GENEVA GRADUATE INSTITUTE

INSTITUT DE HAUTES ÉTUDES INTERNATIONALES ET DU DÉVELOPPEMENT

GRADUATE INSTITUTE OF INTERNATIONAL AND DEVELOPMENT STUDIES

Graduate Institute of International and Development Studies

International Economics Department

Working Paper Series

Working Paper No. HEIDWP19-2026

Macroeconomic Forecasting Using Machine Learning Methods: An Application to Uzbekistan

Abdukakhkhor Abdurakhmonov
Central Bank of Uzbekistan

Chemin Eugène-Rigot 2
P.O. Box 136
CH - 1211 Geneva 21
Switzerland

BCC Bilateral Assistance & Capacity Building for Central Banks

Macroeconomic Forecasting Using Machine Learning Methods: An Application to Uzbekistan

Abdukakhkhor Abdurakhmonov
a.abdukakhkhor@gmail.com
Central Bank of Uzbekistan

Abstract

This paper provides the first systematic assessment of machine learning methods for macroeconomic forecasting in Uzbekistan. Using a comprehensive dataset of more than 170 indicators, we forecast CPI inflation and GDP growth with nine machine learning models and compare them against three traditional benchmarks (ARIMA, VAR, and BVAR). For both targets, the relative performance of machine learning improves as the forecast horizon increases. For inflation, machine learning provides clear and growing gains as the horizon increases, and a simple equal-weighted ensemble of the machine learning models is the most accurate approach overall, achieving the lowest forecast error at nearly every horizon. For GDP growth, by contrast, the traditional benchmarks (ARIMA in particular) remain the most accurate across most horizons, although regularized linear and dimension-reduction machine learning methods are competitive at short horizons. Tree-based models struggle to forecast GDP when growth exceeds the range observed during training because they cannot extrapolate beyond the training data. This limitation is particularly relevant in Uzbekistan's rapidly changing economy, where rapid economic growth in 2024-2025 pushed the level of GDP beyond the range observed in the training sample. We show that forecasting stationary transformations of the target largely removes this weakness. Overall, the findings suggest that machine learning is best used to complement rather than replace the existing forecasting toolkit. It improves the accuracy of medium-term inflation forecasts, whereas traditional models remain more accurate for forecasting GDP.

Keywords: Machine Learning, Macroeconomic Forecasting, Inflation, GDP Growth, Model Evaluation and Selection, Uzbekistan
JEL: (C22, C45, C53, E31, E37, E52)

The author thanks Professor Christophe Gaillac from the University of Geneva and Professor Marko Mlikota from the Geneva Graduate Institute for their academic supervision of this paper. This research took place through the visiting program under the Bilateral Assistance and Capacity Building for Central Banks (BCC), financed by SECO, and the Graduate Institute in Geneva.
The views expressed in this paper are solely those of the author and do not necessarily reflect those of the Central Bank of Uzbekistan.

1 Introduction

Forecasting macroeconomic variables such as inflation and GDP is central to monetary policymaking, and it is particularly challenging in an economy where relationships among macroeconomic variables are more volatile and less predictable. Traditional econometric approaches, including autoregressive and other linear models, often perform well under relatively stable economic conditions but struggle to capture structural change, time-varying dynamics, and nonlinear relationships among variables (Stock and Watson, 2007; Medeiros and Vasconcelos, 2019; Liu, 2024). Moreover, these models typically rely on a limited set of predictors and are not well suited to utilizing high-dimensional datasets.

In contrast, machine learning (ML) methods have demonstrated a strong ability to capture complex and nonlinear relationships that conventional econometric approaches often miss. And ML techniques can process large datasets, adapt to structural changes, and significantly reduce forecast errors (Richardson et al., 2018; Babii et al., 2020; Flores et al., 2025).

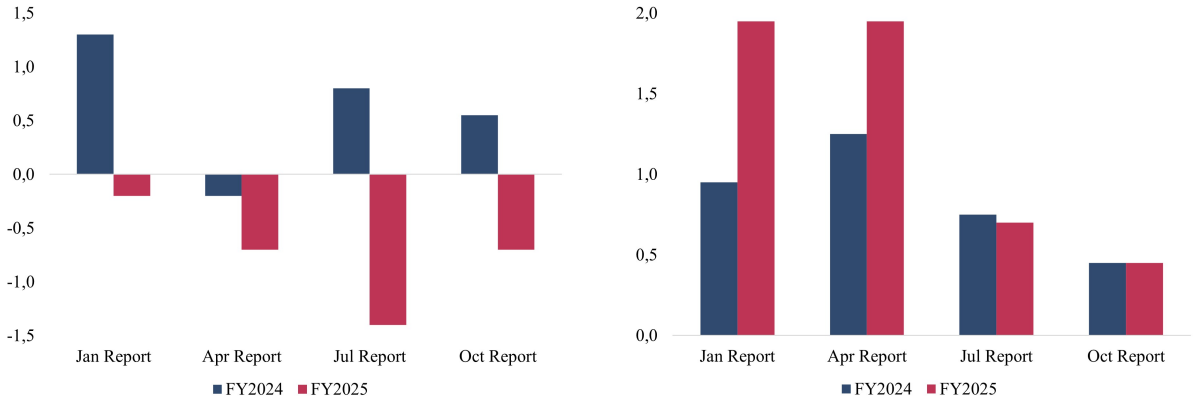
The existing literature on ML has largely focused on nowcasting and short-term forecasting, particularly for advanced economies, while studies covering longer forecast horizons and developing economies remain limited. To the best of our knowledge, no study has yet applied machine learning methods to macroeconomic forecasting in Uzbekistan.

Uzbekistan's economy has undergone substantial structural changes and large external shocks since 2017, making the relationships among macroeconomic variables increasingly unstable and nonlinear. These conditions make it difficult for traditional models to produce reliable forecasts. Inflation and GDP growth have been highly volatile and have experienced significant shifts over the past decade (see Appendix A for the historical evolution of inflation and GDP), further illustrating the challenge of accurate forecasting.

At the same time, forecasting performance at the Central Bank of Uzbekistan (CBU) has been deteriorating due to data quality issues, structural changes in the economy, and methodological limitations of traditional approaches. As shown in Figure 1, the CBU's headline inflation forecasts have been consistently biased, switching between downward and upward directions, with inflation in 2024 and 2025 being largely missed. GDP forecasts, on the other hand, have been persistently biased downward, with actual GDP growth consistently exceeding the central bank's projections. These challenges highlight the potential of machine learning methods, which are better equipped to capture unstable and nonlinear relationships in the data.

From a policy perspective, the adoption of ML methods has already been prioritized by CBU management to enhance the accuracy and robustness of macroeconomic forecasts. This commitment is also reflected in the Monetary Policy Guidelines, which explicitly emphasize the development and implementation of ML techniques in forecasting (CBU, 2024). Therefore, advancing research in this area is both scientifically relevant and institutionally important.

Taking the above into account, this study contributes to the existing literature on machine learning-based macroeconomic forecasting by addressing three key challenges: (i) it examines whether machine learning methods can improve the accuracy and reliability of short- to medium-term forecasts of inflation and GDP compared to traditional econo-



(a) Inflation **(b) GDP growth**

Figure 1: Forecast Errors of the Central Bank of Uzbekistan

Source: Author's calculations based on Monetary Policy Reports of the Central Bank of Uzbekistan.
Note: Forecast error is defined as the difference between the actual and forecasted value. For each fiscal year, forecasts are taken from the January, April, July, and October reports (MPR), corresponding to Q4 of the previous year and Q1, Q2, Q3 of the forecast year, respectively.

metric models in developing economy such as Uzbekistan; (ii) it assesses which classes of models and which modeling choices, including feature selection, target transformation, and forecast combination, are most effective for capturing inflation and GDP growth dynamics in such economies; and (iii) it discusses how ML-based forecasts can complement the existing forecasting framework of the Central Bank of Uzbekistan.

The remainder of this paper is structured as follows. Section 2 reviews the relevant literature on inflation and GDP forecasting and ML applications. Section 3 describes the data and preprocessing steps, including variable selection and data transformations. Section 4 outlines the models employed, and Section 5 presents the estimation strategy, covering sample splits, model configurations, and hyperparameter optimization. Section 6 presents the results, comparing model performance across different forecast horizons and methodological approaches. Finally, Section 7 concludes and discusses implications for monetary policy and future research.

2 Literature review

The literature on macroeconomic forecasting is extensive and has evolved significantly over the past few decades. Traditional forecasting frameworks have long relied on autoregressive models, ranging from univariate ARIMA specifications to multivariate Vector Autoregressions (VARs). These models have served as the workhorse tools of macroeconomic forecasting due to their simplicity, interpretability, and solid theoretical foundations (Sims, 1980; Stock and Watson, 2001; Lütkepohl, 2005; Zhang et al., 2020; George E.P. Box et al., 2016).

However, these models face well-known limitations in high-dimensional settings, where the number of predictors is large relative to the available observations (Bańbura et al., 2010). To address this, Bayesian Vector Autoregressions (BVARs) were introduced as a more flexible alternative. A growing body of literature has demonstrated that BVARs with Minnesota-type priors can effectively handle high-dimensional macroeconomic data through informative prior specifications (Bańbura et al., 2010; Koop, 2013; Bańbura et al., 2014). Large BVARs have been shown to deliver accurate point and density forecasts while remaining computationally tractable, providing a strong benchmark for forecasting in high-dimensional settings.

This literature motivates our choice of benchmark models. We use ARIMA and VAR as conventional benchmarks, whereas the Minnesota-prior BVAR provides a high-dimensional linear benchmark based on the same set of predictors used by our machine learning models. This allows us to assess not only whether ML outperforms conventional linear models, but also whether it outperforms the strongest high-dimensional linear benchmark.

With the growing availability of large, high-dimensional datasets, economics is becoming increasingly data-driven. The emergence of big data provides economists and policymakers with access to much larger and more diverse datasets than ever before, creating new opportunities for research and policy (Reichlin et al., 2017). Economists have long believed that forecasts can be improved by incorporating information from many economic indicators rather than relying on a limited set of variables. As Arthur F. Burns and Wesley C. Mitchell (1946) noted, "Only by analyzing numerous time series, each of restricted significance, can business cycles be made to reveal themselves definitely enough to permit close observation."

This insight remains equally relevant today: no single indicator fully captures the complex dynamics of inflation and output growth, particularly in a small open economy undergoing rapid structural transformation such as Uzbekistan. Modern machine learning methods improve forecasting by simultaneously analyzing a large number of economic and financial indicators. Since each indicator captures only a partial picture of the economy, machine learning combines information across all of them to identify meaningful patterns and generate more accurate forecasts.

In recent years, machine learning (ML) methods have gained significant attention in economic analysis and forecasting (Goulet Coulombe et al., 2022). Unlike traditional approaches, ML methods can effectively handle high-dimensional datasets even when the number of predictors far exceeds the number of observations and are capable of capturing complex, nonlinear relationships among variables (Haris Karagiannakis, 2025; Kotchoni et al., 2019).

Empirical studies demonstrate that ML models often outperform traditional econometric approaches such as ARIMA, VAR, and Bayesian models in forecast accuracy (Maehashi and Shintani, 2020). For instance, Richardson et al. (2018), Medeiros and Vasconcelos (2019), and Babii et al. (2020) show that algorithms such as random forests, gradient boosting, and neural networks can significantly reduce forecasting errors for GDP and inflation compared to conventional methods.

Among the ML methods used in macroeconomic forecasting, linear models such as

Ridge, Lasso, and Elastic Net, and tree-based nonlinear models such as Random Forest, XGBoost, and Gradient Boosting are the most widely used in practice. This is also confirmed by Slimani and Chourabi (2025), who analyzed more than 96 articles and found that Random Forest, XGBoost, Lasso, Elastic Net, Ridge Regression, and Support Vector Regression (SVR) are the most commonly applied methods. The review also finds that the literature has largely focused on short-term forecasting (ranging from 1 to 12 months), and that linear ML models tend to perform better at shorter horizons, while nonlinear models show an advantage at longer horizons. Overall, ML models tend to outperform traditional methods, though their relative performance varies depending on the forecast horizon, data structure, and underlying economic dynamics.

Our paper adopts the same core set of methods but extends the existing literature in several important ways. Whereas previous studies typically compare only a few ML models with a single benchmark for one target variable, we evaluate nine ML models against three benchmark models for two target variables (inflation and GDP) across multiple forecast horizons. This comprehensive framework allows us to test directly whether the short-horizon linear versus long-horizon nonlinear pattern reported in the literature also holds for countries like Uzbekistan.

Goulet Coulombe et al. (2022) argues that the forecasting literature has largely focused on matching specific variables and horizons with the best-performing algorithm, while paying less attention to data transformation. The author constructs numerous information sets based on different transformed versions of a large set of predictors and evaluates their contributions to forecast accuracy across several monthly macroeconomic targets. The study concludes that data transformations matter significantly, including standard transformations such as levels and stationarity, as well as two newly proposed methods designed to compress information within lag polynomials.

This is an important insight, but this literature has mainly focused on transforming all predictors in the estimation, rather than the target itself. Following this insight, we extend the transformation question to the target variable, focusing on tree-based models for GDP. Specifically, we compare level and stationary data to assess whether the transformation sensitivity documented in the literature also explains the poor performance of tree-based models in the case of Uzbekistan.

It is also worth noting that ML methods are sensitive to regularization and validation. A properly regularized and validated model can produce accurate and reliable forecasts. In fact, well-regularized linear models such as Ridge, Lasso, and Elastic Net can sometimes outperform more complex nonlinear models such as Random Forest and XGBoost (Chi et al., 2025).

Using a large panel of Japanese macroeconomic series, Maehashi and Shintani (2020) find that machine learning methods outperform conventional models, particularly at medium and longer forecast horizons. The study also finds that allowing for nonlinearity and interactions between variables further enhances forecast performance.

While these studies highlight the potential of ML-based forecasting, most of the existing literature focuses on advanced or data-rich economies. There is still limited empirical evidence on the application and performance of ML techniques in emerging and transition economies, where data constraints, structural breaks, and frequent policy shifts are more

common. Moreover, only a few studies have explored how ML-based forecasts can be integrated into central bank forecasting and policy frameworks, particularly in the context of economies undergoing structural transformation, like Uzbekistan.

This is the gap our paper fills directly. To the best of our knowledge, no study has yet applied machine learning methods to macroeconomic forecasting in Uzbekistan. This paper contributes to the literature in three important ways. First, it provides the first empirical assessments of machine learning methods for macroeconomic forecasting in Uzbekistan, using both traditional and high-frequency data sources. Second, it evaluates the comparative advantages of different ML methods relative to conventional econometric models such as ARIMA, VAR, and BVAR. Finally, it develops a practical framework for incorporating ML-based forecasting tools into the Central Bank of Uzbekistan's FPAS system.

3 Data

The dataset used for model estimation includes a wide range of variables at different frequencies, from high-frequency to low-frequency data. In total, more than 170 macroeconomic, financial, and other indicators are analyzed and included in the estimation. Some feature engineering is also applied to construct certain variables in the dataset. General information about the variables and their sources is provided in Table 1 (for the full list of variables, see Appendix I).

The dataset incorporates a broad range of domestic, international, financial, and alternative indicators, including Google Trends data, survey-based indicators, and web-scraped data on job postings. This allows the models to capture short-term fluctuations and long-term trends in economic activity, providing a robust foundation for forecasting.

All variables include data through the end of 2025. Since inflation and GDP are forecasted at different frequencies, that's monthly and quarterly, respectively, the datasets for these two variables are constructed and adjusted accordingly.

For CPI, monthly data is used, covering the period from 2018M1 to 2025M12. Some quarterly variables in the CPI dataset are converted to monthly frequency using linear interpolation. For GDP, quarterly data is used, covering the period from 2012Q1 to 2025Q4. Some monthly variables are converted to quarterly frequency by taking either the simple average or the sum over the three months in each quarter, depending on the characteristics of the variable. Variables with insufficient historical observations or substantial missing data are excluded from the estimation to maintain the reliability of the empirical analysis.

Table 1: Overview of data used in the model estimation

Group	Variables	Frequency	Period	Source
Monthly indicators - Total: 111				
Balance of payments	5	Monthly	2010M1-2025M12	CBU
Commodities	5	Monthly	2010M1-2025M12	IMF
Economic activity	11	Monthly	2017M1-2025M12	NSC; CBU
Exchange rate	5	Monthly	2010M1-2025M12	TE; CBU
Fiscal	2	Monthly	2018M1-2025M12	MoEF
Foreign inflation	4	Monthly	2010M1-2025M12	CEIC
Foreign rates	4	Monthly	2010M1-2025M12	CEIC
Google trends[a]	1	Monthly	2010M1-2025M12	Google Trends
Inflation	43	Monthly	2006M1-2025M12	NSC
Global food inflation	6	Monthly	2006M1-2025M12	FAO
Interest rates	5	Monthly	2013M1-2025M12	CBU
Labour market[b]	5	Monthly	2019M1-2025M12	CBU-OLX; CBU
Monetary	15	Monthly	2012M1-2025M12	CBU
Quarterly indicators[c] - Total: 67				
Balance of payments	8	Quarterly	2005Q1-2025Q4	CBU
Commodities	5	Quarterly	2010Q1-2025Q4	IMF
Economic activity	6	Quarterly	2010Q1-2025Q4	CEIC
Exchange rate	5	Quarterly	2010Q1-2025Q4	TE; CBU
Fiscal	2	Quarterly	2012Q1-2025Q4	MoEF
Foreign sector	7	Quarterly	2010Q1-2025Q4	CEIC
Inflation[d]	13	Quarterly	2006Q1-2025Q4	NSC; FAO
Monetary & interest rates	18	Quarterly	2010Q1-2025Q4	CBU
National accounts	3	Quarterly	2010Q1-2025Q4	NSC

Notes: CBU = Central Bank of Uzbekistan; NSC = National Statistics Committee of Uzbekistan; MoEF = Ministry of Economy and Finance; TE = Trading Economics; IMF = IMF database; CEIC = CEIC Data; FAO = FAO database.

