WD/25/1

EFSD
EURASIAN FUND FOR STABILIZATION AND DEVELOPMENT

GDP Nowcasting: From Traditional Econometric Models to Machine Learning Algorithms

T. Tsukarev, K. Poghosyan, K. Lemba, D. Grishin

WP/25/1

Keywords: nowcasting, real GDP growth rate, GDP deflator, bridge equation, MIDAS model, factor model, machine learning.

JEL Codes: C22, C52, C53, C55.

Authors' contact information:

Taras Tsukarev, EFSD Research Department: ttsukarev@efsd.org

Karen Poghosyan, EFSD Research Department: kpoghosyan@efsd.org

Kiryl Lemba, EFSD Research Department: KLemba@efsd.org

Danil Grishin, EFSD Research Department: dgrishin@efsd.org

The electronic version of this document is available at:
https://efsd.org/en/research/working-papers/

Table of Contents

Abbreviations and Acronyms

AIC	Akaike's information criterion
ARDL	autoregressive distributed lag
ARIMA	autoregressive integrated moving average
Armstat	Statistical Committee of the Republic of Armenia
Belstat	National Statistical Committee of the Republic of Belarus
CBA	Central Bank of the Republic of Armenia
DFM	dynamic factor model
DM	Diebold-Mariano
dollar	US dollar
EFSD	Eurasian Fund for Stabilization and Development
GDP	gross domestic product
GRU	gated recurrent unit
LASSO	least absolute shrinkage and selection operator
LSTM	long short-term memory
MAD	mean absolute deviation
MF	mixed-frequency
MIDAS	mixed data sampling
MLP	multilayer perceptron
MSE	mean squared error
NBRB	National Bank of the Republic of Belarus
OLS	ordinary least squares
RMSE	root mean squared error
RNN	recurrent neural network
RSS	residual sum of squares
SVM	support vector machine
SVR	support vector regression
VAR	vector autoregression

Acknowledgements

The authors are grateful to Gergely Akos Ganics (Research Economist at the Directorate-General for Financial Stability, Regulation and Resolution at the Bank of Spain), Dyna Heng (Senior Economist, Middle East and Central Asia Department, IMF), and the EFSD staff for their valuable comments and guidance. The authors bear responsibility for any mistakes or omissions.

Introduction

Analysis of the current economic situation is usually complicated by the fact that much of the key statistical data is released with significant delays and is subject to subsequent revisions (for example, GDP and its composition, government budget indicators, the balance of payments, etc.). Unlike meteorologists, who rely on current weather data to make short-term forecasts, macroeconomists have to analyse the current state of the economy and assess trends in key indicators going back at least 1–2 months. To overcome this problem, nowcasting methods are employed to estimate macroeconomic indicators in near real-time.

Nowcasting refers to the evaluation of the most recent past, the present and the prediction of the very near future (Bańbura, 2010; Malyugin, 2024). Nowcasting is typically used for those key macroeconomic variables which are collected at a low frequency (for example, on a quarterly basis) and released with a substantial lag. To obtain "early estimates" of such macroeconomic indicators, forecasting experts use information collected at a higher frequency and released in a more timely manner. For example, real GDP, which is the key indicator describing the state of the economy, is normally available only at a quarterly frequency, and its first estimates are released with a two to three-month lag. However, there are several variables closely related to GDP that are available at a much higher frequency (weekly or monthly) and released with shorter delays. Examples of such variables include industrial production, financial sector indicators, or the results of various surveys. These data can be employed to obtain early estimates of GDP using nowcasting methods. Key in this process is to use up-to-date, high frequency information in the context when data are released in a non-synchronous manner, with varying lags, which generates observation sets characterised by a so-called "ragged" edge.

A review of international practice indicates that researchers and experts engaged in nowcasting of macroeconomic variables have a wide range of tools at their disposal, typically based on time series econometrics. Among the various methods and models employed, bridge equations, MIDAS models, and dynamic factor models (DFM) have gained the most popularity, especially among central banks (Foroni and Marcellino, 2013; Chernis and Sekkel, 2017). These models are continuously evolving, driven by methodological innovations and integration of diverse data sources, thereby enhancing the accuracy and reliability of real-time economic estimates.

However, in recent years, machine learning methods and algorithms (ridge and LASSO regression, elastic net, boosting, bagging, random forest, support vector machine (SVM), multilayer perceptron (MLP), recurrent neural network (RNN), etc.) have gained increasing popularity, becoming an alternative to conventional econometric models (Desai, 2023). Machine learning algorithms are used for both regression[1] and classification purposes.

The purpose of this Working Paper is to determine whether machine learning methods and algorithms have the potential to improve the accuracy of estimates of macroeconomic variables, and thus become an effective complement or even an alternative to conventional nowcasting tools.

[1] This paper focuses on the application of the algorithms for nowcasting regression (Medeiros et al., 2021; Richardson et al., 2021).

This paper reviews three conventional nowcasting models — the bridge equation, MIDAS model, and factor model. The alternatives selected for comparison are machine learning methods and algorithms, such as ridge and LASSO regression, elastic net, boosting, bagging, random forest, SVM, MLP, and RNN.

Actual time series comprising macroeconomic indicators for the Republic of Armenia and the Republic of Belarus for the period between 2002 and 2024 were used for nowcasting based on conventional models and machine learning algorithms. The study employed monthly time series: 22 indicators for Armenia and 20 for Belarus. The dependent variables were the real GDP growth rate and the GDP deflator with a quarterly frequency. Most of the time series were subjected to primary statistical processing, namely seasonal adjustment, taking the logarithm, and first differencing to eliminate non-stationarity.

Several experiments were carried out to identify the nowcasting method with the best predictive accuracy. The idea behind them is as follows. The total number of observations in the time series was divided into two parts: a training set and a test set. The first part was used to estimate the model parameters, and the second part was used to compare the accuracy of the estimates (forecasts) produced by conventional models and machine learning methods with the actual data. The forecasts were generated using the recursive regression experiment scheme (Poghosyan and Magnus, 2012). The tool that minimises the loss function selected for those purposes is regarded as the best.

Our experiments show that the accuracy of the estimates produced using machine learning algorithms is superior to that of the estimates produced by conventional models. LASSO regression, boosting, random forest, SVM, and RNN proved to be the most effective tools.

Moreover, the Diebold-Mariano statistical test shows that there is a significant difference between the forecasts produced by LASSO regression, SVM, and RNN, and by the conventional bridge equation model. In addition, our calculations indicate that the forecast combination approach helps improve the accuracy of nowcasting even further compared with using individual methods in isolation.

This Working Paper has the following structure. Section 1 provides a brief literature review of the problem examined in the paper. Section 2 explores the models and algorithms employed in our research. Section 3 contains a detailed analysis of the initial data, including methods of preliminary statistical processing of time series. Section 4 describes the experiments and their outcomes. The last section presents the main conclusions and practical recommendations.

1. Literature Review

This section provides an overview of two groups of research publications on nowcasting macroeconomic variables. The first group focuses on conventional models based on econometric analysis of time series (bridge equations, the MIDAS model, and dynamic factor models). The second group comprises state-of-the-art, alternative tools based on machine learning methods and algorithms that have grown in popularity in recent years thanks to their ability to identify complex, non-linear dependencies in data and handle large datasets.

Bridge equations are one of the simplest and most widely used nowcasting tools — they use high-frequency indicators (such as monthly data on industrial production, retail sales or business activity indices) to estimate current values of low-frequency macroeconomic indicators (such as quarterly GDP). Bridge equations offer simplicity and interpretability, making them a key component of nowcasting systems widely used by central banks and economic forecasting institutions to monitor the current performance of GDP and other important macroeconomic variables. For example, Baffigi et al. (2004) used bridge equations to nowcast the GDP for Germany, France, Italy, and the euro area. The authors demonstrate that the use of national models and their subsequent aggregation help better account for the specific characteristics inherent to individual countries, thereby ensuring more accurate forecasts for the euro area as a whole. Thus, the paper emphasises the effectiveness of bridge equations in the context of nowcasting and their advantage over other basic models for short-term forecasting of GDP (ARIMA, VAR or structural models).

The nowcasting methodology has been further developed by Clements and Galvao (2008), who propose integrating mixed-frequency data into forecasting models. They examine the construction of various bridge equations that link high-frequency indicators to the target variable — US GDP growth. The authors provide empirical evidence of the effectiveness of their approach by conducting forecasting experiments using real-time data. They demonstrate how incorporating real-time data can improve forecasting accuracy by comparing the accuracy of forecasts produced by models incorporating mixed-frequency data with models based solely on low-frequency data. While emphasising the benefits of using a wider range of data sources, the authors stress the importance of adapting forecasting techniques to the digital age, characterised by increasingly available high-frequency data.

Notwithstanding their extensive application, the use of bridge equations is accompanied by a number of challenges (Zhemkov, 2021). Firstly, forecasts for all high-frequency exogenous variables in the model are needed to get estimates of the target low-frequency variable. These forecasts may rely on both simple (for example, autoregressive) and/or complex models. However, there is always a risk of increasing error due to inaccurate predictions of exogenous factors. Secondly, as practice suggests, it is challenging to identify the most informative variables — the selection of key predictors of many indicators available is far from a simple task. Therefore, modern literature tends to use the so-called ‘*bridging with factors*’ approach, when aggregate latent factors extracted from a dataset are used in equations instead of individual variables (Bernanke et al., 2005; Giannone et al., 2008). This approach has become dominant and is now standard practice in most central banks and international organisations.

The article by Giannone et al. (2008) was one of the key papers in the emergence of nowcasting as an independent field in applied macroeconomics. The authors put forward a dynamic factor model (DFM) designed to handle large mixed-frequency datasets on a real time basis. The main focus is on how the publication of new macroeconomic indicators during the quarter updates the estimate of current US GDP growth. Using real-time data, the authors illustrate that the model effectively tracks macroeconomic trends, providing a notable improvement in forecasting accuracy over traditional approaches. The paper also emphasises the importance of the sequence of publication of statistical data and shows that the greatest improvement in the accuracy of forecasts occurs early in the quarter, due to the publication of soft indicators (surveys and expectations). Since Giannone et al. (2008), quite a number of studies have been published on nowcasting of macroeconomic indicators for economies such as France (Barhoumi et al., 2010), New Zealand (Matheson, 2010), Norway (Aastveit et al., 2011), China (Giannone et al., 2013), Canada (Chernis and Sekkel, 2017), and others.

For example, Chernis and Sekkel (2017) developed and estimated a DFM for nowcasting Canada's GDP growth. The model uses a mix of hard and soft indicators, featuring a high share of international data, in particular, US data. In pseudo real-time experiments, the authors show that the DFM is superior in terms of accuracy to both simple models and other commonly used nowcasting approaches and models, such as MIDAS and bridge equations. The significance of incorporating US variables, such as employment and business surveys, is underscored given the rapid release of these data and the substantial impact of the US economy on Canada. The paper thus validates the effectiveness of the DFM given the presence of heterogeneous and imbalanced data, which makes it particularly useful for central banks and analysts involved in nowcasting and short-term forecasting.

