, K. Deng, Efficient locally weighted polynomial regression predictions, in: *Proc. Fourteenth International Conference on Machine Learning*, 1997, pp. 236–244.
29. D. Ruppert, M.P. Wand, Multivariate locally weighted least squares regression, *Ann. Stat.*, **22**, pp. 1346–1370, 1994.
30. R. Sitte, J. Sitte, Neural networks approach to the random walk dilemma of financial time series, *Appl. Intell.*, **16**, pp. 163–171, 2002.
31. K.J. Lang, G.E. Hinton, The development of the time-delay neural network architecture for speech recognition, Carnegie-Mellon University, CMU-CS-88-152, 1988.
32. A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, K.J. Lang, Phoneme recognition using time-delay neural networks, *IEEE Trans. Acoust. Speech Signal Process.*, **37**, pp. 328–339, 1989.
33. J.L. Elman, Finding structure in time, *Cognitive Sci.*, **14**, pp. 179–211, 1990.