[a] The is the job search index and it combines weekly search interest for Uzbek, query *ish -mustaqil -reja -vaqti* ("job," excluding "independent," "plan," and "time") and Russian, query работа -контрольная -проектная -курсовая -самостоятельная -москве -россии ("work," excluding terms associated with schoolwork and coursework, as well as results referring to Moscow or Russia), weighted by each language's average share of combined search interest over the calendar year: $I_t = w_{uz}^{y(t)} \cdot U_t + w_{ru}^{y(t)} \cdot R_t$, where U_t and R_t are the weekly Uzbek and Russian indices, $y(t)$ denotes the calendar year containing week t, and $w_{uz}^{y} = U_t/(U_t + R_t)$ averaged over weeks in year y (and $w_{ru}^{y} = 1 - w_{uz}^{y}$). The final monthly series is the simple average of weekly I_t within each month.

[b] This group also includes survey-based indicators of economic activity and business conditions, including firms' expectations regarding hiring and layoffs as well as producing. Additionaly, it also includes data on job vacancies posted on OLX.uz, the largest online classifieds platform in Uzbekistan. The job vacancy data are collected through web scraping.

[c] Most quarterly variables are constructed by aggregating monthly data using either a three-month average or a three-month sum, depending on the nature of the variable. These quarterly series are primarily used for GDP forecasting.

[d] This group mainly includes inflation-related indicators. Since no official quarterly GDP deflator is available for Uzbekistan, we construct proxy measures using the Producer Price Index (PPI) and services inflation.

4 Methodology

To forecast inflation and GDP growth, this study employs three traditional time series models (ARIMA, VAR, and BVAR) alongside nine machine learning methods (Ridge, Lasso, Elastic Net, PLS, PCR, SVR, Random Forest, XGBoost, and Gradient Boosting). A broad description of each model is provided below.

4.1 Benchmark models: ARIMA, VAR and BVAR

1. Univariate ARIMA model. The ARIMA model[1] is one of the most widely used methods for time series forecasting. It was originally proposed in 1970 by Box and Jenkins (George E.P. Box et al., 2016) and commonly referred to as the Box-Jenkins method.

Since ARIMA relies solely on the historical values of the target series, it provides a simple univariate benchmark against which more complex forecasting models can be evaluated. The ARIMA model is characterized by three parameters: p, d, and q, which are selected to best fit the time series data[2]. The general form of the ARIMA model is given by

$$\Delta^d y_t = c + \phi_1 \Delta^d y_{t-1} + \cdots + \phi_p \Delta^d y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t, \qquad \varepsilon_t \sim N(0, \sigma^2) \quad (1)$$

where $\Delta^d y_t$ denotes the series y_t after applying d differences, p and q are the orders of the autoregressive and moving-average terms, $\phi_1, \ldots, \phi_p$ and $\theta_1, \ldots, \theta_q$ are the corresponding autoregressive and moving-average coefficients, and ε_t is a white noise error term with mean zero and variance σ^2.

2. Vector Autoregression (VAR) model. Unlike ARIMA, VAR is a multivariate model that captures dynamic interactions among a small set of macroeconomically relevant variables. The VAR model is given by

$$Y_t = c + B_1 Y_{t-1} + B_2 Y_{t-2} + \cdots + B_p Y_{t-p} + u_t, \qquad u_t \sim N(0, \Sigma), \quad (2)$$

where Y_t is an n-dimensional vector of variables, $B_1, \ldots, B_p$ are the coefficient matrices, and u_t is an n-dimensional vector of innovations with covariance matrix Σ. As the number of variables increases, the number of unrestricted parameters also rises, which leads to an over-parameterization problem when estimating the model using traditional VAR methods. In such cases, due to overfitting, the model results may become unreliable. As a solution to this problem, the Bayesian approach (BVAR) can be applied.

3. Bayesian VAR (BVAR) model. The BVAR serves as a high-dimensional linear benchmark, allowing us to assess whether ML outperforms not only conventional linear models but also a high-dimensional shrinkage-based linear benchmark built on the same predictor set.

[1] Further estimation details for all three benchmark models (ARIMA, VAR and BVAR) are provided in Appendix B

[2] In the estimation, a seasonal component was also searched.

The BVAR addresses the curse of dimensionality by automatically selecting the degree of shrinkage. When the number of unknown coefficients is large relative to the available data, tighter prior distributions are applied, otherwise, looser priors are used. The prior distributions belong to the normal-inverse-Wishart family. Defining $\beta = [c, B_1, \ldots, B_p]$, the prior can be written as

$$\beta \mid \Sigma \sim N(b, \Sigma \otimes \Omega), \tag{3}$$

$$\Sigma \sim IW(\psi, \nu), \tag{4}$$

where b is the prior mean vector, Ω is the prior scale matrix, ψ is the scale matrix of the inverse-Wishart distribution, ν is the degrees of freedom, and ψ_{jj} denotes the j-th diagonal element of ψ.

In this study, the Minnesota prior is employed. This approach was introduced by Litterman (1979, 1985) and it shrinks each variable toward a persistent autoregressive process by centering the coefficient on the first own lag close to one, while centering the coefficients on higher-order lags and cross-variable lags at zero. Such an assumption provides a simple yet economically reasonable approximation of the behavior of macroeconomic variables.

4.2 Machine learning methods

This study employs nine machine learning methods that can be grouped into three families. The first group consists of penalized linear regression methods Ridge, Lasso, and Elastic Net which extend ordinary least squares by adding a regularization term to handle high-dimensional predictors. The second group includes dimension reduction methods Principal Component Regression (PCR) and Partial Least Squares (PLS) which extract a small number of components from the full set of predictors. The third group comprises Support Vector Regression (SVR) and the tree-based ensemble methods Random Forest, Gradient Boosting, and XGBoost. Although SVR can capture nonlinear relationships through kernel functions, it is implemented with a linear kernel in this study and therefore operates as a regularized linear method. The tree-based methods capture complex, nonlinear relationships and interaction effects in the data. Each method is described below.

1. Ridge, Lasso, and Elastic Net.[3] Ridge regression is a linear method (L2 regularization) that penalizes the sum of squared coefficients, shrinking them toward zero without eliminating any predictor from the model. As a result, every predictor remains in the model but with reduced influence. This makes Ridge particularly effective when many predictors are all somewhat relevant. Lasso (L1 regularization) also penalizes the regression coefficients, but forces some of them to exactly zero, effectively removing less relevant predictors and retaining only the most important ones. This makes Lasso particularly useful for variable selection, helping to prevent overfitting and improving model interpretability. Elastic Net combines both penalties (L1 and L2), performing well when

[3]For further details, see Tibshirani (1996) and Zou and Hastie (2005).

predictors are highly correlated or when the number of predictors exceeds the number of observations. All three methods share the same linear structure but differ in how they penalize the regression coefficients. The general objective function is:

$$\hat{\beta} = \arg\min \left[\sum_{t=1}^{T} \left(y_t - \beta_0 - \sum_{i=1}^{d} x_{it}\beta_i \right)^2 + \lambda \sum_{i=1}^{d} \left((1-\alpha)\beta_i^2 + \alpha|\beta_i| \right) \right] \tag{5}$$

where β_0 is the intercept, β_i are the regression coefficients, x_{it} is the value of predictor i at time t, d is the number of predictors, T is the number of observations, $\lambda \geq 0$ controls the overall regularization strength, and $\alpha \in [0, 1]$ determines the mix between the two penalties. When $\alpha = 0$, the penalty reduces to Ridge, when $\alpha = 1$, it reduces to Lasso.[4] Intermediate values of α balance these two properties, making Elastic Net particularly suitable when predictors are highly correlated.

2. Principal Component Regression (PCR) and Partial Least Squares (PLS).[5] Both PCR and PLS are dimension reduction methods that address the challenge of forecasting with a large number of potentially correlated predictors. Rather than using the original predictors directly, both methods first extract a small number of components $z_1, \ldots, z_K$ from the predictor matrix X, and then regress the target variable on these components:

$$\hat{y}_{t+h} = \gamma_0 + \sum_{k=1}^{K} \gamma_k z_{kt} \tag{6}$$

where $z_{kt} = \mathbf{w}_k' \mathbf{x}_t$ are linear combinations of the original predictors, $\mathbf{w}_k$ are the component weight vectors, γ_k are the regression coefficients on the components, and $K \ll d$ is the number of retained components selected by cross-validation.

The two methods differ in how the weight vectors $\mathbf{w}_k$ are determined. PCR applies Principal Component Analysis (PCA) to the predictor matrix X via its singular value decomposition:

$$X = USV^\top \tag{7}$$

where U and V are orthogonal matrices of left and right singular vectors, and S is a diagonal matrix of singular values. The columns of US, which are the principal components (with weight vectors $\mathbf{w}_k$ given by the columns of V), are then used as regressors. Since these components are chosen to explain maximum variance in X alone, they may not be the most informative for predicting y.[6]

PLS addresses this limitation by extracting components that jointly maximize the covariance between X and y. Specifically, the weight vector $\mathbf{w}$ for the first PLS component is obtained by solving:

$$\mathbf{w} = \arg\max_{\|\mathbf{w}\|=1} \operatorname{Cov}(X\mathbf{w}, y)^2 \tag{8}$$

[4] In `scikit-learn`, this mixing parameter α corresponds to `l1_ratio` (reported as the "ℓ_1 ratio" in Table 6), while the overall strength λ corresponds to `alpha`.

[5] For further details, see Ahn and Bae (2022), Groen and Kapetanios (2009), and Abdi (2010).

[6] In our implementation, the regression on the retained components is estimated with a Ridge penalty to further stabilize the coefficient estimates.

and the corresponding component is $\mathbf{t} = X\mathbf{w}$. This ensures that the extracted components are directly informative for predicting y. Subsequent components are extracted iteratively after removing the effect of the previous component from both X and y.

3. Support Vector Regression (SVR). Support Vector Regression, introduced by Cortes and Vapnik (1995); Vapnik (2013), adapts the support vector machine framework to regression problems. It seeks a function that deviates from the observed target by at most ϵ while remaining as flat as possible. The prediction function takes the form:

$$y_t = \mathbf{w}^\top \phi(\mathbf{x}_t) + b + e_t \tag{9}$$

where $\mathbf{w}$ is the weight vector, b is the bias term, $\phi(\mathbf{x}_t)$ maps the predictors into a higher-dimensional feature space, and e_t is the disturbance term. Flatness is achieved by minimizing the norm of the weight vector $\|\mathbf{w}\|^2$, which reduces model complexity and the risk of overfitting.

Since requiring all deviations to fall within ϵ is often infeasible, the method introduces slack variables ξ_t and ξ_t^* that measure the extent to which predictions fall outside the ϵ-margin. The optimization problem is formulated as:

$$\min_{\mathbf{w},b,\xi,\xi^*} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{t=1}^{T}(\xi_t + \xi_t^*) \tag{10}$$

subject to

$$\begin{cases} y_t - \mathbf{w}^\top \phi(\mathbf{x}_t) - b \le \epsilon + \xi_t, \\ \mathbf{w}^\top \phi(\mathbf{x}_t) + b - y_t \le \epsilon + \xi_t^*, \\ \xi_t, \xi_t^* \ge 0, \end{cases} \tag{11}$$

where $C > 0$ is a regularization parameter controlling the trade-off between model flatness and tolerance for deviations larger than ϵ. A larger C enforces stricter adherence to the ϵ-tube, which can increase the risk of overfitting, while a smaller C permits more deviations, improving generalization at the cost of possible underfitting. The mapping $\phi(\mathbf{x}_t)$ is computed implicitly through a kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$, which in general enables SVR to capture nonlinear relationships without explicitly constructing the high-dimensional feature space. In this study, a linear kernel, $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^\top \mathbf{x}_j$, is employed, so that SVR operates as a regularized linear method. It nevertheless differs from the penalized regression methods described above through its ϵ-insensitive loss function, which ignores small forecast errors and makes the estimates more robust to noise in the target variable.

4. Tree-Based Ensemble Methods. The next three models are nonlinear ensemble methods built on decision trees. A single decision tree partitions the predictor space into regions and assigns a constant prediction to each region. While intuitive, individual trees are unstable and prone to overfitting. Ensemble methods overcome this by combining many trees into a single, more robust predictor. They fall into two families that differ in how the trees are constructed and combined: *bagging*, where trees are grown independently

and averaged (Random Forest), and *boosting*, where trees are grown sequentially, each correcting the errors of its predecessors (Gradient Boosting and XGBoost).

4a. Random Forest (RF). Random Forest, proposed by Breiman (2001), is a bagging method that constructs an ensemble of B decision trees and averages their predictions. Each tree is trained on a bootstrap sample of the data, and at each split only a random subset of predictors is considered. This randomization decorrelates the trees, reducing the variance of the combined forecast. The prediction is given by:

$$\hat{y}_{t+h} = \frac{1}{B} \sum_{b=1}^{B} \hat{g}_b(\mathbf{x}_t) \tag{12}$$

where B is the number of trees and $\hat{g}_b(\mathbf{x}_t)$ is the prediction of the b-th tree. Because the trees are grown independently and then averaged, Random Forest is effective at reducing overfitting while requiring relatively little tuning.

4b. Gradient Boosting (GB). Gradient Boosting, introduced by Friedman (2001), is a boosting method that builds trees sequentially. Starting from an initial prediction, each new tree is fitted to the residual errors of the current ensemble, and is then added to the model with a small weight. The model is updated iteratively:

$$F_m(\mathbf{x}_t) = F_{m-1}(\mathbf{x}_t) + \eta\, h_m(\mathbf{x}_t) \tag{13}$$

where $F_m(\mathbf{x}_t)$ is the prediction after m trees, $h_m(\mathbf{x}_t)$ is the tree fitted to the residuals at step m, and $\eta \in (0, 1]$ is the learning rate that controls the contribution of each tree. By focusing successively on the errors that remain, boosting reduces bias and can capture complex nonlinear patterns, though it is more sensitive to overfitting and requires careful tuning of the learning rate and the number of trees.

4c. Extreme Gradient Boosting (XGBoost). XGBoost, developed by Chen and Guestrin (2016), follows the same sequential boosting principle as Gradient Boosting but augments the objective function with an explicit regularization term that penalizes model complexity. At each step, the trees are chosen to minimize:

$$\mathcal{L} = \sum_{t=1}^{T} \ell(y_t, \hat{y}_{t+h}) + \sum_{m} \Omega(h_m), \qquad \Omega(h) = \gamma\, L + \frac{1}{2}\lambda \|\boldsymbol{\omega}\|^2 \tag{14}$$

where $\ell(\cdot)$ is a differentiable loss function, $\Omega(h_m)$ is the regularization term for tree h_m, L is the number of leaves in the tree, $\boldsymbol{\omega}$ is the vector of leaf weights, and γ and λ are regularization parameters penalizing the number of leaves and the magnitude of the leaf weights, respectively. This explicit regularization, together with several computational optimizations, makes XGBoost more robust to overfitting and typically faster than standard Gradient Boosting.

5 Estimation strategy

There are many ways to improve the performance of ML methods, as they are highly sensitive to data preprocessing, feature selection, and hyperparameter tuning. To ensure reliable and reproducible forecasts, we therefore follow a carefully designed estimation procedure that covers all steps from data preparation to forecast evaluation. Figure 2 provides an overview of the full estimation pipeline.

5.1 Forecasting framework

As mentioned in the introduction, inflation and GDP growth are our target variables.

To forecast **inflation**, we use the consumer price index (CPI). Since the CPI is available at monthly frequency, monthly data are used for all indicators in the inflation models. The forecast horizons are 1, 3, 6, 9, and 12 months ahead ($h = 1, 3, 6, 9, 12$ months).

Economic development in Uzbekistan can be divided into two periods, before and after 2017 (marked by the start of FX liberalization), and the quality and reliability of the data differ significantly between these two periods. As shown in Figure 10, the behavior of the inflation differs substantially across these two periods, with high volatility and large jumps in inflation.[7] Although inflation was volatile after 2017, the reliability of the data in this period is higher than before 2017.[8] For these reasons, monthly data from 2018M1 to 2025M12 are used to forecast inflation.

To forecast **GDP growth**, we use GDP data in levels. Since GDP is measured at quarterly frequency, quarterly data are used in the estimation. The data cover the period from 2012Q1 to 2025Q4. Since ML models require a sufficient amount of data to produce reliable estimates, the dataset begins in 2012Q1, including the period before the FX liberalization.[9] The forecast horizons for GDP growth are 1, 2, 3, and 4 quarters ahead ($h = 1, 2, 3, 4$ quarters).

It should also be noted that GDP growth was volatile and experienced significant jumps over the estimation period (see Figure 10). To mitigate the issues arising from covering two markedly different periods, several methods are applied, which are discussed in the following sections. In addition, to use more reliable data in the GDP estimation, around 100 candidate variables are analyzed and the most informative ones are selected.