Although factor models are commonly used in nowcasting due to their ability to handle large heterogeneous datasets, they have a number of limitations. Firstly, such models require complex specification and estimation, including the selection of the number of factors and lags, which can lead to overfitting and high computational burden (Doz et al., 2011; BańBura et al., 2010). Secondly, the interpretation of latent factors is challenging, which has a negative impact on the transparency of the model in the context of central bank applications (Giannone et al., 2008). In addition, despite its ability to handle unbalanced data panels, the performance of DFM can significantly decline when the number of outliers is large or the heterogeneity of the data is high (Mariano and Murasawa, 2003). The models also have poor adaptability to structural shifts in the economy and require regular recalibration. Finally, they are sensitive to errors in primary data processing and are affected by lags in data release and subsequent data refinements (Angelini et al., 2011). This underscores the importance of employing factor models with caution and leveraging alternative approaches within a systemic framework for nowcasting.

Another class of models that has become popular over the past two decades and that efficiently combines data of varying frequency is MIDAS models. The idea of using this tool was first proposed in Ghysels et al. (2004). Unlike other conventional models, MIDAS enables the direct incorporation of high-frequency indicators (for example, monthly or even daily data) into forecasts of low-frequency variables such as quarterly GDP. This renders the models particularly appealing within the framework of nowcasting, a field in which timeliness and flexibility in the use of incoming information are of importance (Ghysels et al., 2007). The main merits of MIDAS include reduced information loss due to data aggregation, the capacity for real-time adjustment of forecasts, and superior simplicity in comparison with DFM, whilst maintaining high levels of predictive accuracy. In the field of applied research, MIDAS models frequently demonstrate superior predictive accuracy, in contrast to bridge equations and other basic models, early in the quarter in particular (Clements and Galvao, 2008; Marcellino and Schumacher, 2010).

However, the performance of MIDAS is contingent on the appropriate choice of the weight function specification and the number of lags included, a process that requires meticulous empirical testing.

To date, the nowcasting literature has produced a set of advanced versions of econometric models designed to deal with mixed frequency data. These include: (1) univariate and multivariate mixed data sampling regression models, restricted (MIDAS) and unrestricted (U-MIDAS); (2) mixed-frequency vector autoregression models (MF-VAR), including their Bayesian modifications (MF-BVAR); (3) dynamic factor models adapted for mixed-frequency data analysis; and (4) Markov-switching mixed-frequency vector autoregression models (Markov-switching MF-VAR). These and other approaches are systematised and summarised in Foroni and Marcellino (2013), as well as in more recent regional and country reviews such as Linzenich and Meunier (2024), Zhemkov (2021), and Malyugin (2024), which reflect current applied practices.

However, in recent years, machine learning methods and algorithms have been increasingly used in nowcasting macroeconomic indicators, demonstrating competitive results compared with conventional econometric models. The key advantage of these approaches lies in their ability to detect non-linear dependencies, efficiently process large and high-frequency datasets, and adapt to structural changes in the economy. Apaydın et al. (2019) and Coulombe et al. (2020) show that random forest, gradient boosting, and neural network algorithms provide high accuracy for short-term GDP and inflation forecasting in a volatile economic environment. Medeiros et al. (2021) demonstrate that regularisation methods, such as LASSO and elastic net, are particularly effective when macroeconomic indicators are high dimensional and correlated.

Zhang et al. (2023) compared the performance of different machine learning algorithms, factor models, and MIDAS models in nowcasting China's real GDP growth rate, using 89 macroeconomic variables from 1995 to 2020. The authors found that some machine learning methods outperform the benchmark dynamic factor model. The ridge regression demonstrated the strongest performance, with superior results compared with other models in terms of forecasting error and effective recognition of the impact of the global financial crisis and COVID-19 shocks.

Sharma and Kathuria (2024) observed that a key benefit of conventional econometric models for nowcasting is their capacity to manage challenges related to data, including high dimensionality, delayed and non-synchronous data release, as well as frequent revisions. However, the authors argue that traditional models may not be suitable in the case of big data where there is exponential growth of hyperparameters. This is where machine learning models come in as an additional tool: they can deal with big data and perform better cross-validation for efficient hyperparameter selection.

Kapetanios and Papailias (2018) examined the potential for using big data in macroeconomic nowcasting, providing an empirical comparison of different methods for processing high-dimensional datasets. The authors show that approaches based on regularisation and machine learning (in particular, LASSO and ridge regression) can significantly improve the accuracy of predictions compared with traditional models, especially in the presence of a large number of predictors. The paper focuses on the selection of variables, the robustness of the model to noise and overfitting, and the benefits of combining signals from different data sources. In addition, Kapetanios and Papailias (2018) note the effectiveness of forecast combinations. The article emphasises the importance of adapting existing macroeconometric tools to the new normal of big data.

As Rossi and Sekhposyan (2019) emphasise, integrating machine learning with traditional macroeconomic models to improve the interpretability of results and respect abstract bounds remains an important area of development. Machine learning is thus becoming an integral part of today's nowcasting toolset, especially in the context of digital transformation and increasing availability of big data.

In the current macroeconomic environment, characterised by high volatility and rapid changes in business cycles, it is imperative for central banks, other economic decision-making institutions, and researchers to continuously enhance the tools for nowcasting. The review of existing approaches shows that traditional models such as bridge equations, DFM, and MIDAS remain relevant and commonly used. However, the rapid development of digital technologies and the increasing availability of high-frequency and unstructured data offer new opportunities to improve the accuracy of short-term forecasts. In this context, machine learning methods can serve as a valuable complement to conventional tools, particularly in the domains of predictor selection, the identification of non-linearities, and the adaptation to structural shifts. The need for further basic and applied research aimed at integrating these methods into the standard forecasting toolset is becoming increasingly evident. The development of hybrid models and systematic evaluation of their predictive accuracy are key areas for further basic and applied research in the field of nowcasting.

2. Models Overview

This section describes the conventional nowcasting econometric models, such as the bridge equation, MIDAS, and factor model, as well as machine learning methods and algorithms, compared in this paper.

Conventional Econometric Models

Bridge Equation

We use an autoregressive distributed lag model (ARDL model) including the lagged values of the dependent variable and the current and lagged values of the high-frequency explanatory variable. The model can be presented as follows:

$$y_{t_q} = a_0 + \sum_{i=1}^{p} a_i L^i y_{t_q} + \sum_{j=0}^{q} b_j L^j x_{t_q} + \varepsilon_t, \quad (1)$$

where y_{t_q} is a low-frequency time series (dependent variable),
x_{t_q} is a high-frequency time series (explanatory variable),
a_i, b_j are the unknown model parameters estimated using the OLS method.

Equation (1) is called a bridge equation because it links high-frequency and low-frequency data. This facilitates the estimation (prediction) of the values of a low-frequency variable, based on already available high-frequency data. Forecasting using model (1) is performed in two steps. The first step is to predict the high-frequency series until the end of the current quarter, with the produced values then aggregated to the quarterly frequency. The conventional ARIMA model is used to predict the high-frequency series. The second step is to replace x_{t_q} with the aggregated values of the high-frequency variable in model (1) and calculate the forecast for y_{t_q} based on the estimated model parameters. It should be noted that the optimal lag orders are determined based on the Akaike information criterion (AIC). When multiple high-frequency indicators are employed, the resulting forecast y_{t_q} is derived as the arithmetic mean of forecasts generated by different bridge equations.

For the purposes of this paper, the bridge equation approach will serve as a benchmark for comparison with other alternative nowcasting approaches.

MIDAS (Mixed Data Sampling) Model

A distinctive feature of the MIDAS model is the way in which it manipulates high-frequency data. MIDAS models use polynomial weight functions to link high- and low-frequency data. This approach makes MIDAS a direct forecasting tool. This is the key distinction between this approach and bridge equations, where each high-frequency indicator is predicted separately, and then the results are used for nowcasting. In the MIDAS model, the values of a low-frequency variable (quarterly real GDP growth or GDP deflator) are directly linked to the current and lagged values of a high-frequency indicator (for example, the industrial production index).

To explain in more detail how the low- and high-frequency variables are linked, we will make the following notations:

$t = 1,2, \ldots, T$ are the time steps for the low-frequency variable (y_t^L),

$t = 1,2, \ldots, m \cdot T$ are the time steps for the high-frequency variable (x_t^H),

m is an integer that indicates the number of times the value of the high-frequency variable appears per unit of time of the low-frequency variable.

For example, for quarterly GDP growth and monthly industrial production index $m = 3$.

We will refer to the low-frequency variable (for example, GDP growth rate) as y_t^L and to the high-frequency variable (industrial production index) as x_t^H. Then, if we want to build a regression model with the current value and the first lag x_t^H, the data matrix in the MIDAS model will take the following form:

$$\begin{bmatrix} y_2^L \\ y_3^L \\ \vdots \\ y_n^L \end{bmatrix} = \begin{bmatrix} x_6^H & x_5^H & \cdots & x_1^H \\ x_9^H & x_8^H & \cdots & x_4^H \\ \cdots & \cdots & \cdots & \cdots \\ x_{3n}^H & x_{3n-1}^H & \cdots & x_{3n-5}^H \end{bmatrix} \times \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_5 \end{bmatrix} + \begin{bmatrix} \epsilon_2 \\ \epsilon_3 \\ \vdots \\ \epsilon_n \end{bmatrix} \tag{2}$$

Since the model includes the current value and the first lag of the high-frequency variable, the second quarter of the low-frequency variable y_2^L will depend on three current values of x_t^H, which correspond to $\left\{x_6^H, x_5^H, x_4^H\right\}$, and three values of the high-frequency variable for the previous quarter, which correspond to values $\left\{x_3^H, x_2^H, x_1^H\right\}$. All other rows of the above model are explained in the same way.

The next step is to estimate the MIDAS model. The basic MIDAS model, which uses a single high-frequency indicator, can be represented as follows:

$$y_{t+h}^L = a_h + b_h C\left(L^{\frac{1}{m}}; \theta\right) x_t^H + \varepsilon_{t+h}^L, \tag{3}$$

where a_h, b_h are the unknown parameters of the model to be estimated;

$C\left(L^{\frac{1}{m}}; \theta\right) = \sum_{i=0}^{N} c\left(i; \theta\right) L^{i/m}$ are the coefficients of a polynomial of degree N;

$L^{i/m} x_t = x_{t-i/m}$ is the lag operator.

Model (3) can be expanded as follows:

$$y_{t+h}^L = a_h + b_h\left(c\left(0;\theta\right) x_{t-0/m} + c\left(1;\theta\right) x_{t-1/m} + \ldots + c\left(N;\theta\right) x_{t-N/m}\right) + \varepsilon_{t+h}^L. \tag{4}$$

Model (4) shows that the parameters of model (3) are estimated in two steps. The first step is to calculate the values of weights $c(i; \theta)$. The second step is to use the calculated weights for lagged values and aggregate x_t to produce a weighted variable. The point of using weight coefficients is to assign less weight to later lagged values x_t^H and vice versa. Weight coefficients $c(i;\theta)$ are calculated according to the following formula:

$$c(i;\theta)=\frac{x_i^{\theta_1-1}\left(1-x_i\right)^{\theta_2-1}}{\sum_{i=1}^{N}x_i^{\theta_1-1}\left(1-x_i\right)^{\theta_2-1}}, \tag{5}$$

where $x_i=\frac{i}{N+1}$, and the weight coefficients of $c(i;\theta)$ depend on θ_1 and θ_2.