In both cases (inflation and GDP growth), a direct multi-horizon forecasting approach is used, in which the value at horizon h is forecast directly from information available at the forecast origin.

[7] This further illustrates why such indicators are difficult to forecast, as they are highly volatile and subject to significant structural changes and shocks.

[8] The calculation methodology also changed across periods. For example, the CPI was previously calculated using flexible weights, whereas after 2018 it has been calculated using fixed weights.

[9] We also tried using monthly indicators to forecast GDP growth, but this produced high errors. Specifically, we used monthly indicators from the services, industry, and construction sectors, which together account for more than 80 percent of GDP in Uzbekistan. However, since the sum of these sectoral indicators does not equal total GDP (due to value added and statistical adjustments), this approach resulted in errors of around 1 percentage point in the calculated GDP growth.

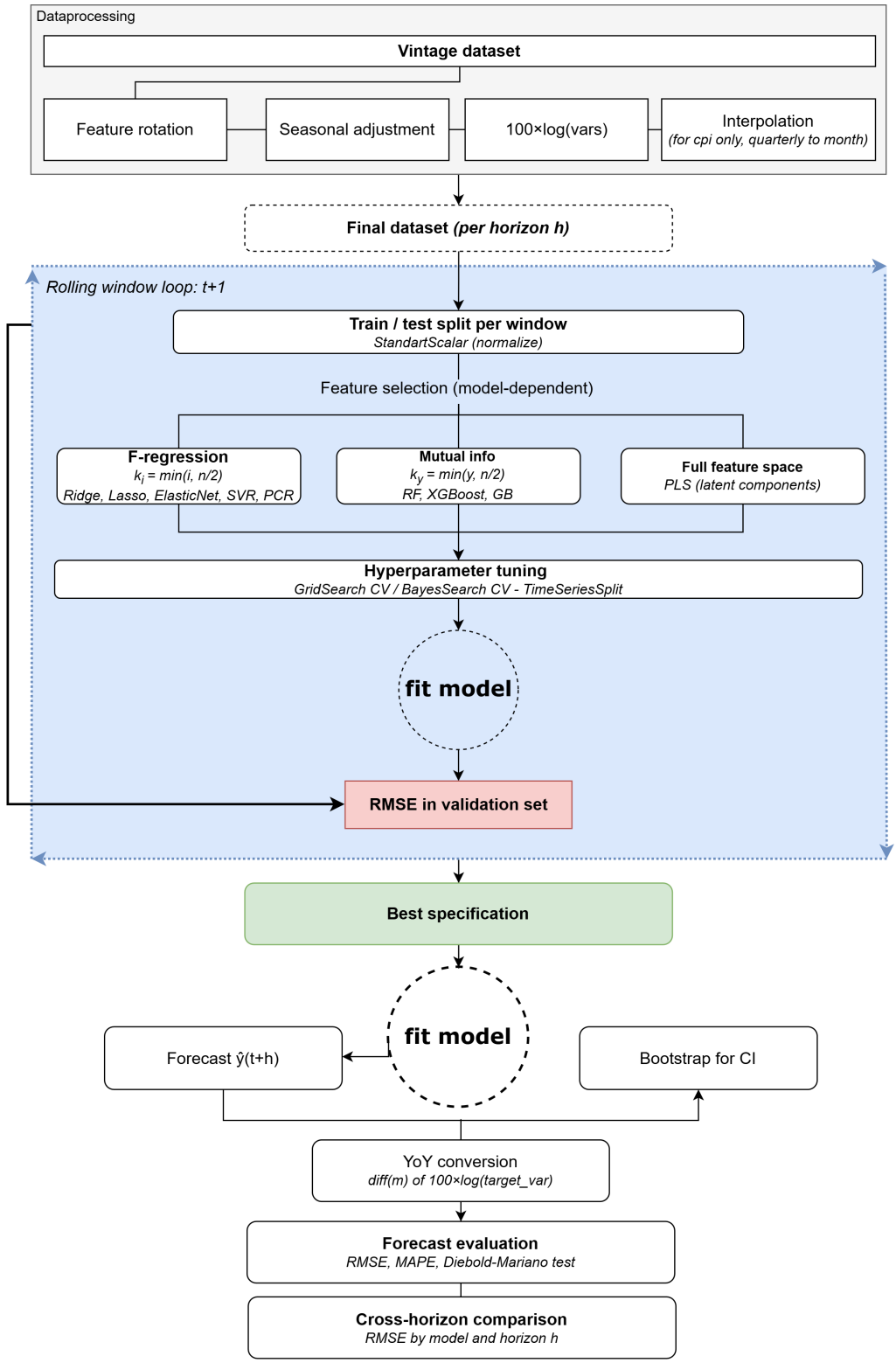


Figure 2: Estimation strategy

Source: Author's own illustration of the machine learning estimation pipeline implemented in this study.

5.2 Data processing and dimensionality reduction

1. Data processing. For both target variables, the same general preprocessing steps are applied. We first analyze the available data, aiming to cover as many relevant indicators as possible. Based on the available data, we also perform some feature engineering, mainly for the high-frequency series extracted from Google Trends and for variables related to the national accounts, such as the GDP deflator and several other variables in real terms.[10]

We first seasonally adjust all variables using the X-13ARIMA-SEATS method, implemented via the IRIS package in Python, with the exception of a few series whose characteristics make adjustment unnecessary.[11] For the CPI models, certain quarterly variables are interpolated to monthly frequency using linear interpolation, as this was found to improve forecast performance. Variables are then transformed depending on their characteristics. Some are converted to log form as $100 \times \log(\text{var})$, while others are expressed as monthly or annual percentage changes.

Once the final dataset is prepared, all variables are normalized to remove differences in scale across predictors. Several models are sensitive to the scale of predictors, in particular, regularization-based models such as Ridge, Lasso, and Elastic Net penalize coefficient magnitudes, so variables measured on different scales would be penalized unequally without normalization. We therefore apply Z-score normalization (standardization), which transforms each variable to have zero mean and unit variance.[12]

$$\tilde{x}_{it} = \frac{x_{it} - \bar{x}_i}{s_i} \tag{15}$$

where $\bar{x}_i$ is the sample mean and s_i is the standard deviation of variable i. The scaling parameters (mean and standard deviation) are estimated using the training data only, and then applied to the test observation. This ensures that the model never sees any information from the test period during training, preventing data leakage.

2. Dimensionality reduction. Although machine learning methods are generally robust to high-dimensional predictor sets, the combination of a large number of predictors ($d > 170$) and a relatively short training window ($T = 72$ months for CPI and $T = 44$ quarters for GDP) creates a challenging high-dimensional setting in which even flexible models risk overfitting. To address this, a pre-screening step is applied as a preliminary feature selection stage that reduces the number of candidate predictors before model

[10] We construct proxy indicators for the quarterly GDP deflator based on the producer price index (PPI) and services inflation, as Uzbekistan does not publish a standard quarterly GDP deflator (only a cumulative one is available).

[11] Seasonal patterns repeat at the same time every year and are highly predictable. If we include them in the data, it would cause the model to learn these patterns instead of the true economic signal, which is what matters for policy.

[12] Other common normalization methods include Min-Max scaling, which bounds variables to the $[0, 1]$ interval: $\tilde{x}_{it} = \frac{x_{it} - x_{i,\min}}{x_{i,\max} - x_{i,\min}}$, and Robust scaling, which uses the median and interquartile range (IQR) instead of the mean and standard deviation, making it less sensitive to outliers: $\tilde{x}_{it} = \frac{x_{it} - \text{median}(x_i)}{\text{IQR}(x_i)}$.

estimation. This step complements rather than replaces the built-in regularization mechanisms of the machine learning models, improving both computational efficiency and the stability of cross-validation results.

Several filter methods were considered for feature selection, including Lasso-based selection, F-regression, and mutual information. While Lasso is commonly used as a variable selector in the macroeconomic forecasting literature (Li and Chen, 2014; Chan-Lau, 2017; Bai and Ng, 2008), its linear selection criterion makes it unsuitable as a pre-screener for nonlinear tree-based models. This study therefore applies F-regression (Chapman and Desai, 2023) for linear models (Ridge, Lasso, Elastic Net, SVR, and PCR), which ranks predictors by their linear association with the target, and mutual information (Kraskov et al., 2004) for tree-based models (RF, XGBoost, and GB), which captures nonlinear dependencies.[13] This matching of the selection criterion to the model's underlying assumptions ensures that informative predictors are not discarded before estimation. PLS is the only exception, as it operates on the full predictor space and performs its own internal dimension reduction through the extraction of latent components.

To capture temporal dependencies, each predictor is augmented with its lagged values. For the CPI models, three lags are included for each variable, while for the GDP models, four lags are used, reflecting the quarterly frequency and the importance of annual patterns in GDP growth. Combined with the original variables, this lag augmentation results in a predictor matrix of over 200 features for both CPI and GDP models.

5.3 Sample split and cross-validation

1. Sample split. The dataset is divided into training and test sets. Approximately 85 percent of the data is used for training and the remaining 15 percent for testing. Specifically, for CPI the training set covers 72 months and the test set covers 12 months (1 year), while for GDP the training set covers 44 quarters and the test set covers 8 quarters (2 years).[14][15] The full sample spans 2018M1-2025M12 for CPI (96 monthly observations) and 2012Q1-2025Q4 for GDP (56 quarterly observations).

Since the GDP sample covers two structurally different periods (before 2017, which was relatively stable, and after 2017, which was characterized by significant shocks and structural changes), we apply exponential decay weights during estimation to reduce the influence of the earlier, less representative observations. Each observation t in a training window of length T receives a weight:

$$w_t = \frac{\rho^{T-t}}{\sum_{s=1}^{T} \rho^{T-s}} \cdot T, \quad t = 1, \ldots, T \tag{16}$$

[13] Both methods are implemented using the `SelectKBest` function from the `scikit-learn` library in Python, with `f_regression` for linear models and `mutual_info_regression` for tree-based models.

[14] A longer test period is generally recommended in practice. However, given the limited length of the available time series for Uzbekistan, a shorter test period is unavoidable.

[15] Since the training window is fixed at 72 months for CPI and the full sample covers 96 observations (93 after lag adjustment), the first rolling window leaves 21 months available for testing. As the window moves forward, the available test period shortens. To ensure consistency across windows, the test period is fixed at 12 months. The same applies to GDP.

where $\rho \in (0, 1)$ is the decay parameter, set to $\rho = 0.92$.[16] Weights are normalized to sum to T to preserve the effective sample size in the penalized objective function, so that older observations are progressively downweighted relative to more recent ones. This approach is motivated by two considerations. First, Pesaran and Timmermann (2007) show that pre-break observations need not be discarded but can be discounted to improve forecast accuracy, and is consistent with the finding that exponentially weighted moving averages are robust to structural change in macroeconomic forecasting (Giraitis and Kapetanios, 2012). Second, sample weighting has been shown to improve the forecasting performance of machine learning models (Wang et al., 2023; Gu et al., 2020).

2. Cross-validation. There are two common cross-validation approaches for hyperparameter tuning: standard k-fold cross-validation and walk-forward cross-validation.[17] Standard k-fold cross-validation is not appropriate for time series data, as it randomly shuffles observations, allowing future information to leak into the training set. Instead, this study uses walk-forward cross-validation, which ensures that validation observations always follow training observations in time (see Figure 3 for an illustration). Goulet Coulombe et al. (2022) show that this form of cross-validation is best practice for hyperparameter selection in macroeconomic forecasting.

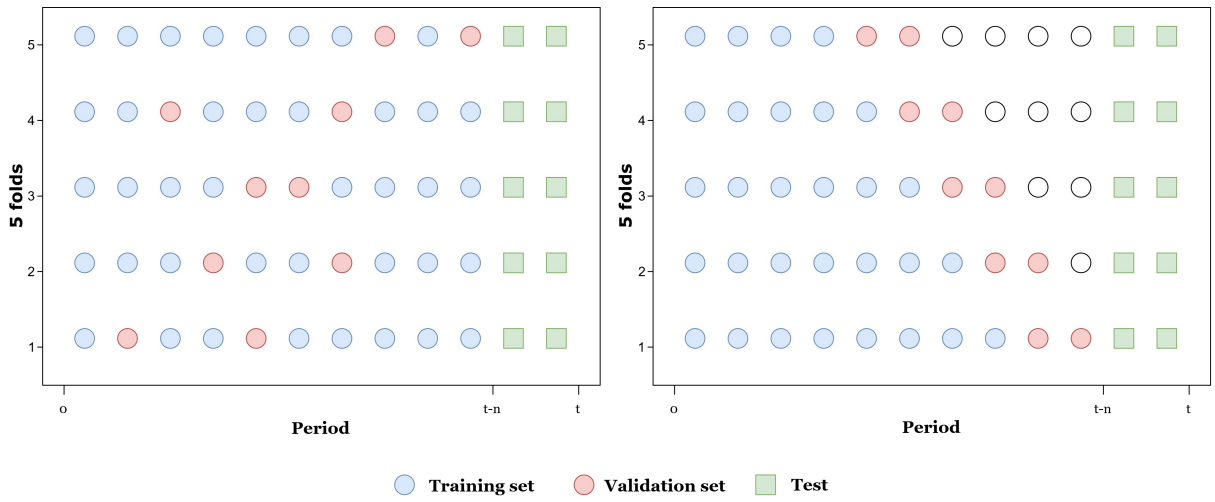


Figure 3: Standard k-Fold (left) vs. Walk-Forward Cross-Validation (right)

Note: Standard k-fold cross-validation (left) randomly splits the data, so the model may train on future observations to predict the past. Walk-forward cross-validation (right) always trains on past data and validates on future data, which is the correct approach for time series forecasting.

Regarding evaluation schemes,[18] this study adopts a rolling window approach, as it

[16] Several values of ρ were tested, and $\rho = 0.92$ yielded the best overall results across models and horizons. This value is therefore applied uniformly across all GDP models.

[17] Walk-forward cross-validation is also known as time-series cross-validation, implemented via `TimeSeriesSplit` in `scikit-learn`.

[18] In the forecasting literature, three evaluation schemes are common: a *rolling window*, where a fixed-length training window moves forward one period at a time; an *expanding window*, where all historical

allows the model to focus on the most recent observations by dropping older ones as the window moves forward (see Figure 4 for an illustration). This is particularly important in the case of Uzbekistan, where the available time series are short and the economy has undergone significant structural changes, making more recent data more reflective of the current relationships among variables (Giraitis and Kapetanios, 2012).

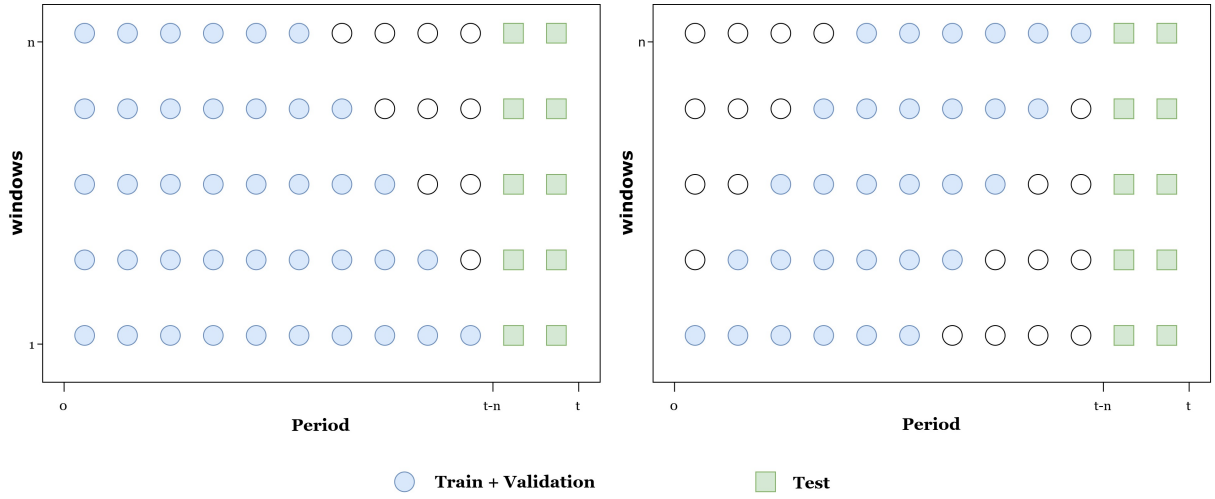


Figure 4: Expanding Window (left) vs. Rolling Window (right)

Note: In the expanding window approach (left), all historical observations are retained as the sample grows. This study adopts the rolling window approach. In the rolling window approach (right), the training window has a fixed length and moves forward one period at a time, dropping the oldest observation as a new one is added. Blue circles denote training and validation observations; green squares denote test observations; white circles denote observations excluded from the current window.

For hyperparameter tuning, `TimeSeriesSplit` is configured with $n\_splits = 5$ and a test fold size of 12 months for CPI, and $n\_splits = 4$ and a test fold size of 2 quarters for GDP.[19]

5.4 Hyperparameter optimization

Hyperparameter tuning is a critical step in ML model estimation, as the predictive performance of each model depends heavily on the choice of its hyperparameters. Unlike model parameters, which are estimated from the data, hyperparameters control the complexity and regularization of the model and must be selected prior to estimation. Poor hyperparameter choices can lead to two problems: *overfitting*, where the model learns the training

observations are retained as the sample grows; and a *fixed window*, where a single train-test split is used throughout.