The problem is to find values of θ_1 and θ_2 at which estimated model (3) will have the minimum value of the residual sum of squares. This problem can be solved using the non-linear least squares method. In other words, the objective is to simultaneously find values of a_h, b_h, θ_1, θ_2 that minimise the residual sum of squares in model (3).

As demonstrated by Ghysels et al. (2004), the MIDAS model produces superior estimates when compared with the approach of aggregating a high-frequency series to a lower-frequency series, followed by model estimation. At the same time, the authors note that the bias in the estimates as a result of the discretisation of the explanatory variable is almost the same as in the conventional distributed lag model. Moreover, the bias of estimates decreases with increasing frequency of explanatory variables.

Factor Model

In contrast to the bridge equation and the MIDAS model, this approach relies on the use of a relatively large input dataset. We will use the methodology proposed in Doz et al. (2011) to explain the nowcasting of a low-frequency target variable with a factor model. This methodology was developed to extract a small number of dynamic principal components (factors) from a large dataset, allowing efficient compression of information and accounting for hidden dependencies.

The methodology of Doz et al. (2011) comprises two steps. The first step is to calculate static principal components based on balanced data. These components are then used to determine the initial conditions of the Kalman filter. The second step is to use the defined initial conditions in the Kalman filter and reverse smoothing to calculate the dynamic principal components.

The model can be presented as follows:

$$x_{t_m}=\Lambda f_{t_m}+\xi_{t_m},\ \xi_{t_m}\sim N\left(0,\Sigma_\xi\right), \tag{6}$$

$$f_{t_m}=\sum_{i=1}^{p}A_i f_{t_m-i}+B\eta_{t_m},\ \eta_{t_m}\sim N\left(0,I_q\right). \tag{7}$$

Equation (6) links N of monthly indicators to principal components $r\times 1$ through a matrix of factor loadings Λ.

Equation (7) describes the rule that generates unobserved component f_{t_m}, where A is the coefficient matrix estimated based on the static principal components using the OLS method.

The forecast of the dependent variable for the current quarter is based on the bridge equation using the following formula:

$$\hat{y}_{t_q} = a + \beta \hat{f}_{t_q}, \tag{8}$$

where $\hat{f}_{t_q}$ are the values of $\hat{f}_{t_m}$ aggregated to quarterly frequency.

Machine Learning Methods and Algorithms

One thing that machine learning approaches have in common is that their algorithms run with minimum human intervention. One of the taxonomies used to classify machine learning algorithms is presented in the Working Paper by Tsukarev, Poghosyan and Lemba (2024).

This Working Paper presents a concise overview of the machine learning methods and algorithms that are sufficiently prominent in the field of nowcasting and can be utilised to reveal comparative advantages.

Ridge Regression

The ridge regression algorithm is similar to OLS. It was developed to overcome the instability of OLS estimates[2] by penalising the sum of squared parameter values on the basis of regularisation[3] $L_2 \left(\sum_{j=1}^{p} \beta_j^2\right)$. In particular, the model's coefficients are estimated using the following optimisation formula (Hoerl and Kennard, 1970):

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left[\sum_{i=1}^{n} \Bigg(\underbrace{y_i - \sum_{j=1}^{p} x_{ij}\beta_j}_{RSS} \Bigg)^2 + \underbrace{\lambda \sum_{j=1}^{p} \beta_j^2}_{\text{ridge penalisation}} \right], \tag{9}$$

where n is the number of observations,
p is the number of explanatory variables,
$\lambda \geq 0$ is the parameter controlling the degree of shrinkage[4].

The formula (9) can be presented in matrix form:

$$RSS(\lambda) = (Y - X\beta)^T (Y - X\beta) + \lambda \beta^T \beta. \tag{10}$$

If we take the derivatives with respect to β, we produce the estimates of unknown parameters:

$$\hat{\beta} = \left(X^T X + \lambda I\right)^{-1} X^T Y, \tag{11}$$

where I is the identity matrix.

[2] Caused by the strong correlation among the independent variables (multicollinearity).

[3] Regularisation is a method used to reduce complexity of a model to prevent its overfitting. Regularisation L_1 penalises weights by adding the sum of absolute values of the model's coefficients to the loss function. Regularisation L_2 performs a similar operation by adding the sum of squared values of the model's coefficients.

[4] The higher the value of λ, the stronger the shrinkage.

According to Formula (11), unlike in the case of OLS, a positive constant λ is added to the diagonal of matrix X^TX. If $\lambda = 0$, then ridge regression estimates are exactly equal to OLS estimates, and if $\lambda \to \infty$, the model's coefficients tend to zero. Therefore, in practice parameter λ ranges from 0 to ∞.

Ridge regression parameters can be estimated using the above analytical formula. However, it is also possible to perform the estimation by applying the gradient descent algorithm, which will save a lot of computation time. Convergence of this method depends on the selected initial parameters.

LASSO Regression (Least Absolute Shrinkage and Selection Operator)

If the LASSO regression is applied, regularisation $L_1\left(\sum_{j=1}^{p}|\beta_j|\right)$ is used for the shrinkage of coefficients, this being the main difference from the ridge regression (Tibshirani, 1996):

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\left[\sum_{i=1}^{n}\Bigg(\underbrace{y_i - \sum_{j=1}^{p}x_{ij}\beta_j}_{RSS}\Bigg)^2 + \underbrace{\lambda\sum_{j=1}^{p}|\beta_j|}_{\text{LASSO penalisation}}\right]. \tag{12}$$

Constraint $\sum_{j=1}^{p}|\beta_j|$ renders the solution non-linear and, accordingly, the gradient descent algorithm is employed in the solution.

Elastic Net

Elastic net is a weighted combination of the ridge regression and the LASSO regression, and has the following optimisation formula:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\left[\sum_{i=1}^{n}\Bigg(\underbrace{y_i - \sum_{j=1}^{p}x_{ij}\beta_j}_{RSS}\Bigg)^2 + \lambda\sum_{j=1}^{p}\Bigg[\underbrace{(1-a)\beta_j^2}_{Ridge} + \underbrace{a|\beta_j|}_{LASSO}\Bigg]\right], \tag{13}$$

where a is the penalisation weight.

Ensemble Machine Learning Methods

Inasmuch as ensemble machine learning methods are related to the decision tree, we will elaborate on the latter before describing the methods used in our research.

The main task of the decision tree is to optimally divide the original set of observations (the parent node) into two subsets (two child nodes).

Suppose x_{nd} is the d^{th} predictor from vector x_n. If x_{nd} is a quantitative variable, the division rule is as follows: $x_{nd} \le t$, where t is a certain threshold level.

Suppose N_m is the m^{th} node of the decision tree, while C_m^L and C_m^R are two child nodes for N_m. Then observations in node N_m will go to child node C_m^L if $x_{nd} \le t$, and to child node C_m^R otherwise.

Suppose $n \in N_m$ is the number of observations in node N_m, and $\bar{y}_m$ is the mean value of node N_m. Then the residual sum of squares (RSS) is calculated as follows:

$$RSS_m = \sum_{n \in N_m} \left(y_n - \bar{y}_m\right)^2, \tag{14}$$

where y_n are the observations in node N_m.

The loss function for the entire decision tree is the total value of RSS_T, including losses at buds and leaves. Suppose I_m is an index of the m^{th} bud or leaf. Then total loss for the entire tree will be equal to:

$$RSS_T = \sum_m \sum_{n \in N_m} I_m RSS_m . \tag{15}$$

The main criterion of optimal division is the minimisation of indicator RSS_T. Its reduction as a result of division of node N_m into child nodes C_m^L and C_m^R is calculated as follows:

$$\Delta RSS_T = RSS_m - \left(RSS_{C_m^L} + RSS_{C_m^R}\right). \tag{16}$$

As we see, ΔRSS_T is minimised when the differential between the two child nodes, $RSS_{C_m^L}$ and $RSS_{C_m^R}$, is maximised (Breiman and Ihaka, 1984).

Bagging

Bagging, or bootstrap aggregating, is based on a decision tree and includes the following steps (Breiman, 1996; Hastie et al., 2009):

1. The total number of bootstrap iterations[5] and maximum and minimum decision tree depths[6] are set (Efron and Tibshirani, 1993).
2. A decision tree is built for each bootstrap sample, and mean values are estimated for the dependent variable at the ends of the tree (leaves). The obtained forecasts for the given bootstrap sample are saved.
3. Dependent variable values are forecast on the basis of the test sample, and the result is saved.
4. The first three steps described above are repeated for the set number of bootstrap iterations. The projected dependent variable values obtained from all bootstrap iterations are averaged, producing the final forecast of the dependent variable (Tsukarev, Poghosyan and Lemba, 2024).

Random Forest

The main difference is that while bagging uses all explanatory variables included in the analysis, the random forest algorithm uses only some randomly selected ones (Breiman, 2001; Tsukarev, Poghosyan and Lemba, 2024).

---

[5] As a rule, about 100 bootstrap iterations are sufficient to obtain stable decision tree forecasts.
[6] The maximum depth is set so as to prevent decision tree overfitting, while the minimum depth is usually set to two.

Boosting

Boosting is also based on the application of a decision tree and involves the following steps (Freund and Schapire, 1997):

1. Original values of weights w_n^1 are set, where n = 1, 2, ..., N. It should be noted that N is the number of lines in the training sample.
2. The total number of bootstrap iterations is set; for example, suppose that T = 100. Then, for each bootstrap iteration, a decision tree is built, and mean values of the dependent variable at the ends of the tree (leaves) are predicted based on the training sample.
3. Then error vector L_n^t is calculated for n = 1, 2, …, N:

$$L_n^t = \frac{|y_n - \hat{y}_n|}{D^t}, \tag{17}$$

where $D^t = \underbrace{\max}_{n}\{|y_n - \hat{y}_n|\}$.

4. The mean value of observation errors $\overline{L}^t$ is calculated as follows:

$$\overline{L}^t = \sum_{n=1}^{N} L_n^t w_n^t. \tag{18}$$

If $\overline{L}_n^t \geq 0.5$, the iteration process is finished, and the total number of bootstrap iterations T = 100 is substituted with t – 1, where t = 1, 2, …, T. If not, the iteration process continues.

5. We calculate $\beta^t = \frac{\overline{L}^t}{1-\overline{L}^t}$. The lower β^t, the higher the degree of confidence in the model.
6. The model's weight coefficients are updated in accordance with the following rule: $w_n^{t+1} = \frac{w_n^t (\beta^t)^{1-L_n^t}}{\sum_{n=1}^{N} w_n^t (\beta^t)^{1-L_n^t}}$, which decreases the weight of the observations with a relatively larger error.
7. If $\overline{L}^t \geq 0.5$, the values of dependent variable forecasts are calculated as a weighted median with weights $ln\left(\frac{1}{\beta^t}\right)$.

Support Vector Machine

The support vector machine is a machine learning algorithm originally developed for classification purposes, but effectively applied to regression as well. The basic SVM methodology is presented in Cortes and Vapnik (1995). The main idea of this approach is to find a hyperplane that maximises the separation of observations of different classes (in the case of classification) or approximates the relationship between variables with minimum error (in the case of regression).

A key feature of the approach is the use of kernel functions, which allow efficient modelling of non-linear dependencies by translating input data into a higher dimensional space where linear separation becomes possible. The model retains good generalisation performance due to structural error control (*regularisation*) and selection of an area of acceptable deviations (ε-*insensitive zone* for SVR).

In the context of macroeconomic nowcasting, SVM has been used to forecast key indicators such as GDP, inflation or business activity indexes, especially when non-linear dependencies or complex interactions between predictors are present (Huang et al., 2005; Kim, 2003; Tay and Cao, 2001). The method is robust to overfitting, especially in small samples, making it useful when the number of quarterly observations is limited.

SVM involves the minimisation of $J(\beta)=\frac{1}{2}\beta'\beta$ while restricting $\forall n: \left| y_n - (x_n'\beta + b) \right| \leq \epsilon$. The optimal values of β, which minimise $J(\beta)$, are found using the gradient descent algorithm. Those errors that are at a distance ϵ from value y are ignored and equated to zero. The above can be described as follows:

$$L_\epsilon = \begin{cases} 0, \text{ if } \left| y - f(x) \right| \leq \epsilon \\ \left| y - f(x) \right| - \epsilon, \text{ otherwise}. \end{cases} \tag{19}$$

Multilayer Perceptron

A multilayer perceptron (multilayer neural network) is a generalisation of a single-layer neural network. Each layer consists of neurons, with the number of neurons in the first layer corresponding to the number of input variables $x = (x_1, x_2, \ldots, x_n)$. Subsequent layers are called *hidden layers*. Individual neurons of each layer are a linear combination of the neurons of the previous layer. For example, Figure 1 shows a schematic representation of a neural network comprising three hidden layers.

Figure 1. Neural Network Architecture

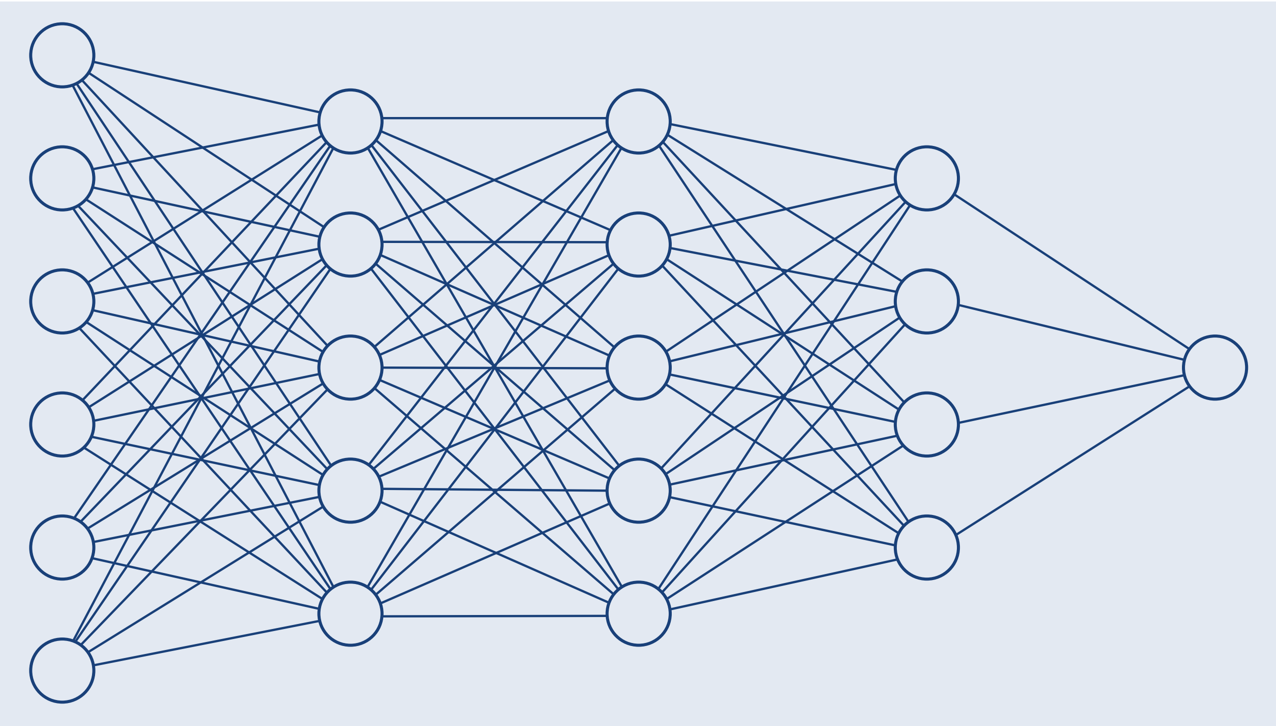


Figure 1 shows that the input layer contains six neurons because the number of input variables is six. Next come three hidden layers, the number of neurons in the first two layers is five and in the third hidden layer it is four. The output layer consists of a single neuron.

The algorithm for the interaction between neurons works as follows:

1. In the first layer, weight coefficients are randomly generated for each neuron depending on the number of predictors.
2. The output variables are calculated using the original weight matrix, and the weights are optimised based on this.
3. The optimised weight matrices are passed to the next layer for which they serve as original weights.
4. The weight coefficients are optimised at each layer based on the *backpropagation* algorithm.

In Figure 1, the algorithm is completed after the third layer, where the number of neurons is four, and a forecast is made. This algorithm is discussed in detail in Chong and Zak (2013), as well as Tsukarev, Poghosyan and Lemba (2024)[7].

Recurrent Neural Network

A recurrent neural network is a class of neural networks specifically designed to process sequential data inputs such as time series. The RNN is distinct from conventional neural networks in that it possesses an internal memory and can thus account for the dependence of the current value on previous states. The key concept underlying RNN is the cyclic transfer of information: at each step, the model receives an input value and simultaneously considers the latent state accumulated from previous steps. This makes the RNN particularly suitable for forecasting exercises in an economy where there is a time dependency between indicators (for example, between current and past values of GDP, inflation, production, etc.).

However, classical RNN face the vanishing or exploding gradient problem, making them perform poorly on long data sequences. To address this problem, improved architectures such as LSTM and GRU have been developed to model longer-term dependencies and are widely used for nowcasting and time series forecasting in economics (for example, see Bandara et al., 2020; Fischer and Krauss, 2018).

In general, the RNN algorithm includes the following steps[8]:

1. The start of each sequence is initialised with zeros.
2. A fragment of the data and target value sequence is extracted from the entire input sequence.

---

[7] The reader may also find the following repository with a detailed tutorial on the multilayer perceptron for regression useful: https://github.com/AndreasLeitherer/Tutorial_multilayer_perceptron/blob/main/nn_regression.ipynb.

[8] A detailed description of the RNN algorithm and related functions, as well as codes for Python, is available at: https://www.kaggle.com/code/fareselmenshawii/rnn-from-scratch.

3. The forward propagation method is applied to the current sequence and output probabilities are calculated.
4. Weight coefficients are updated based on the backpropagation method, and parameter gradients are then calculated.
5. The loss function is calculated and updated based on the updated weights.
6. The updated weights of the previous latent state are applied to the next data sequence.

3. Data and Software

In this Working Paper, we used time series for macroeconomic indicators for the Republic of Armenia and the Republic of Belarus for the period from 2002 to 2024. The main target variables were the real GDP growth rate and GDP deflator with quarterly frequency. All other indicators were used with monthly frequency. The full list of indicators analysed is available in Annexes 1 and 2.

22 high-frequency macroeconomic indicators were selected for Armenia and 20 for Belarus. The sample includes indicators characterising various sectors of the economy. The real sector is represented by output indicators and sectoral producer prices. The external sector includes data on exports and imports of goods and services, gross international reserves, and nominal exchange rates. The financial sector covers key monetary and financial indicators.

It should be noted that although the number of explanatory variables could be larger, there is no empirical evidence to suggest that including more variables would improve the accuracy of the forecasting (Barhoumi et al., 2010; Alvarez and Perez-Quiros 2016).

Prior to modelling, the time series were subjected to preliminary processing which consisted of the following steps:

1. Seasonal adjustment of the time series (where seasonality was identified),
2. Taking logarithms of all time series variables except interest rate series,
3. Transformation of the time series to a stationary form through first differencing.

All econometric modelling (except for the bridge equation) was performed in the MATLAB environment[9]. An interesting practical solution was also proposed by Linzenich and Meunier (2024), who developed a software package for nowcasting using the bridge equation, MIDAS, and DFM[10].

Python and its libraries were used to develop machine learning algorithms, including scikit-learn for traditional methods and Keras and TensorFlow for deep learning tasks. In addition, the authors of the Working Paper created their own auxiliary codes to perform the calculations and visualise the data.

---

9 MATLAB codes for MIDAS are available at: https://www.mathworks.com/matlabcentral/fileexchange/45150-midas-matlab-toolbox. MATLAB codes for DFM are presented here: https://github.com/FRBNY-TimeSeriesAnalysis/Nowcasting.

10 The package can be found at: https://github.com/baptiste-meunier/Nowcasting_toolbox.

4. Empirical Results

The main objective of the experiments in this study was to estimate the real GDP growth rate and the GDP deflator using actual monthly macroeconomic indicators.

It is important to note, however, that a key methodological challenge in real-time nowcasting is the absence of released high-frequency data for individual months of the current quarter. To address this problem, it is proposed to use the approach described in the Working Paper by Tsukarev, Poghosyan and Lemba (2024), where the ARIMA model is used as the basic method.

To compare predictive properties of the nowcasting methods reviewed in Section 3, we conducted a regression experiment. The experiment scheme comprised the following steps:

1. The available high-frequency series were converted to a quarterly frequency, producing a sample of data spanning from 2002Q2 to 2024Q4 (91 observations for each variable).
2. The entire data sample was initially divided into two parts: a training sample (the first 70% of all observations, 2002Q2–2018Q2, 65 observations) and a test sample (the last 30% of all observations, 2018Q3–2024Q4, 26 observations).
3. The model was estimated on the basis of the actual time series in the training sample (2002Q2–2018Q2).
4. The quarterly values of high-frequency series were used to nowcast for one quarter ahead (the first forecast period was 2018Q3).
5. The length of the training sample was increased by one quarter (2002Q2–2018Q3) and model parameters were re-estimated.
6. The re-estimated model was used to make a nowcast for one quarter ahead (2018Q4).
7. Steps 5 and 6 were repeated until the training sample covered the period through 2024Q3.
8. The resultant point nowcasts were compared against the actual data, and two forecast accuracy metrics representing loss functions were computed:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}\left(y_t - \hat{y}_t\right)^2}{n}}, \tag{20}$$

$$MAD = \frac{1}{n}\sum_{t=1}^{n}\left|y_t - \hat{y}_t\right|, \tag{21}$$

where y_t and $\hat{y}_t$ are the actual and estimated values of the indicator,
n is the total number of estimations, in this case, $n = 26$.

The results of RMSE and MAD calculations for the real GDP growth rates and the GDP deflators in Armenia and Belarus are presented in Tables 1–4.