[19] A test fold size of two quarters is chosen to balance validation reliability against training sample size for GDP. With a 44-quarter rolling window and four splits, a larger test fold of four quarters would reduce the minimum training size to only 28 quarters ($44 - 4 \times 4 = 28$), which is insufficient for stable hyperparameter tuning. Retaining two quarters as the test fold preserves at least 36 quarters for training in the first fold ($44 - 4 \times 2 = 36$).

data too well including its noise, and performs poorly on new data, or *underfitting*, where the model is too simple to capture the true patterns in the data. Proper hyperparameter tuning finds the right balance between these two extremes. In this study, two hyperparameter optimization techniques are evaluated: `GridSearchCV` and `BayesSearchCV`.[20]

GridSearchCV exhaustively searches over a predefined grid of hyperparameter values, evaluating every possible combination. While this guarantees that the best combination within the grid is found, it becomes computationally expensive as the number of hyperparameters and candidate values grows.

BayesSearchCV is a more efficient alternative that avoids evaluating every combination. Instead, it learns from each evaluation. If certain hyperparameter values produce good results, the search focuses on that region of the parameter space. This allows it to find a good solution in far fewer evaluations than an exhaustive grid search. In both cases, `TimeSeriesSplit` cross-validation is used as the inner evaluation criterion, with RMSE as the selection metric.

In this study, two hyperparameter tuning strategies are employed:

1. **Tune-once**: hyperparameters are selected on the first window and held fixed for all subsequent windows.

2. **Re-tune**: hyperparameters are re-selected at each rolling window, alongside feature selection.

Although re-tuning at each window did not yield substantial improvements in forecast accuracy, it is retained as the baseline approach since it automatically adapts to new data and is considered best practice in the rolling window literature. The tune-once strategy, by contrast, is considerably faster but requires careful selection of the initial hyperparameters, as these are held fixed throughout the entire evaluation period.

The hyperparameter search spaces for all models are summarized in Appendix C.

5.5 Model performance evaluation

In line with standard practice in the forecasting literature, model performance is evaluated using the Root Mean Square Error (RMSE), a widely used measure of predictive accuracy. RMSE quantifies the average magnitude of forecast errors and assigns greater weight to larger errors, making it a reliable criterion for comparing models. For a given model m, it is computed as the square root of the average squared difference between the actual

[20] Bayesian search did not yield meaningful improvements in forecast accuracy relative to grid search, and required substantially longer computation time. `GridSearchCV` is therefore adopted as the primary tuning technique.

values y_t and the forecasts $\hat{y}_t^m$:[21]

$$\text{RMSE}^m = \sqrt{\frac{1}{Z}\sum_{t=1}^{Z}(y_t - \hat{y}_t^m)^2} \tag{17}$$

where y_t is the target variable at time t, $\hat{y}_t^m$ is the forecast generated by model m, and Z is the total number of out-of-sample forecasts. A lower RMSE indicates more accurate predictions.

In addition to evaluating individual models, we assess the performance of a simple equal-weighted combination of forecasts. For each forecast horizon h, the combined forecast is constructed as the arithmetic mean of the individual model forecasts:

$$\hat{y}_t^{\text{ensemble}} = \frac{1}{N}\sum_{m=1}^{N}\hat{y}_t^m \tag{18}$$

where N is the number of models in the combination and $\hat{y}_t^m$ is the forecast from model m.

We construct two separate combinations: one averaging across the nine ML models (Ridge, Lasso, Elastic Net, PLS, PCR, SVR, Random Forest, XGBoost, and Gradient Boosting ($N = 9$)), and one averaging across the three benchmark models (ARIMA, VAR, and BVAR ($N = 3$)). The forecast accuracy of each combination is evaluated using RMSE over the same out-of-sample evaluation period as the individual models:

$$\text{RMSE}^{\text{ensemble}} = \sqrt{\frac{1}{Z}\sum_{t=1}^{Z}(y_t - \hat{y}_t^{\text{ensemble}})^2} \tag{19}$$

For the machine learning models, forecast uncertainty (confidence intervals) is estimated using a moving-block bootstrap, which resamples consecutive observations to preserve the time-series structure of the data.

6 Empirical results

To evaluate and compare forecast performance across models, we report RMSE over a one-year out-of-sample period for CPI and a two-year out-of-sample period for GDP. The out-of-sample evaluation period covers 2025M1-2025M12 for CPI and 2024Q1-2025Q4 for GDP. Results for each variable are presented and discussed below.

In addition to point forecasts, we report confidence intervals for the machine learning models using a moving-block bootstrap, which resamples consecutive observations while

[21] In addition to RMSE, the Diebold–Mariano test (Diebold and Mariano, 2002) was applied pairwise against the benchmark models to assess whether differences in predictive accuracy are statistically significant. However, given the small out-of-sample evaluation windows (particularly at longer horizons), the test has low statistical power and did not yield significant differences between models. The DM results are therefore not reported, and model comparison relies on RMSE.

preserving the time-series dependence of the data (see Appendix D for the methodology and Appendix G for the forecast paths with confidence intervals for both CPI and GDP). For each bootstrap sample, the models are re-estimated using the hyperparameters selected from the original training data (without re-tuning). Consequently, the reported intervals capture only coefficient estimation uncertainty conditional on the chosen hyperparameters. They do not account for uncertainty arising from hyperparameter selection or irreducible forecast error and should therefore be interpreted as a lower bound on total forecast uncertainty rather than as fully calibrated prediction intervals.

6.1 CPI forecasting

For CPI, models are estimated separately for five forecast horizons ($h = 1, 3, 6, 9, 12$ months), resulting in a total of 45 ML models and 3 benchmark models.[22] The estimated models may differ from one another in terms of the predictors selected, the training window used, and the estimation period covered.

Figure 5 reports the RMSE results for all models over different horizons. The results show that ML models (crimson) consistently outperform the benchmark models (blue) at longer horizons. While the benchmark models (ARIMA, VAR, and BVAR) are competitive at short horizons, their performance deteriorates rapidly as the horizon extends. ML models, by contrast, remain relatively stable or even improve at longer horizons, demonstrating their ability to maintain forecast accuracy where traditional models break down.

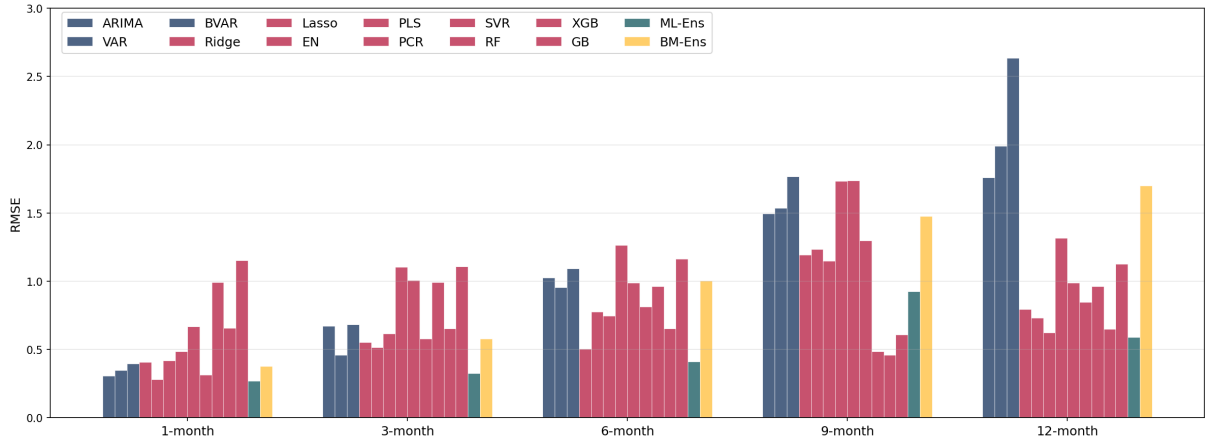


Figure 5: CPI forecast RMSE by model and horizon.

Note: ML-Ens and BM-Ens denote the ML Ensemble and Benchmark Ensemble, respectively, computed as the simple average of the nine ML models and three benchmark models.

It is also worth noting that machine learning methods improve upon traditional benchmarks at every forecast horizon when combined into an ensemble. Simply averaging the forecasts from the nine ML models improves forecast performance.

[22] For the benchmark models, the same model specification is used across all horizons, only the estimation window is adjusted accordingly.

This can be clearly seen in Table 2. The ML Ensemble achieves the lowest RMSE among all models at four of the five horizons ($h = 1, 3, 6, 12$), outperforming both the best individual ML model and the best individual benchmark, and outperforming the Benchmark Ensemble by a wide margin throughout. This confirms that forecast combination, rather than any single model specification, is the most robust strategy for CPI inflation forecasting at the CBU.

Table 2: RMSE - CPI Inflation Forecasting

Model	$h = 1$	$h = 3$	$h = 6$	$h = 9$	$h = 12$
Benchmark models					
ARIMA	0.3076	0.6709	1.0273	1.4951	1.7614
VAR	0.3486	0.4603	0.9540	1.5366	1.9919
BVAR	0.3964	0.6842	1.0942	1.7670	2.6356
Machine learning models					
Ridge	0.4080	0.5530	0.5059	1.1942	0.7955
Lasso	0.2826	0.5143	0.7753	1.2343	0.7322
Elastic Net	0.4171	0.6148	0.7474	1.1507	0.6224
PLS	0.4845	1.1060	1.2658	1.7345	1.3170
PCR	0.6703	1.0064	0.9874	1.7386	0.9888
SVR	0.3155	0.5803	0.8140	1.2991	0.8457
RF	0.9941	0.9919	0.9635	0.4845	0.9626
XGBoost	0.6557	0.6530	0.6542	0.4585	0.6496
GB	1.1521	1.1097	1.1650	0.6103	1.1271
Combination forecasts					
ML Ensemble	0.2691	0.3241	0.4132	0.9243	0.5888
BM Ensemble	0.3795	0.5790	1.0047	1.4784	1.6987

Notes: RMSE values for year-on-year CPI inflation forecasts across horizons $h = 1, 3, 6, 9, 12$ months. Lower RMSE is better.

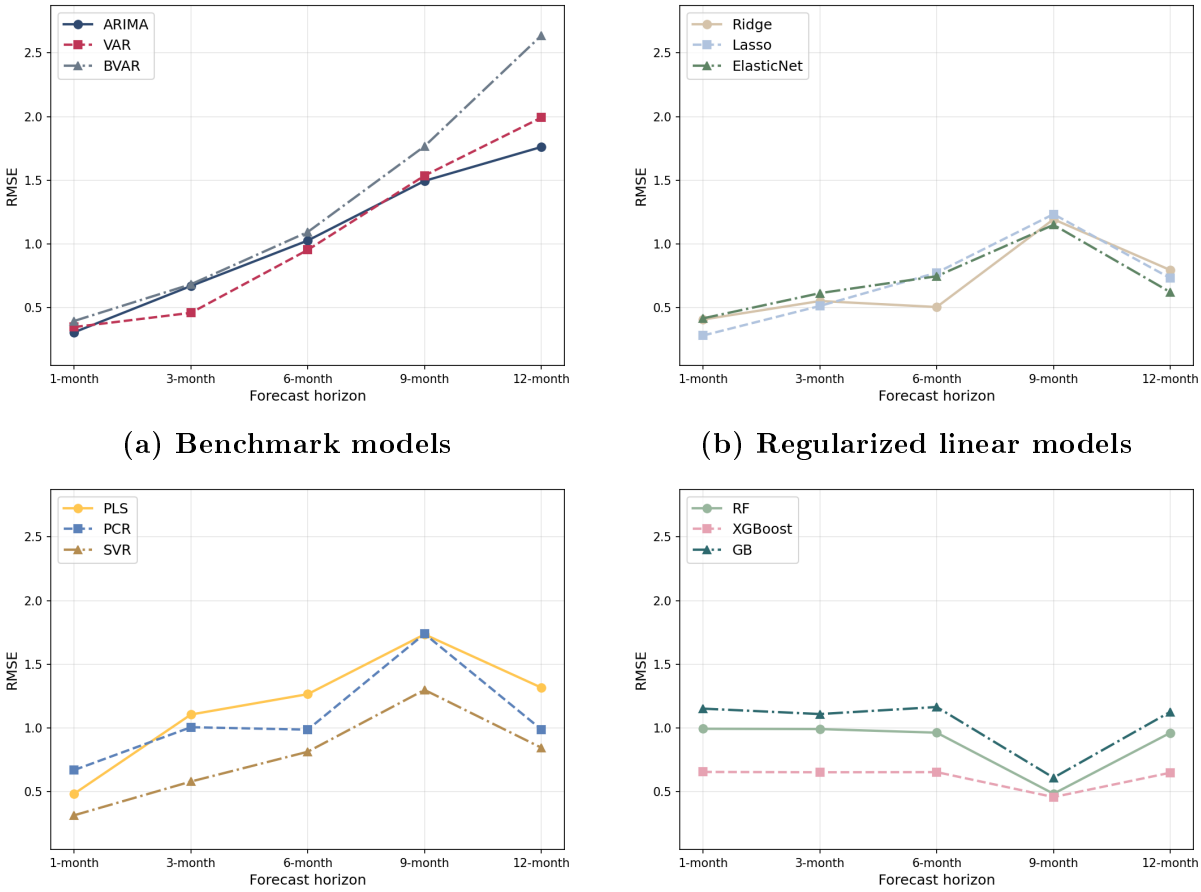
Among individual models, the relative performance of regularized linear models shifts systematically with the forecast horizon. Lasso achieves the lowest RMSE among ML models at $h = 1$ (0.2826), consistent with the intuition that only a small number of predictors carry signal for near-term inflation, so sparse selection avoids overfitting. At longer horizons, Ridge ($h = 6$: 0.5059) and Elastic Net ($h = 12$: 0.6224) outperform Lasso, suggesting that as the horizon increases, predictive signal becomes more diffuse across a larger set of correlated variables, favoring models that shrink rather than eliminate coefficients.

PLS and PCR consistently underperform across all horizons and are the weakest ML models in the panel, confirming that variance-maximizing dimension reduction is poorly aligned with CPI's predictive structure.

In the case of PLS, this is consistent with Ahn and Bae (2022), who show that the optimal number of PLS factors for forecasting can be considerably smaller than the number

of factors needed to explain the predictor set itself, and that adding factors beyond this optimal point can reduce out-of-sample forecasting power even as in-sample fit continues to improve. In other words, PLS's tendency to extract factors that maximize explained variance in the predictors does not guarantee that those factors are useful for prediction. This may explain why PLS and PCR fail to match the performance of models that select or shrink predictors based on their relevance to the target variable directly.

Tree-based models (RF, XGBoost, GB) are broadly uncompetitive across horizons, with RMSE values that remain nearly unchanged from $h = 1$ to $h = 12$. The exception is $h = 9$, where XGBoost (0.4585) and RF (0.4845) record the lowest RMSE in the panel. This result should be interpreted with caution. The $h = 9$ column is anomalous for the linear models as well, whose RMSE rises sharply at this horizon before falling back at $h = 12$, suggesting that the $h = 9$ evaluation window is unusually unfavorable for momentum-driven forecasts rather than that tree-based models possess a true advantage at this particular horizon. Given the short evaluation window of twelve months, a single horizon-specific episode can dominate the RMSE ranking.



(a) Benchmark models

(b) Regularized linear models

(c) Dimension reduction models

(d) Non-linear ML models

Figure 6: CPI forecast RMSE by model group across horizons.

Generally speaking, at shorter horizons, all models successfully capture the overall

inflation trend and move in the same direction, with linear ML models performing particularly well (see Appendix E). ML models and benchmarks perform comparably at these horizons, suggesting that short-term forecasting relies heavily on recent momentum rather than complex nonlinear relationships.

At longer horizons, the picture changes, that's ML models track the actual inflation path more closely, while benchmark models (VAR, ARIMA, BVAR) overshoot and fail to capture the inflation trend. As recent momentum becomes less informative, forecasting accuracy increasingly depends on capturing complex relationships between inflation and its many drivers. ML methods handle this better by capturing nonlinear interactions across a large set of macroeconomic variables, as can also be clearly seen in Appendix E, where ML models track the inflation trend more closely at longer horizons.

Both findings are consistent with the broader literature, which shows that allowing for more complex relationships improves forecast performance, linear ML models tend to perform better at short horizons, while nonlinear models better capture the underlying dynamics at longer horizons (Slimani and Chourabi, 2025; Medeiros and Vasconcelos, 2019; Maehashi and Shintani, 2020).

Figure 6 also illustrates this pattern clearly. The benchmark models show a significant increase in RMSE as the horizon extends, while ML models deteriorate much more slowly. Most ML models recover and improve at $h = 12$, reflecting their ability to capture complex, nonlinear relationships that become increasingly important for longer-term inflation forecasting. It is also worth noting that forecasting inflation in Uzbekistan over longer horizons (up to one year ahead) appears to require more information rather than fewer variables. That is, inflation dynamics at longer horizons are better explained by a larger set of predictors than by a small, parsimonious set. In this setting, ML models hold a clear advantage, as they are better equipped to handle high-dimensional predictor sets than traditional benchmark models.

6.2 GDP forecasting

For GDP, models are estimated for four horizons ($h = 1, 2, 3, 4$ quarters), resulting in a total of 36 ML models and 3 benchmark models. As mentioned in the previous section, the estimated models may also differ in terms of predictors, training window, and estimation period.

Most models capture the overall GDP growth trend well. However, tree-based models struggled to forecast 2024-2025 GDP growth, as they cannot predict values outside the range observed during training (see Appendix F). GDP growth between 2017 and 2023 was highly volatile and, on average, significantly lower than in earlier periods, reflecting the sequential shocks of FX liberalization, COVID-19 and the Russia-Ukraine war (see Figure 10). When GDP growth rose sharply in 2024-2025, driven mainly by structural economic reforms, tree-based models had no historical reference point for this level of growth, given the weak growth rates in the preceding periods and systematically undershot the actual values. Linear models do not share this limitation and therefore tracked the acceleration more reliably.

For GDP forecasting, benchmark models remain competitive across all horizons (ex-

cept BVAR at $h = 4$ horizon). Among the ML models, regularized linear methods track close to the benchmarks, and PLS stands out as particularly strong at $h = 2$. The persistently high RMSE of tree-based models reflects their significant forecast errors in 2024-2025, discussed earlier, rather than a fundamental weakness at longer horizons. Notably, the ML Ensemble underperforms relative to the Benchmark Ensemble across all horizons, which is mainly driven by the large errors of tree-based models pulling the ensemble average upward. At longer horizons ($h = 3$ and $h = 4$), regularized linear ML models narrow the gap with the benchmarks, although ARIMA and VAR continue to outperform all ML models at these horizons (see Figure 7).

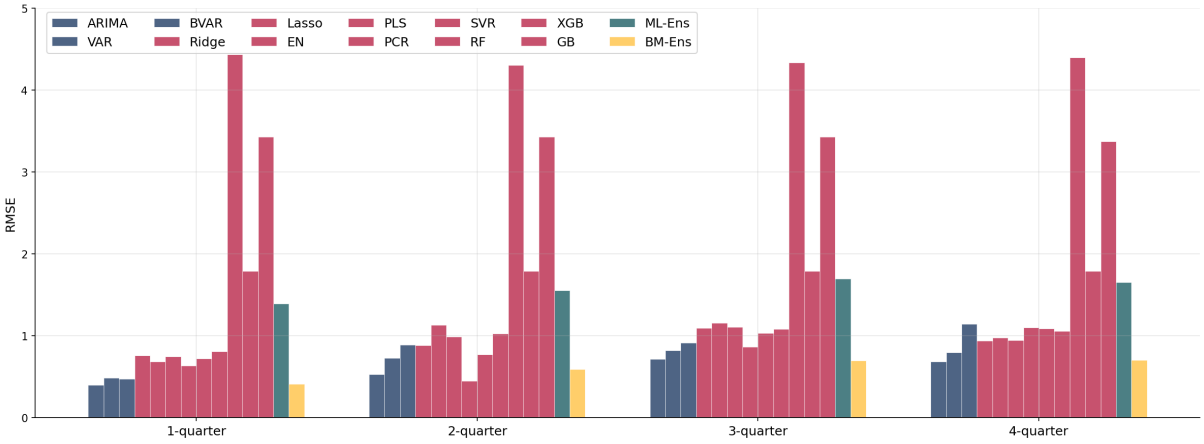


Figure 7: GDP forecast RMSE by model and horizon.

Note: ML-Ens means ML Ensemble, and BM-Ens means BM Ensemble

Taking the above into account, we test tree-based models under three different transformations of the target variable, that's GDP in levels, quarter-on-quarter (QoQ) changes, and year-on-year (YoY) changes.

This sensitivity to the choice of target transformation is consistent with Goulet Coulombe et al. (2021), who argue that data preprocessing can effectively alter the implicit regularization of machine learning algorithms, particularly for tree-based models such as Random Forest and Gradient Boosting. Because these models cannot extrapolate beyond the range of target values observed during training, a more stationary target can reduce approximation error when the level series goes outside its historical range.

To address this limitation, we train each tree-based model (Random Forest, XGBoost, and Gradient Boosting) on three versions of the target variable while keeping the predictors and the rolling-window setup unchanged throughout. All quantities are expressed as $100 \times$ log GDP, so a first difference is a growth rate in percentage points. The first version predicts the GDP level directly. The second predicts the QoQ change ($\Delta_1 y_t$), and the YoY growth forecast is obtained by summing four consecutive h-step-ahead QoQ forecasts, which equals exactly to $y_D - y_{D-4}$ in logs. The third predicts the YoY change ($\Delta_4 y_t$) directly. RMSE is evaluated on the resulting YoY growth series in every case. Since the predictors, the rolling-window setup, and the evaluation metric are identical across all three versions, any difference in RMSE reflects purely the effect of the target

transformation during training, providing a clean comparison of which transformation works best for each tree-based model.

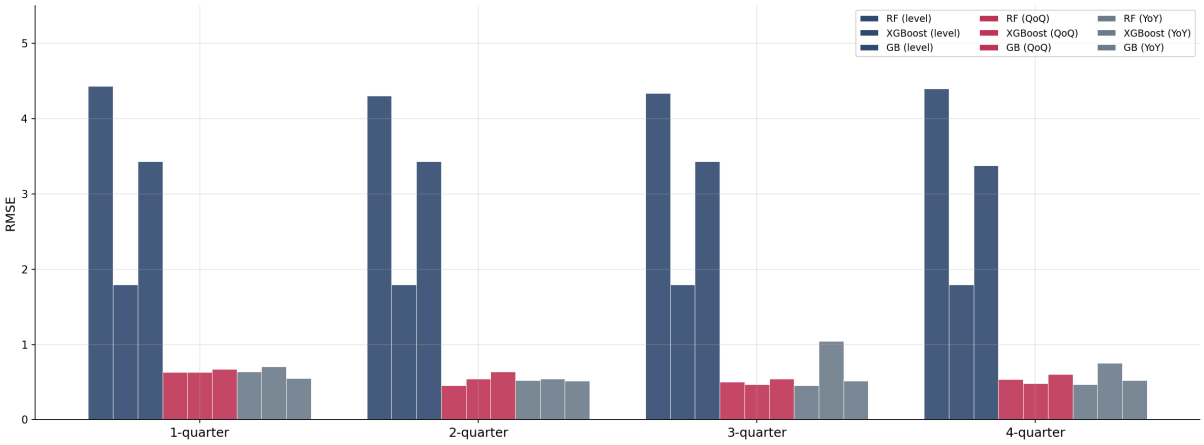


Figure 8: Tree-based models: RMSE by target transform and horizon.

The results of these transformations are shown in Figure 8. It can be clearly seen that applying the appropriate target transformation significantly improves the forecast performance of all three tree-based models. This is further confirmed in Appendix H, where the transformed forecasts now track the actual GDP growth path much more closely. These findings highlight a key limitation of tree-based models, their sensitivity to the stationarity and transformation of the target variable.

Table 3 presents the RMSE results for GDP growth forecasting across all four horizons. ARIMA is the strongest individual model at three of the four horizons, with RMSE ranging from 0.4018 at $h = 1$ to 0.6836 at $h = 4$. The only exception is $h = 2$, where PLS (0.4486) outperforms ARIMA (0.5300). VAR performs consistently but slightly below ARIMA throughout. BVAR deteriorates as the horizon extends, reaching an RMSE of 1.1463 at $h = 4$, though the deterioration is less severe than for the ML Ensemble, suggesting that the Minnesota prior becomes progressively less informative though not entirely uncompetitive at longer horizons. The Benchmark Ensemble outperforms the ML Ensemble at every horizon by a wide margin, which, as discussed above, reflects the large errors of tree-based models inflating the ML Ensemble average rather than a general inferiority of ML methods.

Among individual ML models, dimension-reduction and regularized linear methods are the most competitive, while tree-based models perform poorly for GDP. PLS is the strongest ML model at the three shorter horizons, with RMSE of 0.6343, 0.4486, and 0.8674 at $h = 1$, $h = 2$, and $h = 3$. At $h = 2$ its RMSE of 0.4486 is the lowest of any model in the table, outperforming even ARIMA. At the longest horizon, however, PLS deteriorates to 1.1010, and the regularized linear models become the most competitive ML alternatives, with Ridge (0.9380) and Elastic Net (0.9491) recording the lowest ML RMSE at $h = 4$. Lasso is the weakest at the medium horizon among linear models, and SVR and PCR are mid-ranked throughout. The tree-based models (Random Forest, XGBoost, and Gradient Boosting) produce RMSE values several times larger than the linear models at

every horizon and show little variation across horizons, consistent with their inability to extrapolate beyond the range of GDP growth observed in training. Overall, the results indicate that for GDP growth forecasting the benchmark models such as ARIMA in particular retain a clear advantage, while among ML methods PLS is competitive at short horizons and regularized linear models become the strongest ML alternatives as the horizon extends.

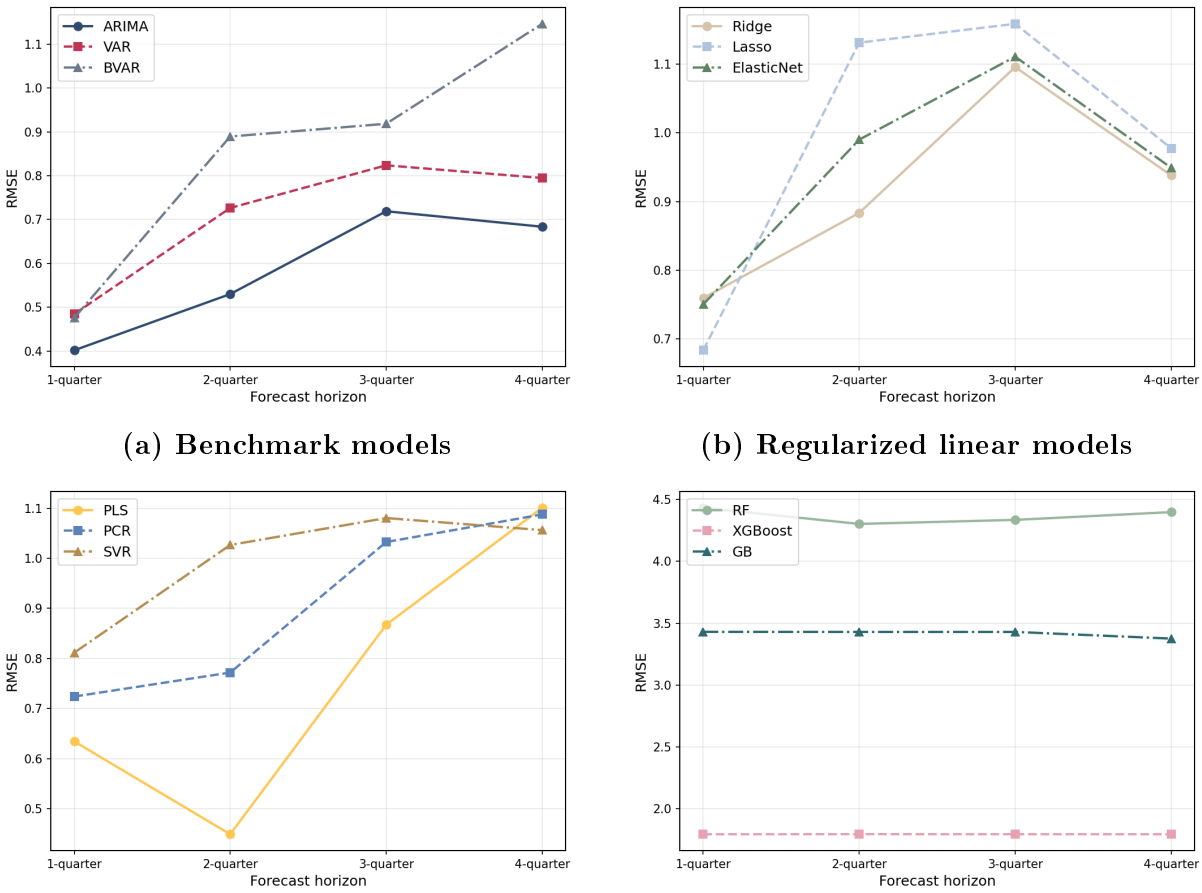
Table 3: RMSE - GDP Growth Forecasting

Model	$h = 1$	$h = 2$	$h = 3$	$h = 4$
Benchmark models				
ARIMA	0.4018	0.5300	0.7188	0.6836
VAR	0.4845	0.7263	0.8236	0.7947
BVAR	0.4757	0.8893	0.9184	1.1463
Machine learning models				
Ridge	0.7593	0.8834	1.0958	0.9380
Lasso	0.6833	1.1313	1.1588	0.9775
Elastic Net	0.7500	0.9901	1.1109	0.9491
PLS	0.6343	0.4486	0.8674	1.1010
PCR	0.7239	0.7718	1.0327	1.0878
SVR	0.8116	1.0268	1.0806	1.0565
RF	4.4335	4.3033	4.3360	4.3990
XGBoost	1.7923	1.7931	1.7926	1.7926
GB	3.4301	3.4294	3.4294	3.3754
Combination forecasts				
ML Ensemble	1.3930	1.5562	1.7009	1.6542
BM Ensemble	0.4100	0.5921	0.6950	0.7072

Notes: RMSE values for GDP growth forecasts across horizons $h = 1, 2, 3, 4$ quarters. ML Ensemble and BM Ensemble denote the RMSE of the simple-average forecast of the nine ML models and three benchmark models respectively. Lower RMSE is better.

Figure 9 shows the RMSE trajectories by model group across horizons. Unlike the CPI results, benchmark model performance in the GDP setting deteriorates only moderately with the forecast horizon. ARIMA and VAR remain close to each other throughout, and BVAR, while somewhat less accurate, does not diverge sharply from the other benchmarks.

Among ML models, regularized linear methods show the opposite pattern, with RMSE decreasing significantly from $h = 3$ onwards, narrowing the gap with the benchmarks considerably at longer horizons. The dimension-reduction methods and non-linear ML models are a notable exception. RMSE of non-linear ML models remains nearly constant across all four horizons, which at first glance may suggest stability, but for tree-based models in fact reflects the structural extrapolation limitation discussed above. However, the RMSE of dimension-reduction methods (PLS and PCR) deteriorates slightly as the horizon increases.



(a) Benchmark models

(b) Regularized linear models

(c) Dimension reduction models

(d) Non-linear ML models

Figure 9: GDP forecast RMSE by model group across horizons.

Overall, the figure highlights the conclusion that benchmark models, particularly ARIMA, remain the most reliable choice for GDP forecasting across all horizons, while tree-based models are held back not by horizon-specific weaknesses but by a fundamental inability to generalize beyond the range of values observed during training. Among the ML methods, PLS is the most competitive at short horizons, matching or outperforming the benchmarks at $h = 2$, while regularized linear and dimension-reduction models close much of the gap with the benchmarks at longer horizons.

The GDP results show a pattern similar to the CPI results. In both cases, the relative performance of machine learning improves as the forecast horizon becomes longer. However, unlike CPI, the benchmark models still perform better than ML models for GDP at every horizon.

6.3 Discussion of the performance differences between CPI and GDP models

The difference between the CPI and GDP results is clear. Machine learning, particularly the ML ensemble, delivers clear and increasing gains for inflation forecasting, whereas

traditional benchmarks, especially ARIMA, remain the most accurate choice for GDP across all forecast horizons. This difference can be explained by fundamental differences in the statistical properties of the two forecasting problems.

First, the two targets differ substantially in how much predictive information is contained in their own past values rather than in external indicators. The benchmark GDP model selects a near-random-walk specification with strong seasonal persistence (Appendix B), suggesting that much of the predictive information is already contained in GDP's own recent history. In contrast, the benchmark CPI model is more complex and exhibits no exploitable seasonality, indicating that a larger share of its predictable variation is contained in other predictors. When most of the predictive signal is already captured by the target's own dynamics, adding a large set of predictors is more likely to introduce estimation noise than useful information. Conversely, when the target cannot be predicted well from its own history alone and useful information is spread across many other indicators, machine learning can exploit this richer information set more effectively than traditional linear models.

Second, the two targets differ in both sample size and structural stability. CPI models are estimated using a 72-month rolling window drawn entirely from the post-2018 period, providing a comparatively homogeneous estimation sample. In contrast, GDP models are estimated using a shorter 44-quarter rolling window drawn from a dataset spanning 2012 onward, which covers both the pre- and post-2017 exchange-rate liberalization periods as well as other shocks (e.g. the COVID-19, structural changes, Russian-Ukraine and so on) and subsequent recovery. This greater structural instability required the exponential sample weighting described in Section 5. As a result, GDP models are estimated from less data and under more challenging conditions than CPI models, regardless of the forecasting method used.

Moreover, part of the explanation may also lie in the difference in data frequency. GDP is observed quarterly, whereas CPI is observed monthly. Quarterly aggregation tends to smooth high-frequency fluctuations, making GDP more persistent and increasing the predictive content of its own past values. In contrast, monthly CPI retains more short-term variation and exhibits a more complex dynamic structure, leaving greater scope for external predictors to improve forecast accuracy.

The benchmark models themselves provide further evidence for above interpretations. Among ARIMA, VAR, and BVAR, forecast accuracy for GDP generally deteriorates as the information set grows. ARIMA, which uses no external predictors, is the most accurate benchmark at nearly every horizon, while BVAR, despite drawing on many macroeconomic indicators, is typically the least accurate of the three, particularly at longer horizons. This suggests that, for GDP, the target's own history already contains most of the predictive information, and that adding a large number of predictors is more likely to introduce estimation noise than real signal. By contrast, the three CPI benchmarks remain relatively close at the short and medium horizons ($h = 1, 3, 6$), with BVAR falling behind only at the longest horizons ($h = 9, 12$). This suggests that the cost of a richer information set is much smaller for CPI than for GDP[23]

[23]This interpretation is also supported by the performance of the BVAR and some other ML models, whose forecast accuracy is generally similar. Together, these results suggest that for GDP, expanding

Finally, the extrapolation failure of tree-based models is consistent with the different characteristics of the GDP and CPI forecasting problems. GDP rose record levels during 2024-2025 following a prolonged period of slower and more volatile growth. Because tree-based models cannot extrapolate beyond the range of values observed during training, they systematically underestimated this acceleration[24]. This limitation is particularly severe when the training sample is both short and structurally unstable. In contrast, CPI followed a smoother trajectory during the evaluation period and this is why tree-based models did not face the same extrapolation problem.