Table 1. Accuracy Assessment Results of Nowcasting for Armenia's Real GDP Growth Rates (for the period 2018Q3–2024Q4)

Model	RMSE	RMSE (relative to bridge)	DM-stat.	MAD	MAD (relative to bridge)	DM-stat.
Bridge equation	3.56	–	–	2.20	–	–
MIDAS	3.38	0.95	0.42	2.15	0.98	0.07
DFM	3.63	1.02	0.15	2.42	1.10	1.01
Ridge	3.44	0.96	0.37	2.28	1.03	0.53
LASSO	3.38	0.95	0.43	2.35	1.07	0.74
Elastic Net	3.46	0.97	0.26	2.42	1.10	1.03
Boosting	3.52	0.99	0.53	2.13	0.97	0.72
Bagging	3.59	0.99	0.08	2.15	0.98	1.48
Random Forest	3.41	0.96	0.08	2.12	0.96	1.48
SVM	3.45	0.97	0.22	2.53	1.15	1.11
MLP	3.45	0.97	0.28	2.57	1.17	1.25
RNN	3.42	0.96	0.61	2.16	0.98	0.07

Table 2. Accuracy Assessment Results of Nowcasting for Armenia's GDP deflator (for the period 2018Q3–2024Q4)

Model	RMSE	RMSE (relative to bridge)	DM-stat.	MAD	MAD (relative to bridge)	DM-stat.
Bridge equation	2.95	–	–	2.10	–	–
MIDAS	2.29	0.78	1.98**	1.52	0.72	2.79***
DFM	2.65	0.90	0.81	1.60	0.76	2.24**
Ridge	2.34	0.79	2.30**	1.43	0.68	4.06***
LASSO	2.22	0.75	2.06**	1.51	0.72	2.79***
Elastic Net	2.34	0.79	2.01**	1.54	0.73	2.62***
Boosting	2.41	0.81	1.90*	1.52	0.72	2.74***
Bagging	2.36	0.80	1.90*	1.56	0.74	2.74***
Random Forest	2.33	0.79	1.90*	1.52	0.72	2.74***
SVM	2.51	0.85	1.09	1.65	0.78	1.84*
MLP	2.58	0.87	0.88	1.74	0.83	1.22
RNN	2.17	0.73	2.67**	1.47	0.70	3.87***

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

Table 3. Accuracy Assessment Results of Nowcasting for Belarus's Real GDP Growth Rate (for the period 2018Q3–2024Q4)

Model	RMSE	RMSE (relative to bridge)	DM-stat.	MAD	MAD (relative to bridge)	DM-stat.
Bridge	2.34	–	–	2.02	–	–
MIDAS	2.12	0.91	0.81	1.74	0.86	0.71
Factor model	2.05	0.88	1.03	1.68	0.83	0.92
Ridge	1.97	0.84	1.21	1.54	0.76	1.46
LASSO	1.93	0.82	1.33	1.51	0.75	1.56
Elastic Net	1.94	0.83	1.32	1.52	0.75	1.53
Boosting	1.65	0.71	1.68*	1.48	0.77	1.69*
Bagging	1.75	0.75	1.55	1.54	0.76	1.64
Random Forest	1.66	0.71	1.66	1.47	0.73	1.71*
SVM	1.44	0.62	1.96**	1.43	0.71	1.98**
MLP	2.09	0.89	1.04	1.69	0.83	0.87
RNN	2.05	0.87	0.99	1.65	0.82	0.90

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

Table 4. Accuracy Assessment Results of Nowcasting for Belarus's GDP Deflator (for the period 2018Q3–2024Q4)

Model	RMSE	RMSE (relative to bridge)	DM-stat.	MAD	MAD (relative to bridge)	DM-stat.
Bridge	2.14	–	–	1.95	–	–
MIDAS	2.10	0.98	0.25	1.89	0.97	0.28
Factor model	2.06	0.96	0.20	1.86	0.95	0.15
Ridge	1.64	0.77	1.88*	1.42	0.73	1.76*
LASSO	1.54	0.72	1.95*	1.36	0.70	1.98**
Elastic Net	1.57	0.73	1.95**	1.37	0.70	2.05**
Boosting	1.73	0.81	1.77*	1.46	0.75	1.71*
Bagging	1.70	0.79	1.80*	1.53	0.79	1.65*
Random Forest	1.65	0.77	1.80*	1.50	0.77	1.72*
SVM	1.78	0.83	1.59	1.54	0.79	1.71*
MLP	1.80	0.84	1.56	1.58	0.81	1.60
RNN	1.43	0.67	2.44**	1.40	0.72	2.14**

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

As shown in Tables 1 and 3, which present the results of the real GDP growth rate nowcasting, the smallest forecasting error by the RMSE metric for Armenia is achieved using LASSO regression. However, the produced value is not statistically different from the results of conventional nowcasting methods. In the case of Belarus, the smallest forecasting error is observed for the SVM algorithm, and its superiority over the bridge regression is statistically significant.

The results of the GDP deflator nowcasting presented in Tables 2 and 4 demonstrate that for both countries the smallest forecasting error is achieved using the RNN algorithm. This suggests that the method is highly effective in modelling the dynamics of price indicators.

The overall conclusion is that machine learning algorithms significantly minimise the nowcasting error, indicating the comparative accuracy of these algorithms and their usefulness for nowcasting.

Figures 2–5 show the dynamics of actual and estimated values of real GDP growth and the GDP deflator for Armenia and Belarus. The results suggest that the machine learning methods and algorithms demonstrate superior accuracy in capturing the vector of change in the indicators analysed when compared with conventional nowcasting models.

The next question to be answered is how significant the differences are between the nowcasting results produced by machine learning algorithms and the predictions based on econometric models. The bridge equation, a conventional econometric model, was chosen as the base model for comparison. Its use is based on its simplicity and similarity to the nowcasting approach in machine learning.

The Diebold-Mariano statistical test (Diebold and Mariano, 1995) was applied to assess the significance of differences in the accuracy of the estimates, with the results presented in Tables 1–4.

Data analysis shows that there are no statistically significant differences between econometric models and machine learning algorithms in terms of the real GDP growth rate of Armenia (Table 1). However, in the case of Belarus (Table 3), the SVM and boosting methods demonstrate significantly higher accuracy compared with the bridge equation. As for GDP deflator forecasting, machine learning algorithms are superior to conventional approaches in most cases — this is true for both Armenia (Table 2) and Belarus (Table 4).

Figure 2. Nowcasting of Armenia's Real GDP Growth Rates

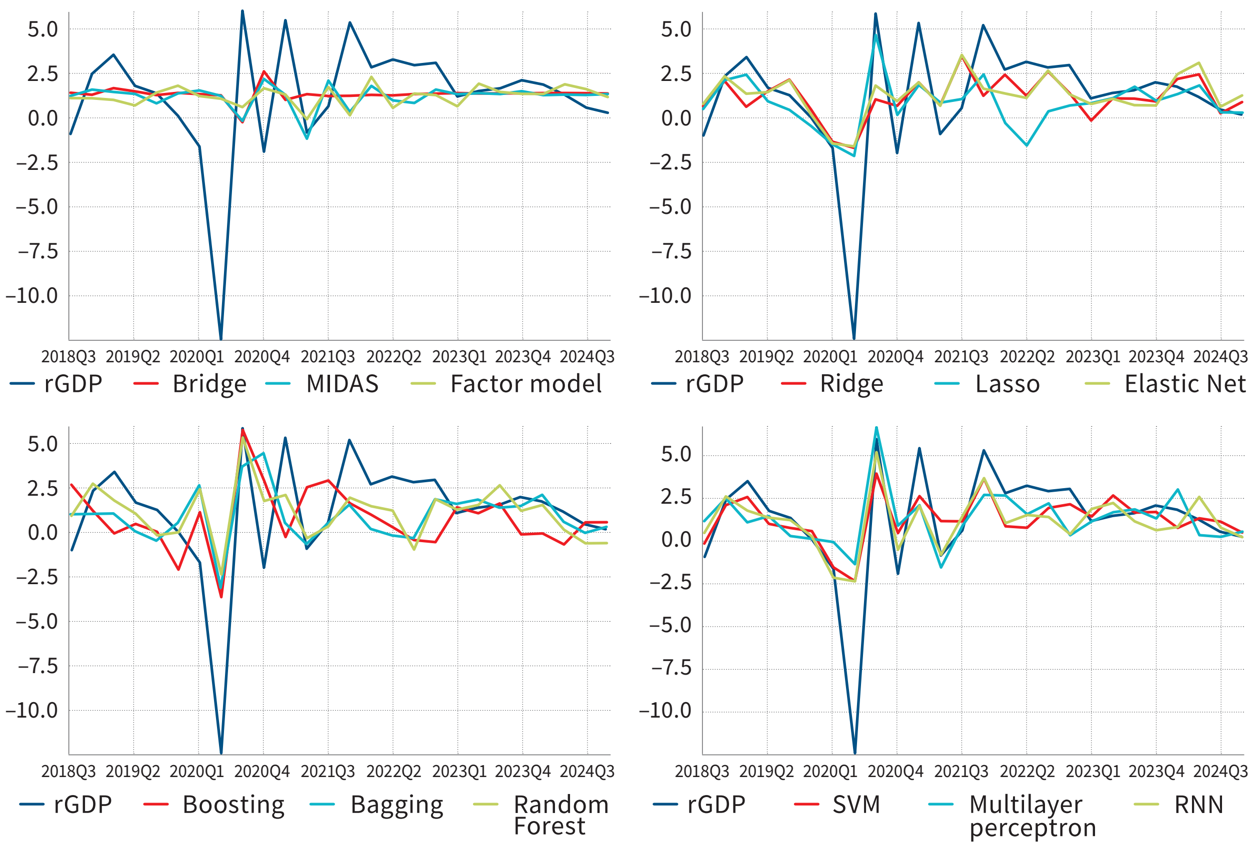


Figure 3. Nowcasting of Belarus's Real GDP Growth Rates

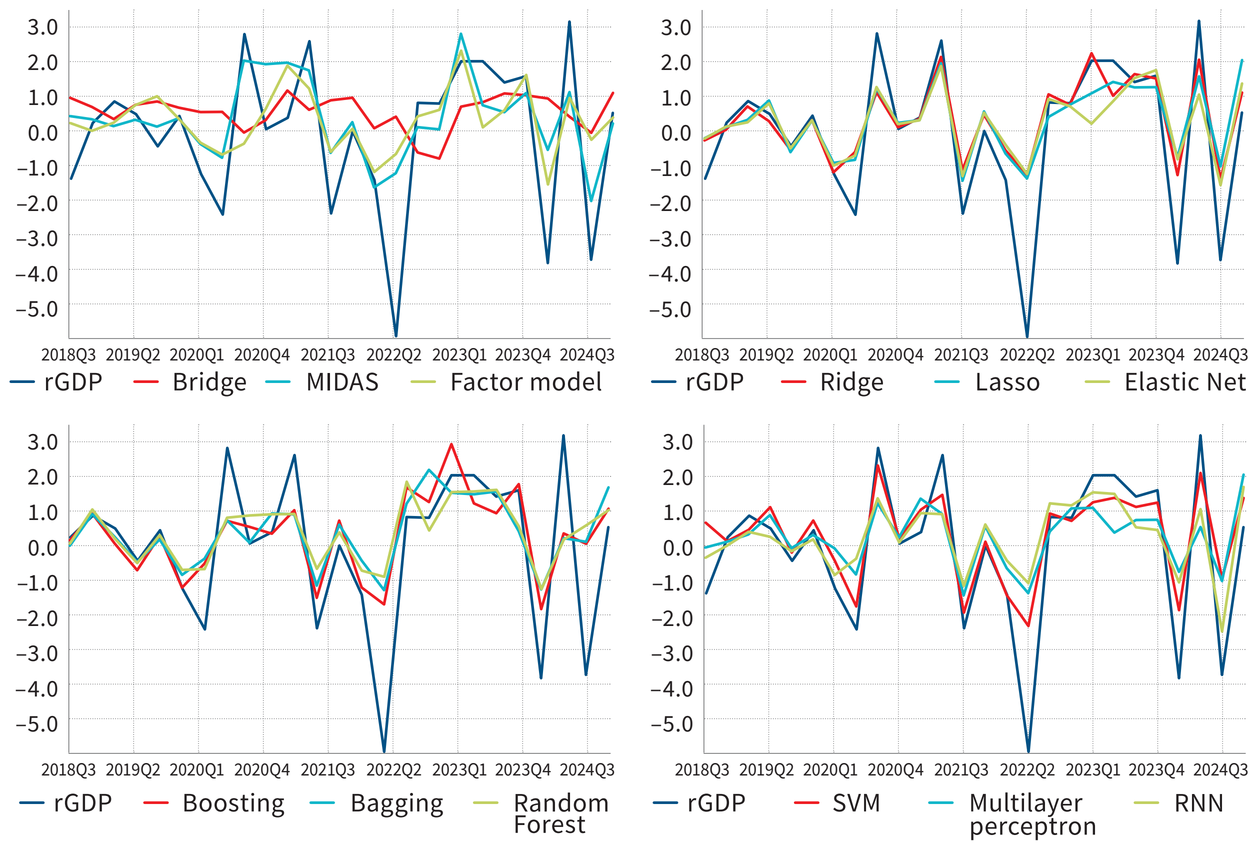


Figure 4. Nowcasting of Armenia's GDP Deflator

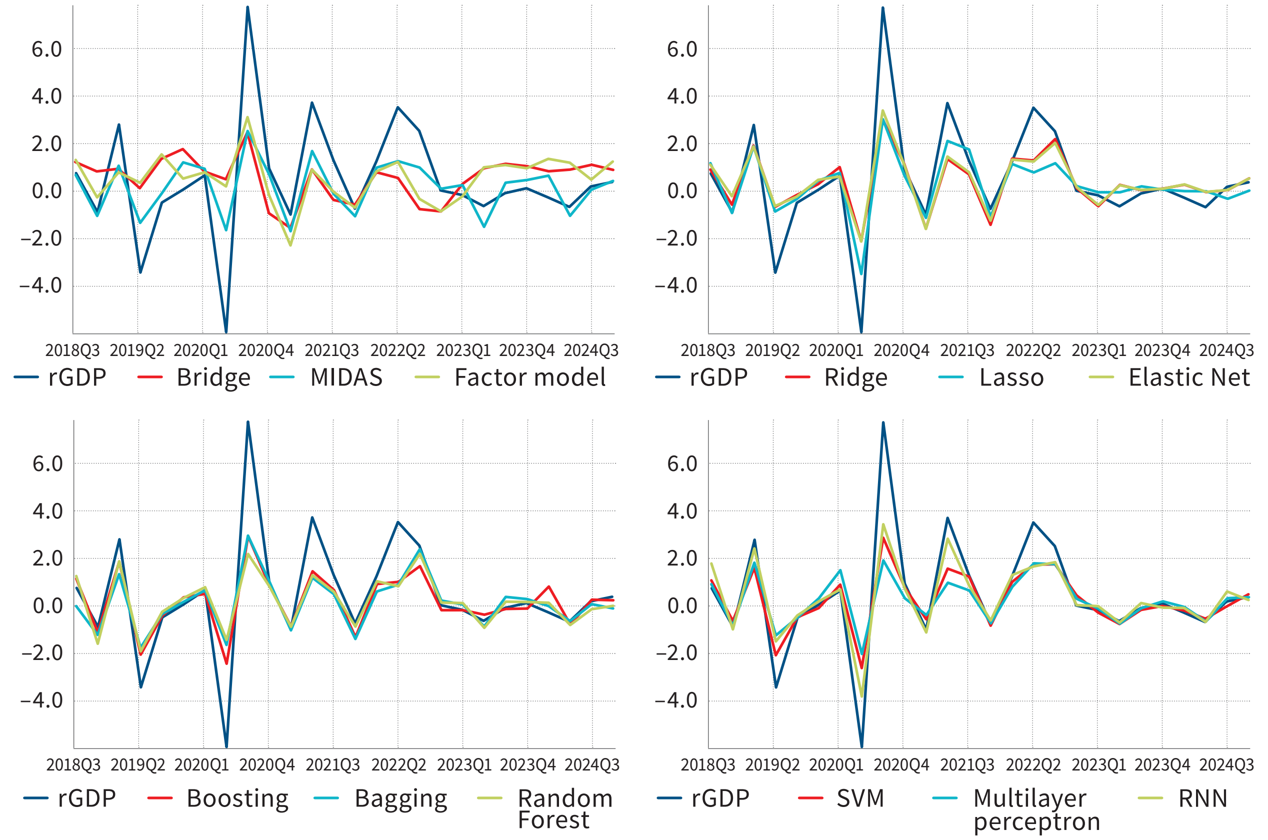
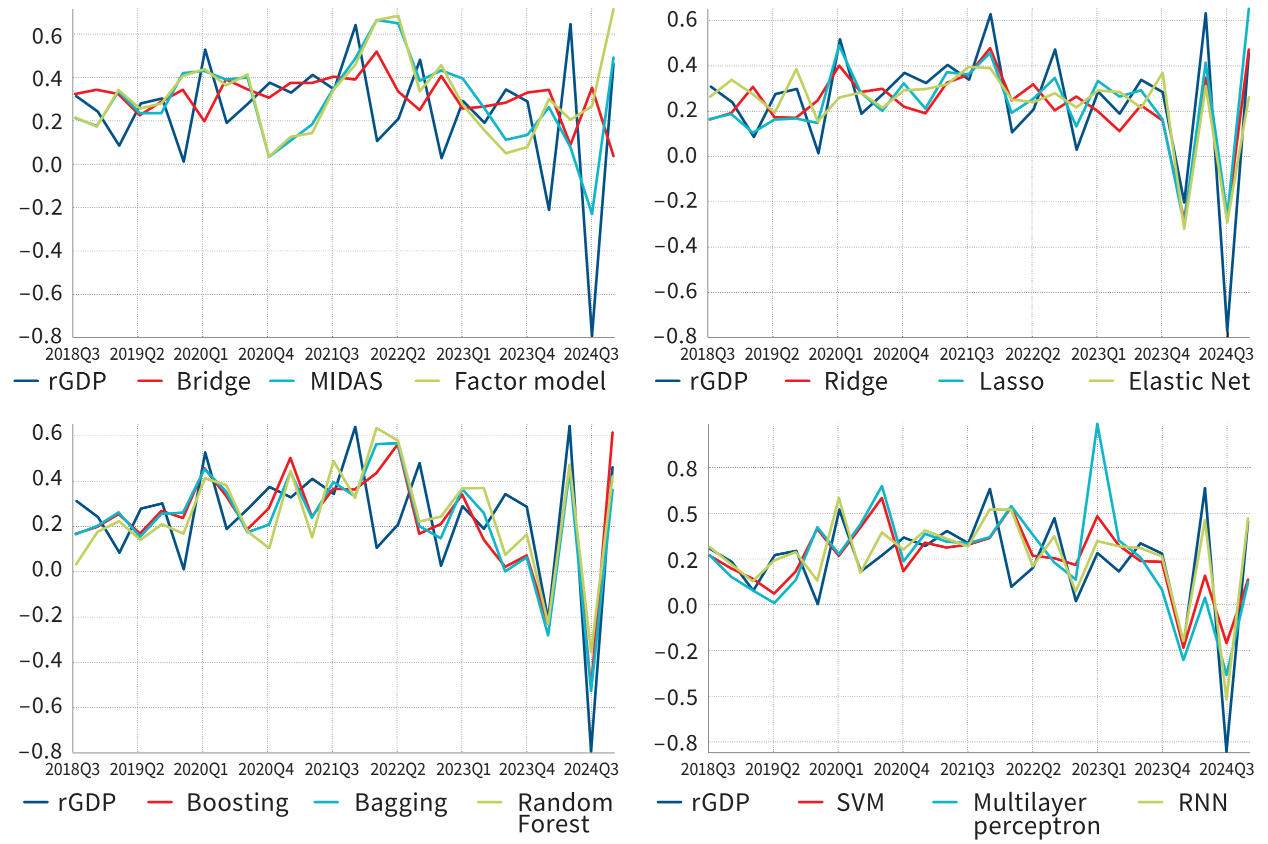


Figure 5. Nowcasting of Belarus's GDP Deflator



The response to the question "Which algorithm should be used for practical purposes?" is that the machine learning algorithm that produces the lowest RMSE and MAD value should be utilised. For example, the calculations performed show that LASSO and SVM produce the best results for the GDP growth rate, and RNN produces the best results for the GDP deflator. However, in dynamic environments, it is advisable to regularly reassess the quality of the models and switch between them as required. An alternative solution could be a combined nowcasting approach, which brings together the advantages of different methods.

Combined Nowcasting

Combined nowcasting (Richardson et al., 2018) can be performed by weighting based on:

1. Equal weights,
2. Weights obtained on the basis of OLS,
3. Inverted loss function values,
4. Inverted loss function rank values.

Weighting based on the methods described above (Tables 5–8) produced lower RMSE and MAD values than the bridge equation. The highest accuracy was observed when OLS weights were used[11].

Table 5. Combined Estimates of Armenia's Real GDP Growth Rates

Weighting method	RMSE	RMSE (relative to bridge)	MAD	MAD (relative to bridge)
Equal weights	3.32	0.93	2.15	0.98
OLS weights	2.55	0.72**	1.52	0.69**
Inverted MSE values	3.22	0.91	2.05	0.93
Inverted MSE rank values	3.10	0.87	2.01	0.91

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

Table 6. Combined Estimates of Armenia's GDP Deflator

Weighting method	RMSE	RMSE (relative to bridge)	MAD	MAD (relative to bridge)
Equal weights	2.11	0.71**	1.50	0.71**
OLS weights	1.78	0.60***	1.41	0.67***
Inverted MSE values	2.09	0.70**	1.45	0.69***
Inverted MSE rank values	2.05	0.69***	1.47	0.70***

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

11 In this study, the weights were determined using OLS and a regression was developed with no constant or parameter restrictions, in accordance with Richardson et al. (2018).

Table 7. Combined Estimates of Belarus's Real GDP Growth Rates

Weighting method	RMSE	RMSE (relative to bridge)	MAD	MAD (relative to bridge)
Equal weights	1.75	0.75^{*}	1.46	0.72^{*}
OLS weights	1.55	0.66^{**}	1.33	0.66^{**}
Inverted MSE values	1.70	0.73^{*}	1.45	0.72
Inverted MSE rank values	1.69	0.72^{*}	1.44	0.71^{*}

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

Table 8. Combined Estimates of Belarus's GDP Deflator

Weighting method	RMSE	RMSE (relative to bridge)	MAD	MAD (relative to bridge)
Equal weights	1.35	0.63^{**}	1.30	0.66^{**}
OLS weights	1.02	0.48^{***}	0.99	0.51^{***}
Inverted MSE values	1.30	0.61^{**}	1.21	0.62^{**}
Inverted MSE rank values	1.29	0.60^{**}	1.15	0.59^{**}

*, **, *** — significant at the 10%, 5%, and 1% levels, respectively.