Together, these differences suggest that the advantage of machine learning over traditional benchmarks is not universal. Rather, it depends on how much useful predictive information lies outside the target's own history and on the size and stability of the estimation sample.

7 Conclusion

This paper provides the first systematic assessments of machine learning methods for macroeconomic forecasting in Uzbekistan, comparing nine ML models against three traditional benchmarks (ARIMA, VAR, and BVAR) for forecasting CPI inflation and GDP growth across multiple horizons. Using a rich dataset of more than 170 domestic, international, financial, and alternative indicators, the study evaluates forecast performance through a rolling-window design with horizon-specific feature selection and hyperparameter tuning.

The results do not support a simple "machine learning is better" conclusion, and the more interesting finding is that the answer depends heavily on the target variable and the forecast horizon. For inflation, machine learning delivers clear gains. The benchmark models remain competitive only at the shortest horizon. As the horizon increases, the accuracy of benchmarks deteriorate sharply, while the machine learning models remain stable and in several cases improve. A simple equal-weighted average of the nine machine learning models is the most reliable strategy of all, achieving the lowest RMSE at nearly every horizon and outperforming both the best individual model and the benchmark combination by a wide margin. Among the individual models, the type of regularization that works best shifts with the horizon, sparse selection (Lasso) is strongest for near-term inflation, where only a few predictors carry signal, while models that shrink rather than eliminate coefficients (Ridge and Elastic Net) take over at longer horizons, where the predictive signal becomes more diffuse across a larger set of correlated variables. This pattern, together with the strength of the ensemble, suggests that forecasting inflation in Uzbekistan over longer horizons benefits from more information rather than less, precisely the setting in which machine learning holds an advantage over low-dimensional benchmarks.

For GDP growth the picture is very different. Here the traditional models, and ARIMA

the information set beyond the target's own history provides little additional predictive value and may instead introduce estimation noise.

[24]This is one reason why the ML ensemble performs substantially worse for GDP.

in particular, remain the most accurate choice at nearly every horizon, and no machine learning model beats them consistently. The machine learning methods are nonetheless competitive at short horizons, where PLS is the standout performer and even outperforms ARIMA at the two-quarter horizon, and the regularized linear models close much of the gap at longer horizons. The clearest weakness is the tree-based models, which perform poorly for GDP throughout. This is not a horizon-specific failure but a structural one. Because Random Forest, XGBoost, and Gradient Boosting cannot predict beyond the range of values seen in training, they systematically undershot the sharp acceleration in GDP growth during 2024-2025, which had no precedent in the weak and volatile growth of the preceding years. We confirmed this interpretation by re-estimating the tree models on stationary transformations of the target. We forecasted quarter-on-quarter or year-on-year changes of GDP rather than the level. It substantially improved the forecast accuracy, which shows that the problem lies in the extrapolation limits of the algorithm and the choice of target transformation, not in the models' ability to capture the underlying dynamics. This same limitation is why forecast combination helps for inflation but hurts for GDP, where the large errors of the tree models make worse the machine learning ensemble below the benchmark combination.

Taken together, these findings carry two main implications for the Central Bank of Uzbekistan. First, machine learning is best viewed as a complement to the existing toolkit rather than a replacement for it. It offers a clear improvement for medium-term inflation forecasting, where the current framework has struggled most, but it does not displace ARIMA and the other benchmarks for GDP. Second, model choice cannot be separated from how the target is measured and transformed. The tree-based results show that a mechanical application of a flexible algorithm to a trending, non-stationary target can produce large and avoidable errors, and that simple preprocessing decisions matter as much as the choice of algorithm itself.

These findings should be read with the limitations of the exercise in mind. The available time series for Uzbekistan are short, especially for GDP, which forces a correspondingly short out-of-sample evaluation window and limits the statistical power of formal tests of predictive accuracy. For example, the Diebold-Mariano comparisons, in particular, rarely reject the null of equal accuracy for this reason. Data quality and definitional changes around the 2017 liberalization further complicate estimation, and although we address this through sample weighting and rolling windows, the underlying structural instability cannot be fully removed. Our conclusions are also confined to point forecasts. We do not assess the models' performance in producing calibrated density or interval forecasts, which are central to risk-based policy analysis.

A further extension concerns the transformation of the input data. In this study, the models are estimated to forecast the target in (log) levels, with the exception of the tree-based models for GDP, where we showed that forecasting stationary transformations of the target substantially improves accuracy. Given how much this single change mattered for the tree models, a systematic evaluation of alternative target and predictor transformations across all model classes, rather than levels alone, is a natural next step and may yield further gains.

These limitations point naturally to future work. As the post-2017 sample increases,

the evaluation window will widen and the machine learning models, which are more data-hungry, should be re-assessed under more reliable conditions. The high-frequency and alternative indicators assembled here, including Google Trends and web-scraped job-posting data, are particularly well suited to nowcasting, which we leave for future research and which could feed directly into the CBU's Forecasting and Policy Analysis System (FPAS). Longer forecast horizons could also be considered, extending the analysis beyond the one-year horizon studied here. More broadly, the results suggest that the most promising path for the CBU is not to adopt machine learning wholesale, but to fold a small number of well-chosen methods, most obviously the machine learning ensemble for medium-term inflation, into the existing framework, while retaining the traditional models where they remain hard to beat. Despite these limitations, the results demonstrate that machine learning methods hold considerable promise for improving macroeconomic forecasting in Uzbekistan and, more broadly, in small open economies undergoing rapid structural transformation.

Data and code availability

The code that implements the models and reproduces the results reported in this paper is available at https://github.com/abu-rahmon/inf-gdp-uzb-forecast. All models except the BVAR are implemented in Python. The BVAR is estimated in MATLAB, and both the Python and MATLAB code are included in the repository. The data are obtained from the sources listed in Table 1. Series from the Central Bank of Uzbekistan and other official providers are available from those institutions, and some third-party series are subject to licensing restrictions and cannot be redistributed.

References

Abdi, H. (2010). Partial least squares regression and projection on latent structure regression (PLS Regression). *WIREs Computational Statistics*, 2(1):97–106.

Ahn, S. C. and Bae, J. (2022). Forecasting with Partial Least Squares When a Large Number of Predictors Are Available. *SSRN Electronic Journal*.

Arthur F. Burns and Wesley C. Mitchell (1946). *Measuring Business Cycles*. NBER.

Babii, A., Ghysels, E., and Striaukas, J. (2020). Machine Learning Time Series Regressions with an Application to Nowcasting. arXiv:2005.14057 [econ.EM].

Bai, J. and Ng, S. (2008). Forecasting economic time series using targeted predictors. *Journal of Econometrics*, 146(2):304–317.

Bańbura, M., Giannone, D., Lenz, M., and European Central Bank, editors (2014). *Conditional forecasts and scenario analysis with vector autoregressions for large cross sections: N°1733, September 2014*. European Central Bank, Frankfurt am Main.

Bańbura, M., Giannone, D., and Reichlin, L. (2010). Large Bayesian vector auto regressions. *Journal of Applied Econometrics*, 25(1):71–92. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jae.1137.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

CBU (2024). Monetary policy guidelines for 2025 and 2026-2027.

Chan-Lau, J. A. (2017). Lasso Regressions and Forecasting Models in Applied Stress Testing. *IMF Working Papers*.

Chapman, J. T. E. and Desai, A. (2023). Macroeconomic Predictions using Payments Data and Machine Learning. *Forecasting*, 5(4):652–683. arXiv:2209.00948 [econ.GN].

Chen, T. and Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. Association for Computing Machinery.

Chi, T.-C., Fan, T.-H., Ghigliazza, R. M., Giannone, D., Zixuan, and Wang (2025). Macroeconomic Forecasting and Machine Learning. arXiv:2510.11008 [econ.EM] version: 1.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Diebold, F. and Mariano, R. (2002). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, 20:134–44.

Flores, J., Gonzaga, B., Ruelas-Huanca, W., and Tang, J. (2025). Nowcasting Peru's GDP with Machine Learning Methods. *Geneva Graduate Institute*.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5).

George E.P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung (2016). *Time series analysis: Forecasting and control*, volume 5. John Wiley & Sons, United States of America.

Giraitis, L. and Kapetanios, G. (2012). Adaptive Forecasting in the presence of recent and ongoing structural change. *Discussion Paper Series*.

Goulet Coulombe, P., Leroux, M., Stevanovic, D., and Surprenant, S. (2021). Macroeconomic data transformations matter. *International Journal of Forecasting*, 37(4):1338–1354.

Goulet Coulombe, P., Leroux, M., Stevanovic, D., and Surprenant, S. (2022). How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics*, 37(5):920–964.

Groen, J. J. and Kapetanios, G. (2009). Revisiting Useful Approaches to Data-Rich Macroeconomic Forecasting.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies*, 33(5):2223–2273.

Haris Karagiannakis (2025). Real-Time GDP Nowcasting in a Data-Rich Mixed-Frequency Environment: Harnessing Machine Learning. *King's College London*.

Koop, G. M. (2013). Forecasting with Medium and Large Bayesian VARS. *Journal of Applied Econometrics*, 28(2):177–203. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jae.1270.

Kotchoni, R., Leroux, M., and Stevanovic, D. (2019). Macroeconomic forecast accuracy in a data-rich environment. *Journal of Applied Econometrics*, 34(7):1050–1072.

Kraskov, A., Stoegbauer, H., and Grassberger, P. (2004). Estimating Mutual Information. *Physical Review E*, 69(6):066138. arXiv:cond-mat/0305641.

Li, J. and Chen, W. (2014). Forecasting macroeconomic time series: LASSO-based approaches and their forecast combinations with dynamic factor models. *International Journal of Forecasting*, 30(4):996–1015.

Litterman, R. B. (1979). Techniques of Forecasting Using Vector Autoregressions.

Litterman, R. B. (1985). Forecasting with Bayesian vector autoregressions five years of experience. *Working Papers*. Number: 274.

Liu, Y. (2024). Mending the Crystal Ball: Enhanced Inflation Forecasts with Machine Learning. *IMF Working Papers*, 2024(206):1.

Lütkepohl, H. (2005). *New introduction to multiple time series analysis.* New York : Springer, Berlin.

Maehashi, K. and Shintani, M. (2020). Macroeconomic forecasting using factor models and machine learning: an application to Japan. *Journal of the Japanese and International Economies*, 58:101104.

Medeiros, M. C. and Vasconcelos, G. F. R. (2019). Forecasting Inflation in A Data-Rich Environment: The Benefits of Machine Learning Methods. *Journal of Business & Economic Statistics*, 39(1):98–119.

Pesaran, M. H. and Timmermann, A. (2007). Selection of estimation window in the presence of breaks. *Journal of Econometrics*, 137(1):134–161.

Reichlin, L., De Mol, C., Gautier, E., Giannone, D., Mullainathan, S., van Dijk, H., and Wooldridge, J. (2017). Big data in economics: evolution or revolution? In Matyas, L., editor, *Big data in economics: evolution or revolution?*, pages 612–632. Cambridge University Press, Cambridge.

Richardson, A., Mulder, T., and L Vehbi, T. (2018). Nowcasting New Zealand GDP using machine learning algorithms. *SSRN Electronic Journal*.

Sims, C. A. (1980). Macroeconomics and Reality. *Econometrica*, 48(1):1.

Slimani, H. and Chourabi, C. (2025). Modeling inflation with machine learning: a cross-horizon systematic review. *International Journal of Data Science and Analytics*, 21.

Stock, J. H. and Watson, M. W. (2001). Vector Autoregressions. *Journal of Economic Perspectiv*, 15.

Stock, J. H. and Watson, M. W. (2007). Why Has U.S. Inflation Become Harder to Forecast? *Journal of Money, Credit and Banking*, 39(s1):3–33.

Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288.

Vapnik, V. N. (2013). *The Nature of Statistical Learning Theory*, volume 2. Springer Science & Business Media. Google-Books-ID: EoDSBwAAQBAJ.

Wang, C., Zhou, Y., Wen, Q., and Wang, Y. (2023). Improving Load Forecasting Performance via Sample Reweighting. *IEEE Transactions on Smart Grid*, 14(4):3317–3320.

Zhang, B., Chan, J. C. C., and Cross, J. L. (2020). Stochastic volatility models with ARMA innovations: An application to G7 inflation forecasts. *International Journal of Forecasting*, 36(4):1318–1328.

Zou, H. and Hastie, T. (2005). Regularization and Variable Selection Via the Elastic Net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320.

Appendices

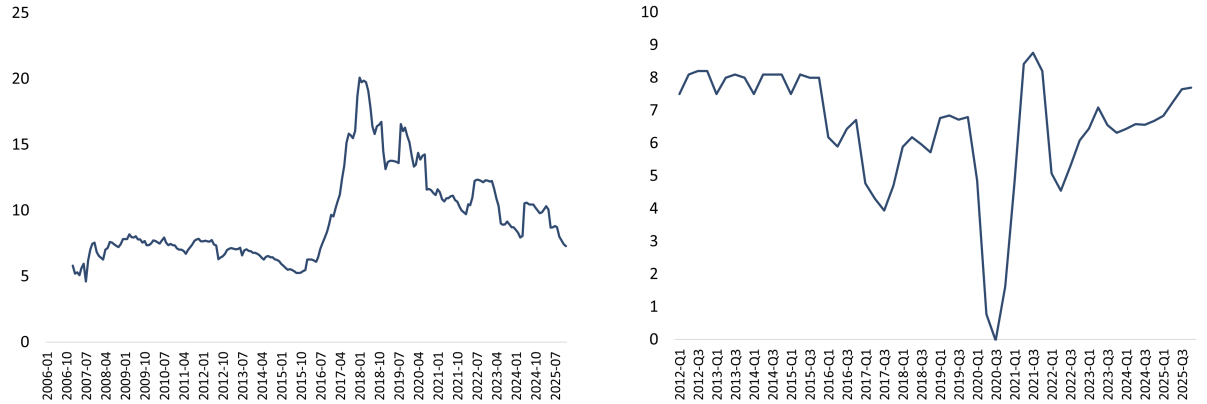
A Historical inflation and GDP growth in Uzbekistan

From independence in 1991 until 2017, Uzbekistan's economy operated under extensive state control. Administratively fixed exchange rates coexisted with an active black market, foreign currency conversion was tightly restricted, and prices in key sectors were subject to direct state intervention. This regime changed fundamentally with the September 2017 foreign exchange liberalization, when the Central Bank of Uzbekistan unified the official and parallel (black market) exchange rates and allowed the soum to float, resulting in an immediate devaluation of approximately 50 percent (from about 4,200 to 8,100 som per US dollar). The reform was accompanied by tariff reductions, banking sector and price reforms, and the gradual liberalization of foreign currency transactions through 2019.

This reform represents the structural break referenced throughout this paper. It directly affected import prices and, through this channel, inflation, while altering the relationships among many macroeconomic variables. These changes motivate both the shorter CPI estimation sample (beginning in 2018) and the exponential sample weighting applied to the longer GDP sample.

Two additional shocks are visible in Figure 10. The COVID-19 pandemic caused a sharp slowdown in GDP growth in 2020, followed by a strong recovery. The Russia-Ukraine war, beginning in 2022, introduced additional volatility through trade and remittance channels, reflecting Uzbekistan's close economic ties with Russia and the large number of Uzbek migrant workers employed there.

Figure 10 presents the historical evolution of inflation and GDP growth in Uzbekistan over the sample period, highlighting the structural breaks and external shocks that motivate the use of machine learning methods.



(a) Inflation **(b) GDP growth**

Figure 10: Historical Overview: Inflation and GDP Growth

Source: National Committee on Statistics.

B Benchmark models estimation details

ARIMA

The order (p, d, q) and seasonal order (P, D, Q, m) are selected once, on the initial estimation window, using `auto_arima` from the `pmdarima` package in Python, applied to the $100 \times \log$ level series (matching the transformation used throughout the paper).

The search is stepwise, minimizing the Akaike Information Criterion (AIC), over the ranges in Table 4. The selected orders, uniform across all horizons within each target, are reported in Table 5. The differencing order d is selected automatically to achieve stationarity, so no separate stationarity testing or manual differencing is applied beforehand. At each step of the expanding window, the model's coefficients are re-estimated on all data available up to that point using the fixed order selected on the initial window, and the model forecasts h periods ahead directly, matching the forecast horizon under evaluation. Forecasted levels are converted to year-on-year terms for evaluation, consistent with the reporting convention used for all other models.