Therefore, we drew the following conclusions:

1. Machine learning algorithms yield more accurate nowcasts than conventional econometric models.
2. The difference between the forecasts produced by LASSO, SVM, and RNN algorithms and those produced by the bridge equation is statistically significant, as demonstrated by the Diebold-Mariano statistics.
3. The best prediction is produced by the algorithm that minimises the loss function. However, as new data become available, the model choice can be adjusted depending on trends in the error metrics (RMSE and MAD).
4. Alternatively, combined nowcasting based on OLS weights can be used. The outcomes show that this approach achieves a statistically significant improvement in accuracy compared with predictions based on the bridge equation.

Conclusion

Conventional econometric models remain the main tools used for nowcasting macroeconomic indicators. However, the intensive development of technology and computing equipment has led to growing demand for machine learning methods and algorithms in economic analysis and forecasting.

This research aims to estimate the potential of machine learning methods in nowcasting. Specifically, it seeks to determine (1) whether machine learning methods can improve predictive accuracy compared with conventional approaches; and (2) whether these methods can serve as an alternative, rather than merely a complement, to conventional econometric models.

The Working Paper presents performance testing of three conventional econometric models (bridge equation, MIDAS, and a factor model) and nine machine learning methods and algorithms (ridge regression, LASSO regression, elastic net, boosting, bagging, random forest, SVM, MLP, and RNN).

The following conclusions were made:

1. Most of the machine learning methods and algorithms tested demonstrate superior predictive accuracy compared with conventional nowcasting tools, which is confirmed by lower values of the loss function.
2. The difference between the nowcasts produced by LASSO, SVM, and RNN algorithms and those produced by conventional econometric models is statistically significant, as demonstrated by the Diebold-Mariano statistics.
3. The outcome of nowcasting is significantly improved when the forecast combination approach is used. The highest accuracy is achieved when combined nowcasting based on OLS weights is used.

Therefore, machine learning methods and algorithms can be viewed as both an effective complement and an alternative to conventional econometric approaches, particularly for the purpose of nowcasting macroeconomic indicators.

References

Aastveit, K.A., Gerdrup, K.R., Jore, A.S., and Thorsrud, L.A. (2011) *Nowcasting GDP in real-time: A density combination approach*. Norges Bank, Working Paper, No. 2011/11. Available at: https://www.econstor.eu/bitstream/10419/209987/1/nb-wp2011-11.pdf.

Alvarez, R., Camacho, M. and Perez-Quiros, G. (2016) Aggregate versus disaggregate information in dynamic factor models, International Journal of Forecasting, vol. 32(3), pp. 680–694. Available at: https://doi.org/10.1016/j.ijforecast.2015.10.006.

Angelini, E., Camba-Mendez, G., Giannone, D., Reichlin, L., and Rünstler, G. (2011) *Short-term forecasts of euro area GDP growth*. Econometrics Journal, 14(1), pp. C25–C44.

Apaydın, M., Erer, D., and Karaçor, Z. (2019) *Nowcasting GDP using machine learning algorithms: A comparative analysis*. Journal of Business Research — Türk, 11(4), 2587–2599.

Baffigi, A., Golinelli, R., and Parigi, G. (2004) *Bridge models to forecast the Euro area GDP*. International Journal of Forecasting, 20(3), pp. 447–460. Available at: https://doi.org/10.1016/S0169-2070(03)00067-0.

BańBura, M., Giannone, D., and Reichlin, L. (2010) *Nowcasting*. ECB Working Paper No. 1275. Available at: http://dx.doi.org/10.2139/ssrn.1717887.

Bandara, K., Bergmeir, C., and Smyl, S. (2020) *Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach*. Expert Systems with Applications, 140, 112896. https://doi.org/10.1016/j.eswa.2019.112896.

Barhoumi, K., Darne, O., and Ferrara, L. (2010) *Are disaggregated data useful for factor analysis in forecasting French GDP?* Journal of Forecasting, 29, 132–144.

Bencivelli, L., Marcellino, M. and Moretti, G.L., (2012) *Selecting predictors by Bayesian model averaging in bridge models*. Banca d'Italia WP.

Bernanke, B.S., Boivin, J., and Eliasz, P. (2005) *Measuring the effects of monetary policy: a factor-augmented vector autoregressive (FAVAR) approach*. The Quarterly Journal of Economics, Vol. 120, No. 1, pp. 387–422.

Breiman, L. (1996) Bagging predictors, Machine Learning, 26, pp. 123–140.

Breiman, L. (2001) Random Forests. Machine Learning, 45, pp. 5–32. Available at: https://doi.org/10.1023/A:1010933404324.

Breiman, L. and Ihaka, R. (1984) Nonlinear discriminant analysis via scaling and ACE. Technical Report, University of California Berkeley.

Chernis, T., and Sekkel, R. (2017) *A dynamic factor model for nowcasting Canadian GDP growth*. Bank of Canada, Staff Working Paper 2017-2. DOI: https://doi.org/10.34989/swp-2017-2.

Chong, E.K.P. and Zak, S.H. (2013) *An Introduction to Optimization*. 4th Edition, John Wiley & Sons, New York.

Clements, M., and Galvao, A. (2008) *Macroeconomic Forecasting with mixed frequency data: Forecasting output growth in the United States*. Journal of Business and Economic Statistics, 26, 546–554.

Cortes, C., Vapnik, V. (1995) *Support-vector networks*. Machine Learning, 20, 273–297. Available at: https://doi.org/10.1007/BF00994018.

Coulombe, P. G., Leroux, M., Stevanovic, D., and Surprenant, S. (2020) *How is machine learning useful for macroeconomic forecasting?* Journal of Applied Econometrics, 37(5), 847–866. Available at: https://doi.org/10.1002/jae.2884.

Desai, A. (2023) Machine learning for economics research: when, what and how. Bank of Canada, Staff Analytical Note 2023-16. Available at: https://www.bankofcanada.ca/2023/10/staff-analytical-note-2023-16/.

Diebold, F. X. and Mariano, R. S. (1995) Comparing Predictive Accuracy. Journal of Business & Economic Statistics 13(3), pp. 253–263.

Diron, M. (2008) *Short-term forecasts of Euro Area real GDP growth. An assessment of real-time performance based on vintage data*. Journal of Forecasting, 27(5), 371–390.

Doz, C., Giannone, D., and Reichlin, L. (2011) *A two-step estimator for large approximate dynamic factor models based on Kalman filtering*. Journal of Econometrics, 164(1), 188–205. Available at: https://doi.org/10.1016/j.jeconom.2011.02.012.

Efron, B. and Tibshirani, R. (1993) An introduction to the Bootstrap. Chapman and Hall, London.

Fischer, T., and Krauss, C. (2018) *Deep learning with long short-term memory networks for financial market predictions*. European Journal of Operational Research, 270(2), 654–669. DOI: 10.1016/j.ejor.2017.11.054.

Foroni, C., and Marcellino, M. (2013) *A survey of econometric methods for mixed-frequency data*. Norges Bank Working Paper 2013/06. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2268912.

Freund, Y. and Shapire, R. (1997) A decision-theoretic generalization of online learning and an application to boosting. Journal of Computer and System Sciences, 55, pp. 119–139.

Ghysels, E., P. Santa-Clara, and Valkanov, R. (2004) *The MIDAS touch: MIxed DAta Sampling regression models*. Mimeo, Chapel Hill, N.C. Available at: https://escholarship.org/uc/item/9mf223rs.

Ghysels, E., Sinko, A., and Valkanov, R. (2007) *MIDAS Regressions: Further Results and New Directions*. Econometric Reviews, 26(1), pp. 53–90. Available at: https://doi.org/10.1080/07474930600972467

Giannone, D., Agrippino, S.M., and Modugno, M. (2013) *Nowcasting China Real GDP*. Working Paper. Available at: https://cirano.qc.ca/conferences/public/pdf/realtime2013/Papers/Giannone_Miranda.pdf.

Giannone, D., Reichlin, L., and Small, D. (2008) *Nowcasting: The real-time informational content of macroeconomic data*. Journal of Monetary Economics, Vol. 55, No. 4, pp. 665–676.

Hastie, T., Tibshirani, R. & Friedman, J. (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.). Stanford, CA: Stanford University.

Hoerl, A. E. and Kennard, R. (1970) Ridge regression: biased estimation for nonorthogonal problems, Technometrics, 12, pp. 55–67.

Huang, W., Nakamori, Y., and Wang, S.Y. (2005) *Forecasting stock market movement direction with support vector machine*. Computers & Operations Research, 32(10), 2513–2522. Available at: https://doi.org/10.1016/j.cor.2004.03.016.

Kapetanios, G., and Papailias, F. (2018) *Big Data & Macroeconomic Nowcasting: Methodological Review*. ESCoE Discussion Paper 2018-12. Available at: https://escoe-website.s3.amazonaws.com/wp-content/uploads/2020/07/13161005/ESCoE-DP-2018-12.pdf.

Kim, H.Y. (2003) *Financial time series forecasting using support vector machines*. Neurocomputing, 55(1–2), 307–319. Available at: https://doi.org/10.1016/S0925-2312(03)00372-2.

Linzenich, J., and Meunier, B. (2024) *Nowcasting Made Easier: A Toolbox for Economists*. ECB Working Paper No. 2024/3004. Available at: http://dx.doi.org/10.2139/ssrn.5060436

Malyugin, V. (2024) *Kratkosrochnoye prognozirovaniye i naukasting tempov rosta inflyatsii na osnove modeley po smeshannym dannym*. Bankovsky vestnik, No. 1/726. Available at: https://www.nbrb.by/bv/pdf/articles/12119.pdf.

Marcellino, M., and Schumacher, C. (2010) *Factor-MIDAS for now- and forecasting with ragged-edge data: A model comparison for German GDP*. Oxford Bulletin of Economics and Statistics, 72, pp. 518–550.

Mariano, R.S., and Murasawa, Y. (2003) *A new coincident index of business cycles based on monthly and quarterly series*. Journal of Applied Econometrics, 18(4), pp. 427–443.

Matheson, T.D. (2010) An analysis of the informational content of New Zealand data releases: the importance of business opinion surveys. Economic Modelling, 27, 304–314.

Medeiros, M., Gabriel, F.R., Vasconcelos, G., Veiga A., and Zilberman, E. (2021) *Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods*. Journal of Business & Economic Statistics, vol. 39(1), pp. 98–119. Available at: https://doi.org/10.1080/07350015.2019.1697174.

Poghosyan, K. and Magnus, J. R. (2012) WALS Estimation and Forecasting in Factor-based Dynamic Models with an Application to Armenia, International Econometric Review, Econometric Research Association, vol. 4(1), pp. 40–58. Available at: https://ideas.repec.org/a/erh/journl/v4y2012i1p40-58.html.

Richardson, A., Florenstein Mulder, T., Vehbi, T. (2018) Nowcasting New Zealand GDP using machine learning algorithms. 2018-47, Centre for Applied Macroeconomic Analysis, Crawford School of Public Policy, the Australian National University. Available at: https://ideas.repec.org/s/een/camaaa.html.

Richardson, A., van Florenstein Mulder, T., and Vehbi, T. (2021) *Nowcasting GDP using machine-learning algorithms: A real-time assessment*. International Journal of Forecasting, vol. 37(2), pp. 941–948. Available at: https://doi.org/10.1016/j.ijforecast.2020.10.005.

Rossi, B., and Sekhposyan, T. (2019) *Alternative forecasting approaches.* In: *Handbook of Economic Forecasting*, Volume 2B, edited by G. Elliott & A. Timmermann, Elsevier, pp. 967–1046.

Sharma, M., Kathuria, V. (2024) *Macroeconomic Nowcasting: What can Central Banks Learn from a Structured Literature Review?* Journal of Quantitative Economics. Available at: https://doi.org/10.1007/s40953-024-00421-x.

Tay, F. E. H., and Cao, L. (2001) *Application of support vector machines in financial time series forecasting*. Omega, 29(4), 309–317. Available at: https://doi.org/10.1016/S0305-0483(01)00026-3.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society, Series B, 58, pp. 267–288.

Tsukarev, T., Poghosyan, K., and Lemba, K. (2024) Machine Learning Algorithms for Short-Term Forecasting of Real GDP Growth Rates. EFSD Working Paper No.5. Available at: https://efsd.org/upload/iblock/a23/0kmlpkr5cgsf55lnkuvyzlfuneh4n7bm/EFSD_WP_ML_EN_2024.pdf.

Zhang, Q., Ni, H., Xu, H. (2023) *Nowcasting Chinese GDP in a data-rich environment: Lessons from machine learning algorithms*. Economic Modelling, Volume 122, https://doi.org/10.1016/j.econmod.2023.106204.

Zhemkov, M. (2021) *Nowcasting Russian GDP using a forecast combination approach*. Bank of Russia, Series of Economic Research Papers. Available at: https://www.cbr.ru/StaticHtml/ File/118608/wp-feb21_e.pdf.

Annex 1

List of Armenia's Macroeconomic Indicators

Indicator	Source	Transformation	Seasonal adjustment
Industrial production (constant 2017)	Armstat	*ln*, Δ	Yes
Construction (constant 2017)	Armstat	*ln*, Δ	Yes
Retail trade (constant 2017)	Armstat	*ln*, Δ	Yes
Exports of goods	Armstat	*ln*, Δ	Yes
Imports of goods	Armstat	*ln*, Δ	Yes
Industrial producer price index, % (2010 = 100)	Armstat	*ln*, Δ	Yes
Construction price index, % (2010 = 100)	Armstat	*ln*, Δ	Yes
Consumer price index, % (2010 = 100)	CBA	*ln*, Δ	Yes
AMD/USD exchange rate, period average	CBA	*ln*, Δ	Yes
Base money (end of period)	CBA	*ln*, Δ	Yes
Cash outside CBA (end of period)	CBA	*ln*, Δ	Yes
Cash in circulation	CBA	*ln*, Δ	Yes
Monetary aggregate M1	CBA	*ln*, Δ	Yes
Monetary aggregate M2	CBA	*ln*, Δ	Yes
Monetary aggregate M2X	CBA	*ln*, Δ	Yes
Loans of commercial banks to residents, total (end of period)	CBA	*ln*, Δ	Yes
Loans of commercial banks to resident enterprises, total (end of period)	CBA	*ln*, Δ	Yes
Loans of commercial banks to households, total (end of period)	CBA	*ln*, Δ	Yes
Average deposit interest rate (up to 1 year), local currency	CBA	Δ	No
Average deposit interest rate (up to 1 year), foreign currency	CBA	Δ	No
Average lending interest rate (up to 1 year), local currency	CBA	Δ	No
Average lending interest rate (up to 1 year), foreign currency	CBA	Δ	No

Note: *ln* denotes taking the logarithm of the indicator, Δ is the first differencing.

Annex 2

List of Belarus's Macroeconomic Indicators

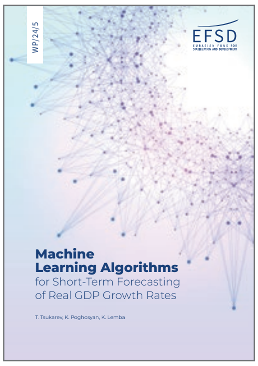
Indicator	Source	Transformation	Seasonal adjustment
Industrial production (constant 2018)	NBRB	*ln*, Δ	Yes
Fixed capital formation (constant 2018)	NBRB	*ln*, Δ	Yes
Retail trade (constant 2018)	NBRB	*ln*, Δ	Yes
Consumer price index, % of previous month	Belstat	*ln*, Δ	Yes
Industrial producer price index, % of previous month	Belstat	*ln*, Δ	No
Agricultural producer price index, % of previous month	Belstat	*ln*, Δ	Yes
Construction price index, % of previous month	Belstat	*ln*, Δ	Yes
Exports of goods and services	Belstat	*ln*, Δ	Yes
Imports of goods and services	Belstat	*ln*, Δ	Yes
Gross international reserves	NBRB	*ln*, Δ	No
BYN/USD nominal exchange rate	NBRB	*ln*, Δ	No
Base money (monthly average)	NBRB	*ln*, Δ	Yes
Broad money supply (monthly average)	NBRB	*ln*, Δ	Yes
Deposits of state-owned commercial enterprises	NBRB	*ln*, Δ	Yes
Private sector deposits	NBRB	*ln*, Δ	Yes
Deposits of households	NBRB	*ln*, Δ	Yes
Bank loans, public form of ownership	NBRB	*ln*, Δ	Yes
Bank loans, private form of ownership	NBRB	*ln*, Δ	Yes
Average interest rate on new deposits, local currency	NBRB	Δ	No
Average interest rate on new loans (excluding interbank loans), local currency	NBRB	Δ	No

Note: *ln* denotes taking the logarithm of the indicator, Δ is the first differencing.

Report on sovereign financing for 2024
(RU/EN)

Sovereign Financing in Eurasia in 2024: Record Amounts of IFIs Support

The Report focuses on the monitoring of sovereign financing in Eurasia for 2024, relying on a database maintained by the EFSD.



Working paper WP/24/5
(RU/EN)

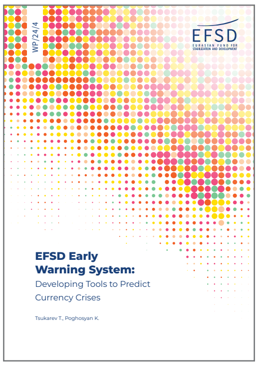
Machine Learning Algorithms for Short-Term Forecasting of Real GDP Growth Rates

The paper evaluates the accuracy of short-term forecasts produced by machine learning methods and algorithms compared to conventional econometric models.

Joint Working Paper by the EFSD and the CAREC Institute
(RU/EN)

Country-level interest rate risk impact on debt and fiscal sustainability: potential use of floating-rate and inflation-indexed liabilities

The Paper examines the interest rate risks associated with recent dynamic of LIBOR (SOFR) and EURIBOR, as well as considers the potential inclusion of obligations with floating-rate and indexed principal in domestic debt portfolios of the countries under review.



Working paper WP/24/4
(RU/EN)

EFSD Early Warning System: Developing Tools to Predict Currency Crises

The paper presents a methodology and a step-by-step algorithm for the development of tools to identify imbalances (crises) and stress situations in the economy. The main emphasis was placed on detection of growing tensions in the foreign exchange market.

Working paper WP/24/3
(RU/EN)

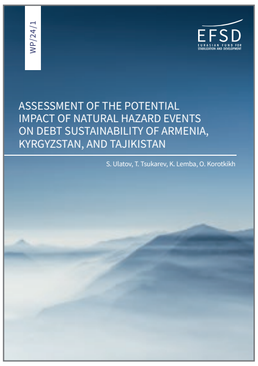
Sovereign financing in Eurasia: 1H 2024.

The Paper focuses on the monitoring of sovereign financing in Eurasia for 1H 2023.

Working paper WP/24/2
(RU/EN)

Sovereign financing in Eurasia: trends and areas

The Report focuses on the monitoring of sovereign financing in Eurasia, relying on a database maintained by the EFSD.



Working paper WP/24/1
(RU/EN)

Assessment of the Potential Impact of Natural Hazard Events on Debt Sustainability of Armenia, Kyrgyzstan, and Tajikistan

The paper presents an algorithm that can be used to assess the impact of natural hazards on macroeconomic indicators and debt sustainability of various countries.

Working paper WP/23/3
(RU/EN)

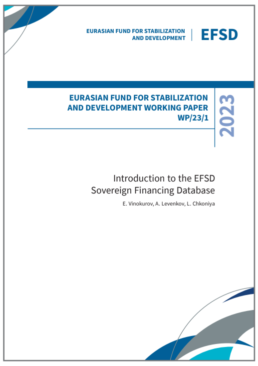
International Reserves as the core element of the GFSN for developing economies

The paper assesses factors affecting the decision of developing economies on the source of anti-crisis support. The study showed that international reserves are the most sought-after instrument among all the elements of the GFSN.

Working paper WP/23/2
(RU/EN)

Sovereign Financing in Eurasia: Water Sector and Hydropower Generation

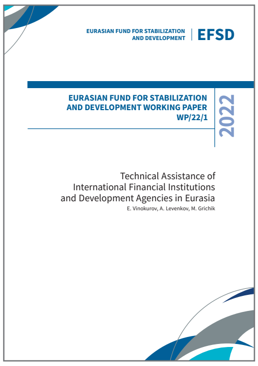
The purpose of this Working Paper is to analyse operations of IFIs, climate funds, and development agencies in the water and HPP sector between 2008 and H1 2023 in 11 countries of the Eurasian region.



Working Paper WP/23/1
(RU/EN)

Introduction to the EFSD Sovereign Financing Database.

In this Working Paper the Sovereign Financing Database (SFD) Methodology is presented and also quantitative and qualitative analysis of sovereign financing operations in 11 countries of the region from 2008 to 2022 is carried out.



Working Paper WP/22/1
(RU/EN)

Technical Assistance of International Financial Institutions and Development Agencies in Eurasia.

The purpose of this analytical document is to review technical assistance projects implemented by international financial institutions and development agencies in 2009–2021 in 11 Eurasian countries with a detailed thematic and institutional breakdown.

GDP Nowcasting: From Traditional Econometric Models to Machine Learning Algorithms

T. Tsukarev, K. Poghosyan, K. Lemba, D. Grishin

The Eurasian Fund for Stabilization and Development (EFSD) totaling over US$ 9 billion was established on June 9th, 2009 by the governments of the Republic of Armenia, the Republic of Belarus, the Republic of Kazakhstan, the Kyrgyz Republic, the Russian Federation, and the Republic of Tajikistan. The objectives of the EFSD are to assist its member countries in overcoming the consequences of the global financial crisis, ensure their economic and financial stability, and foster integration in the region. More information about the EFSD is available at: efsd.org/en/.

EFSD Working Papers and Reports are the main format of the Fund's public research. They reflect the Fund's research on global, regional, and country economic trends, economic modelling, macroeconomic analysis, sectoral analysis, global financial architecture, and other issues. EFSD publications are available at: efsd.org/en/research/.

Address:
Moscow
Chistoprudny Boulevard, 17 b. 1
101000, Russian Federation
Tel: +7 (495) 645 04 45
Fax: +7 (495) 645 04 41
Web: efsd.org/en/