Table 4: ARIMA Search Ranges

Parameter	**Search Range**
p (AR order)	$\{1, \ldots, 6\}$
d (differencing)	$\{0, 1, 2\}$
q (MA order)	$\{0, \ldots, 6\}$
P (seasonal AR)	$\{0, 1, 2\}$
D (seasonal differencing)	$\{0, 1\}$
Q (seasonal MA)	$\{0, 1, 2\}$

Table 5: Selected ARIMA Orders

Target	**Order** (p, d, q)	**Seasonal Order** (P, D, Q, m)
CPI $(h = 1, 3, 6, 9, 12)$	$(0, 2, 1)$	$(0, 0, 0, 12)$
GDP $(h = 1, 2, 3, 4)$	$(0, 1, 0)$	$(0, 1, 1, 4)$

Notes: Orders are identical across all forecast horizons within each target. For CPI, the seasonal order is uniformly zero, so the fitted model reduces to the non-seasonal form in Equation (1). For GDP, a seasonal difference and seasonal moving-average term at lag 4 are selected, reflecting quarterly seasonality in GDP growth not present in the year-on-year CPI series.

For CPI ($m = 12$), the selected seasonal order is uniformly $(0, 0, 0, 12)$ across all horizons ($h = 1, 3, 6, 9, 12$), so the fitted model reduces to the non-seasonal form given in Equation (1), with non-seasonal order $(p, d, q) = (0, 2, 1)$. For GDP ($m = 4$), the selected specification includes a seasonal difference and a seasonal moving-average term, with order

$(p, d, q) = (0, 1, 0)$ and seasonal order $(P, D, Q, 4) = (0, 1, 1, 4)$, reflecting seasonal patterns in quarterly GDP growth not present in the year-on-year CPI series (Table 5).

VAR

In this study, the VAR is estimated with six variables, including the target (CPI or GDP), industrial production, the money market rate, the UZS/USD exchange rate, a monetary aggregate (M2 for CPI and M0 for GDP), and the producer price index. These variables capture the main channels relevant for inflation and output dynamics, namely real activity, monetary policy, exchange-rate pass-through, monetary conditions, and cost-push pressures.

The lag order is selected by minimizing the Akaike Information Criterion (AIC), with a maximum of 6 lags for CPI and 4 lags for GDP. VAR is estimated on an expanding window, as with ARIMA.

At each window, the fitted VAR forecasts the differenced series h periods ahead directly. The level forecast is reconstructed by adding the predicted cumulative difference to the last observed level of the target variable. Ninety-five percent confidence intervals are constructed as $\hat{y}_{t+h} \pm 1.96 \times \mathrm{SE}_h$, where SE_h is obtained from the model's h-step forecast error covariance.

BVAR

The degrees of freedom are set to $\nu = n + 2$. Under the Minnesota prior, the expected value and covariance of the VAR coefficients are

$$\mathbb{E}[(B_s)_{ij} \mid \Sigma, \lambda, \psi] = \begin{cases} \rho, & \text{if } i = j \text{ and } s = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{20}$$

$$\operatorname{cov}[(B_s)_{ij}, (B_r)_{hm} \mid \Sigma, \lambda, \psi] = \begin{cases} \lambda^2 \dfrac{1}{s^2} \dfrac{\Sigma_{ih}}{\psi_{jj}}, & \text{if } m = j \text{ and } s = r, \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

The expected value of the first-lag own-coefficient is set to ρ, while all other coefficients are shrunk toward zero, with prior uncertainty decreasing as the lag order s increases. The overall degree of shrinkage is controlled by a tightness hyperparameter, and Σ_{ih}/ψ_{jj} accounts for the relative scale of each variable.

The BVAR is estimated in MATLAB using the IRIS Toolbox. The BVAR includes a much richer information, same predictors set as the machine learning models. It is estimated with 2 lags, an overall Minnesota shrinkage of $\lambda = 0.10$, a prior mean on each variable's own first lag of $\rho = 0.7$, and a sum-of-coefficients prior tightness of $\mu = 0.5$. As with the other benchmarks, the same specification is used across all horizons, with only the estimation window adjusted.

C Hyperparameter search spaces

Table 6: Hyperparameter Search Space: CPI and GDP Models

Model	Hyperparameter	CPI Search Space	GDP Search Space
Ridge	λ	logspace$(-4, 4, 300)$	logspace$(-4, 4, 300)$
Lasso	λ	logspace$(-2, 2, 200)$	logspace$(-2, 2, 200)$
Elastic Net	λ ℓ_1 ratio	logspace$(-2, 2, 200)$ $\{0.01, 0.05, 0.1, \ldots, 0.9\}$	logspace$(-2, 2, 200)$ $\{0.01, 0.05, 0.1, \ldots, 0.9\}$
SVR	kernel C ϵ	linear (fixed) logspace$(-2, 2, 100)$ $\{0.01, 0.05, 0.1, 0.5, 1.0, 2.0\}$	linear (fixed) logspace$(-2, 2, 100)$ $\{0.01, 0.05, 0.1, 0.5, 1.0, 2.0\}$
PCR	`n_components` λ (Ridge)	$\{2, \ldots, 20\}$ logspace$(-2, 2, 50)$	90% variance threshold logspace$(-2, 2, 200)$
PLS	`n_components`	$\{1, \ldots, 15\}$	$\{1, \ldots, 5\}$

Random Forest	`n_estimators`
	`max_depth`
	`min_samples_split`
	`min_samples_leaf`
	`max_features`
		$\{100, 200, 300, 500\}$
		$\{5, 10, 20, 30, \text{None}\}$
		$\{2, 5, 10\}$
		$\{1, 2, 4\}$
		$\{\text{sqrt}, \text{log2}\}$
			$\{200, 300, 500\}$
			$\{3, 5, 7\}$
			$\{2, 5\}$
			$\{2, 4\}$
			$\{\text{sqrt}, \text{log2}, 0.5\}$

XGBoost	`n_estimators`
	`max_depth`
	`learning_rate`
	`subsample`
	`colsample_bytree`
	`reg_alpha`
	`reg_lambda`
		$\{100, 200, 300\}$
		$\{2, 3, 4, 6\}$
		$\{0.01, 0.05, 0.1, 0.2\}$
		$\{0.7, 0.9, 1.0\}$
		$\{0.7, 0.9, 1.0\}$
		$\{0, 0.1, 0.5\}$
		$\{0.5, 1.0, 2.0\}$
			$\{100, 200\}$
			$\{2, 3, 4\}$
			$\{0.01, 0.05, 0.1\}$
			$\{0.7, 1.0\}$
			$\{0.7, 1.0\}$
			$\{0, 0.1\}$
			$\{1.0, 2.0\}$

Gradient Boosting	`n_estimators`
	`max_depth`
	`learning_rate`
	`subsample`
	`min_samples_leaf`
	`max_features`
		$\{100, 200, 300\}$
		$\{2, 3, 4\}$
		$\{0.01, 0.05, 0.1\}$
		$\{0.7, 1.0\}$
		$\{3, 5, 8\}$
		$\{\text{sqrt}, \text{log2}\}$
			$\{100, 200\}$
			$\{2, 3, 4\}$
			$\{0.01, 0.05, 0.1\}$
			$\{0.7, 1.0\}$
			$\{3, 5\}$
			$\{\text{sqrt}, \text{log2}\}$

Notes: All models are tuned using `GridSearchCV` with `TimeSeriesSplit` ($n\_\text{splits} = 5$, $\text{test}\_\text{size} = 12$ for CPI; $n\_\text{splits} = 4$, $\text{test}\_\text{size} = 2$ for GDP) and RMSE as the selection criterion. λ in Ridge, Lasso, Elastic Net, and PCR corresponds to the `alpha` parameter in `scikit-learn`. logspace(a, b, n) denotes n values logarithmically spaced between 10^a and 10^b. For PCR, the number of components is chosen by grid search for CPI but by a 90% predictor-variance threshold for GDP, whose shorter training window cannot reliably support a grid of up to 20 components.

D Moving-block bootstrap for confidence intervals

For each model, target variable, and rolling window, hyperparameters are selected once by applying `GridSearchCV` to the original training sample $(X_{\text{train}}, y_{\text{train}})$ of size n, yielding the optimal hyperparameter vector $\hat{\theta}$ (e.g., α for Ridge, or `n_estimators` and `max_depth` for tree-based models). The model is then re-estimated on the original training sample using $\hat{\theta}$ to produce the point forecast $\hat{y}_{t+h}$ reported in the main text.

Constructing the bootstrap resamples. To quantify uncertainty around $\hat{y}_{t+h}$, the training data are resampled B times using a moving-block bootstrap, which preserves the local temporal dependence that would be destroyed by an ordinary i.i.d. bootstrap. For each bootstrap replicate $b = 1, \ldots, B$, blocks of L consecutive observations are sampled with replacement from the original training sample and concatenated to form a bootstrap dataset $(X^{(b)}, y^{(b)})$ of length n. The block length L is set to 12 months for CPI and 4 quarters for GDP.

Refitting and generating bootstrap forecasts. For each bootstrap replicate $b = 1, \ldots, B$, the model is refit on the bootstrap sample using the same hyperparameters $\hat{\theta}$ selected from the original training data, without re-tuning. For GDP models, exponentially decaying sample weights are recomputed for each bootstrap sample and reapplied during refitting, consistent with the estimation of the original point forecast. The only exception is PLS, whose implementation does not support sample weighting. CPI models are estimated without sample weights. The refitted model is then used to generate a bootstrap forecast for the test observation, denoted by $\hat{y}^{(b)}_{t+h}$ for $b = 1, \ldots, B$.

Constructing the interval. Let $\bar{y}^{*}_{t+h} = \frac{1}{B}\sum_{b=1}^{B} \hat{y}^{(b)}_{t+h}$ be the mean of the bootstrap forecasts, and let $d^{(b)} = \hat{y}^{(b)}_{t+h} - \bar{y}^{*}_{t+h}$ be the deviation of each bootstrap forecast from this mean. The 95% confidence interval is

$$[\, \hat{y}_{t+h} + Q_{0.025}(d),\ \ \hat{y}_{t+h} + Q_{0.975}(d) \,], \tag{22}$$

where $Q_p(d)$ denotes the p-th percentile of the bootstrap deviations $\{d^{(b)}\}_{b=1}^{B}$, and $\hat{y}_{t+h}$ is the point forecast from the model fit on the true (non-bootstrapped) training data.

Number of bootstrap replications. We use $B = 1{,}000$ bootstrap replications for all models except the tree-based ensemble methods (Random Forest, XGBoost, and Gradient Boosting), for which $B = 100$. This choice reflects the substantially greater computational cost of repeatedly refitting large ensembles of trees for every bootstrap replicate, rolling window, forecast horizon, and target variable.

E CPI forecasts by horizon

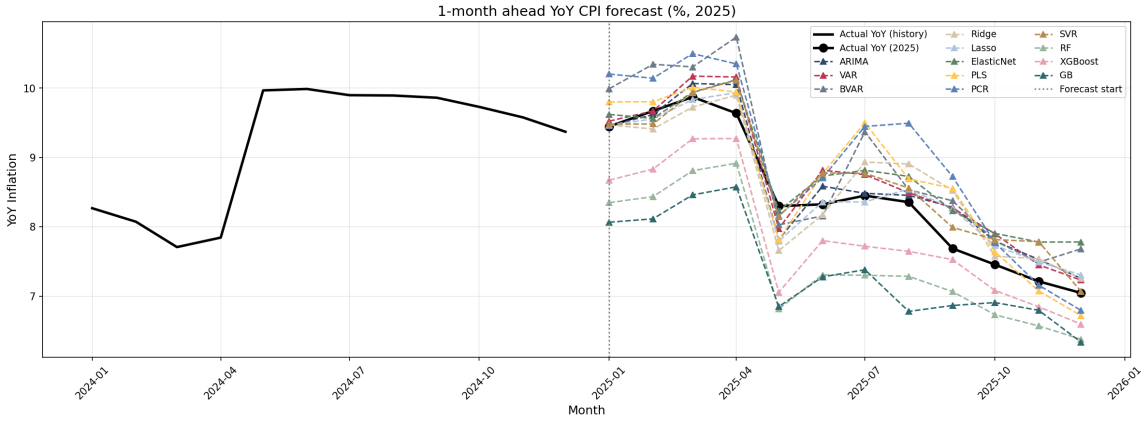


Figure 11: YoY CPI forecast, $h = 1$ month

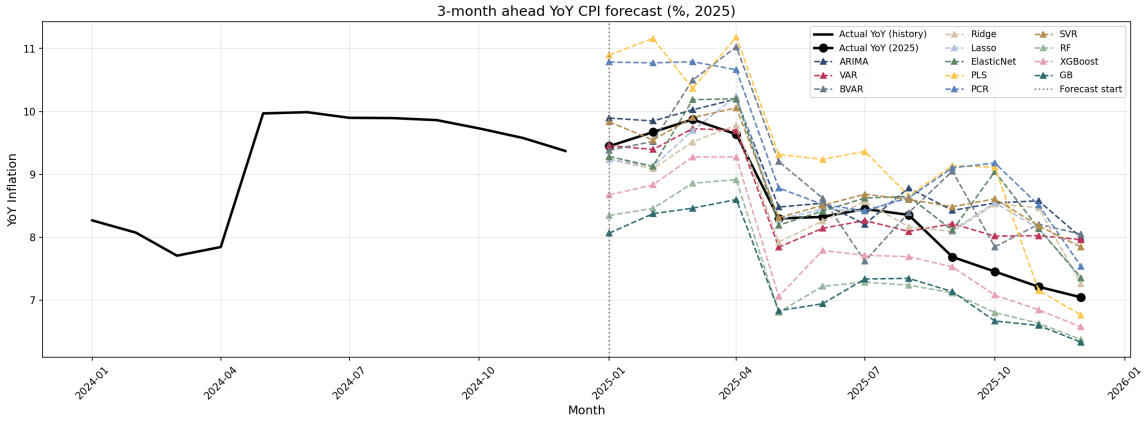


Figure 12: YoY CPI forecast, $h = 3$ month

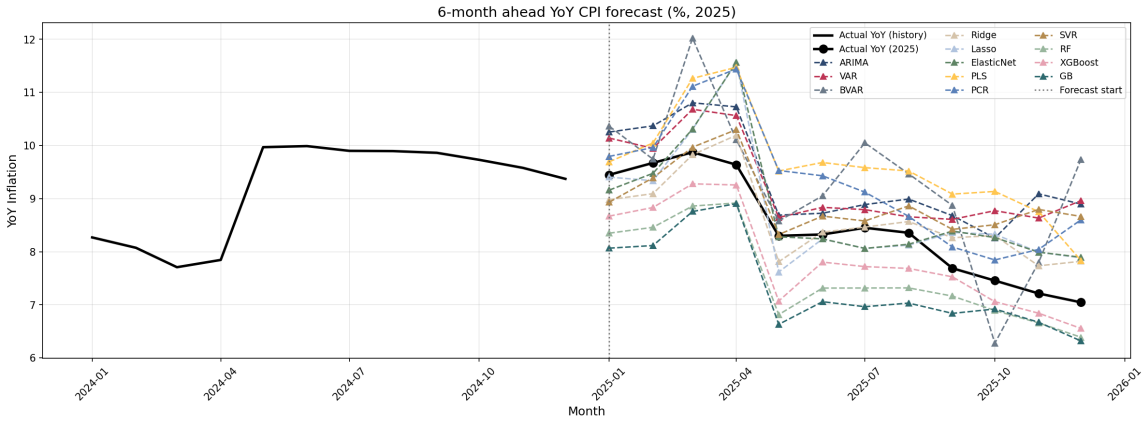


Figure 13: YoY CPI forecast, $h = 6$ month

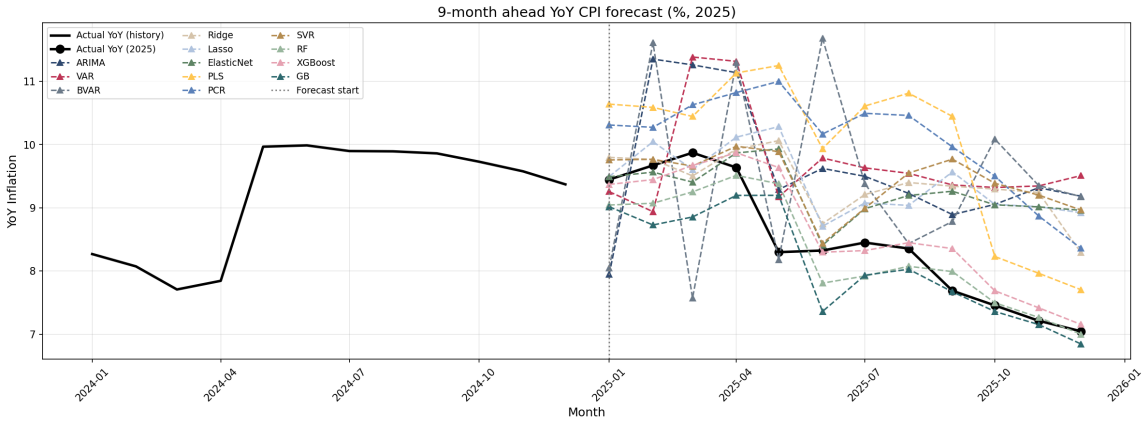


Figure 14: YoY CPI forecast, $h = 9$ month

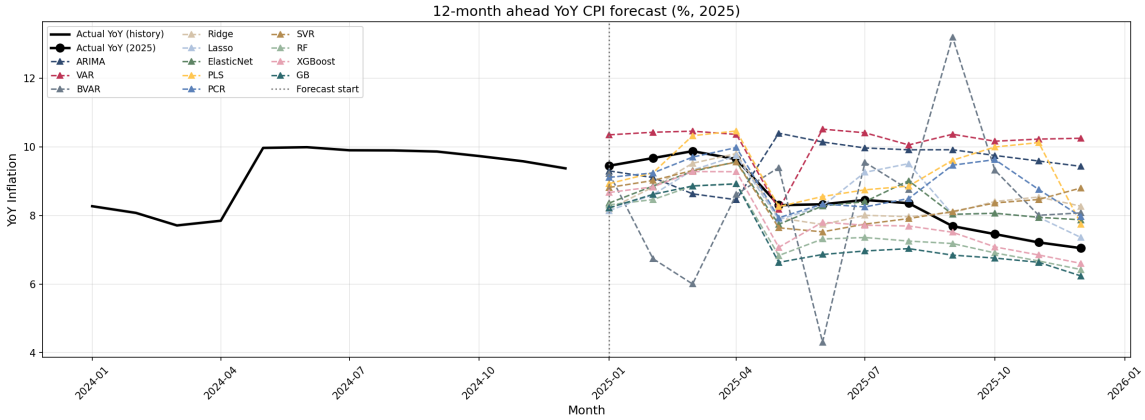


Figure 15: YoY CPI forecast, $h = 12$ month

F GDP forecasts by horizon

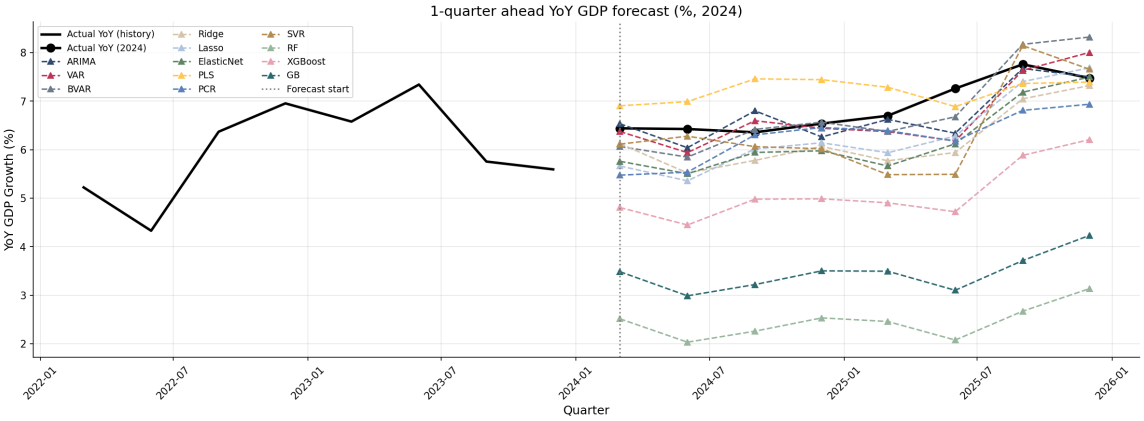


Figure 16: YoY GDP forecast, $h = 1$ quarter

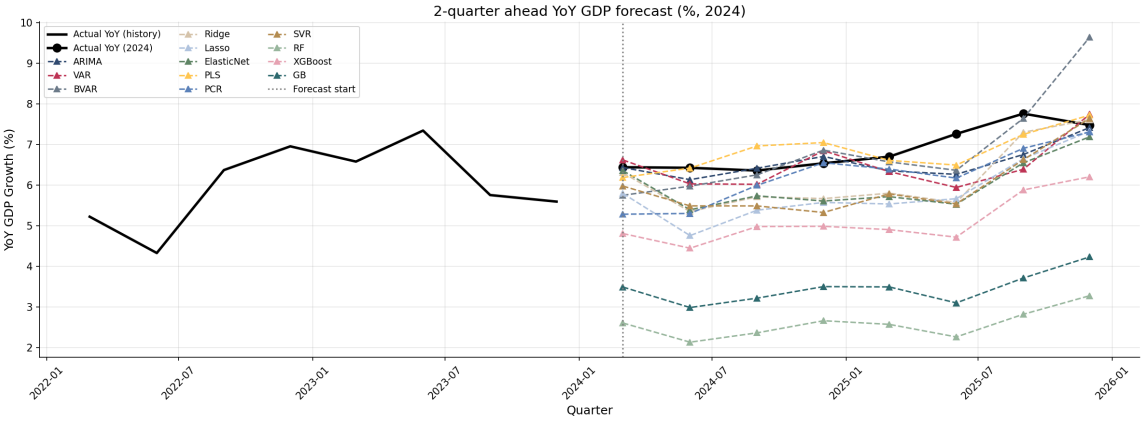


Figure 17: YoY GDP forecast, $h = 2$ quarter

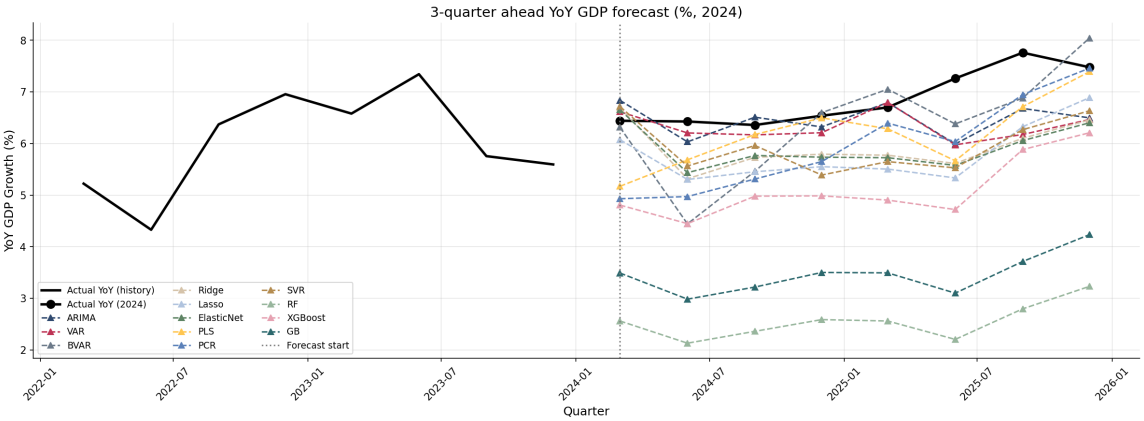


Figure 18: YoY GDP forecast, $h = 3$ quarter

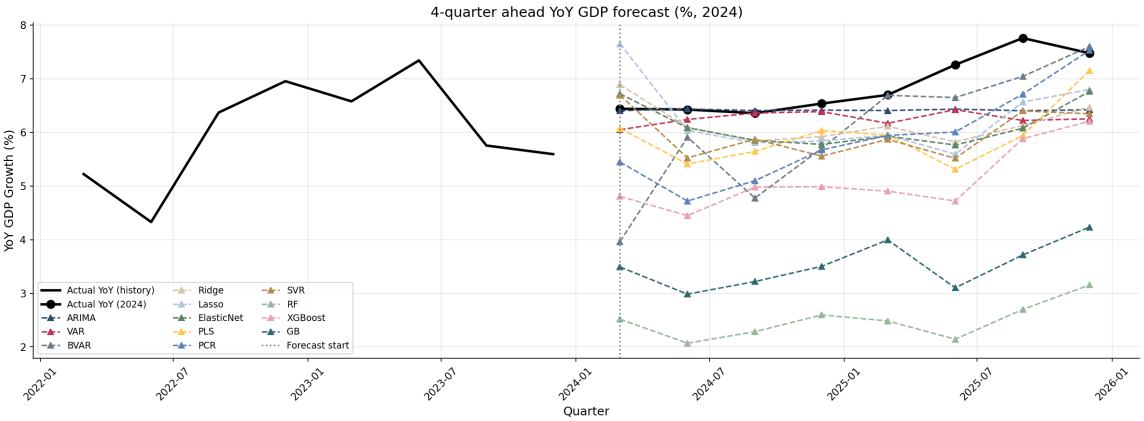
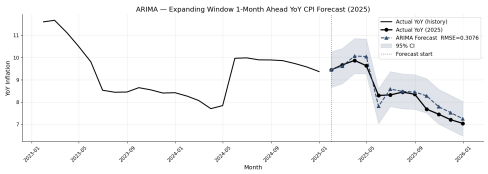


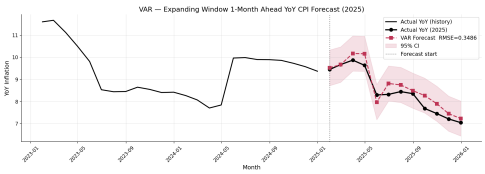
Figure 19: YoY GDP forecast, $h = 4$ quarter

G Forecast paths with confidence intervals

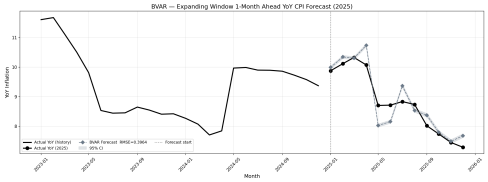
CPI forecast results



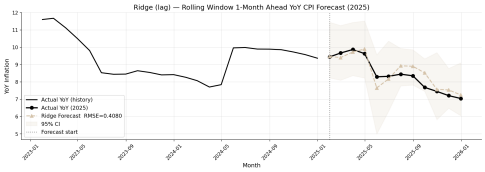
(a) **ARIMA**



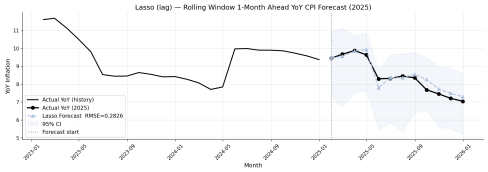
(b) **VAR**



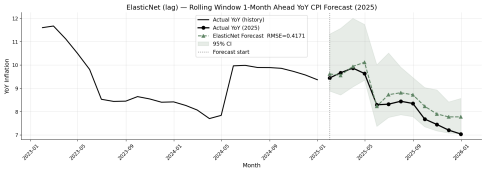
(c) **BVAR**



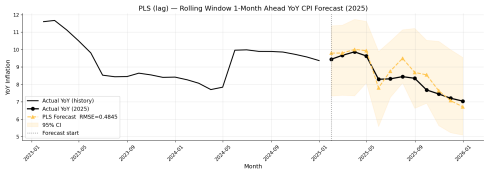
(d) **Ridge**



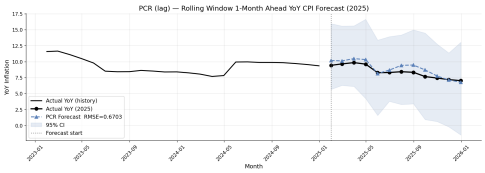
(e) **Lasso**



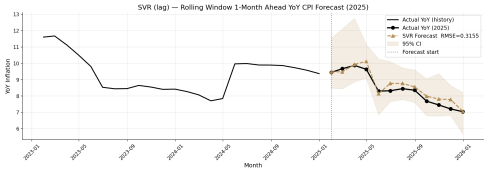
(f) **Elastic Net**



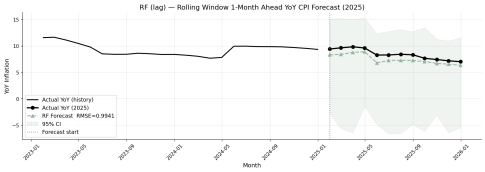
(g) **PLS**



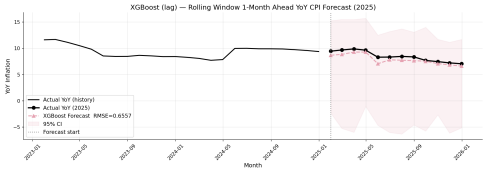
(h) **PCR**



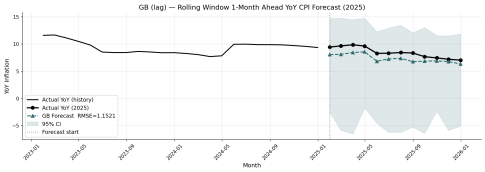
(i) **SVR**



(j) **Random Forest**



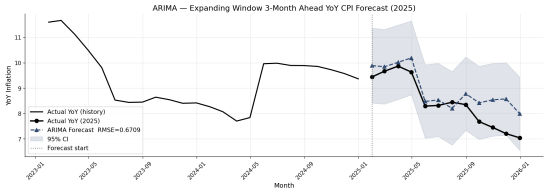
(k) **XGBoost**



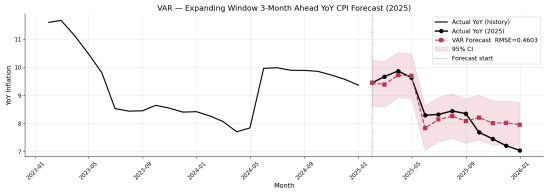
(l) **Gradient Boosting**

Figure 20: Out-of-sample YoY CPI inflation forecasts with 95% confidence intervals, $h = 1$ months ahead.

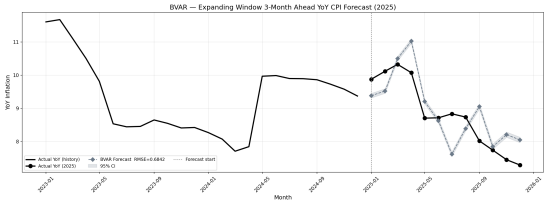
Note: Confidence intervals are constructed using a moving-block bootstrap. See Appendix D for details of the methodology.



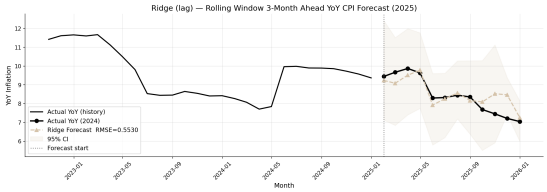
(a) **ARIMA**



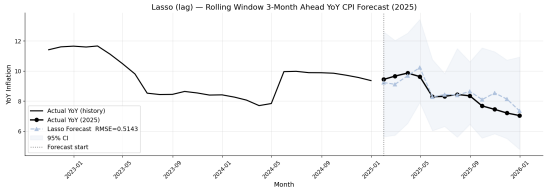
(b) **VAR**



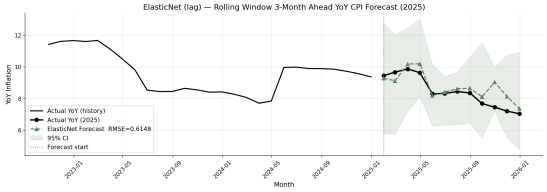
(c) **BVAR**



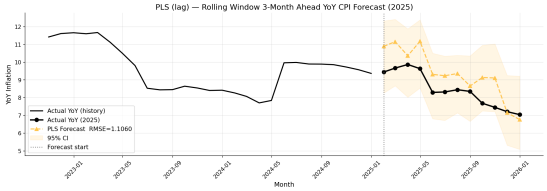
(d) **Ridge**



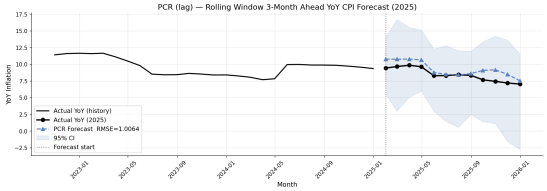
(e) **Lasso**



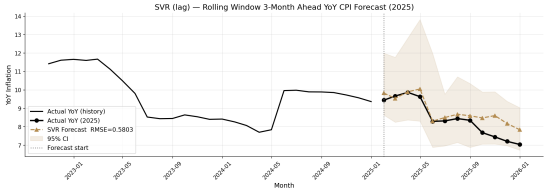
(f) **Elastic Net**



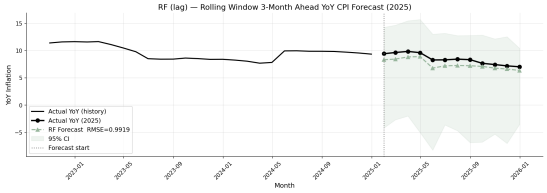
(g) **PLS**



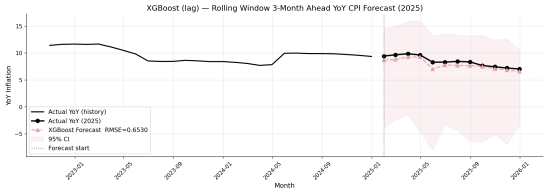
(h) **PCR**



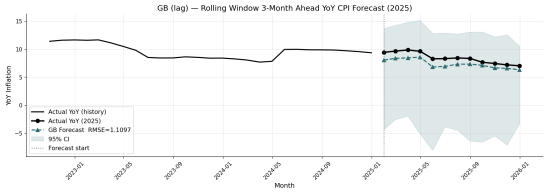
(i) **SVR**



(j) **Random Forest**



(k) **XGBoost**

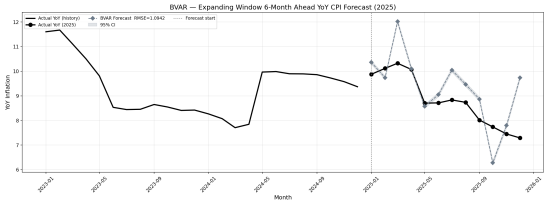


(l) **Gradient Boosting**

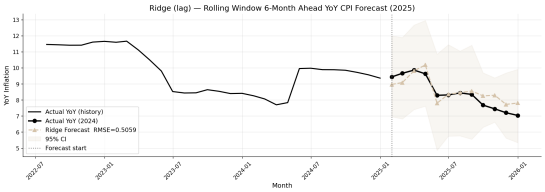
Figure 21: Out-of-sample YoY CPI inflation forecasts with 95% confidence intervals, $h = 3$ months ahead.

(a) **ARIMA**

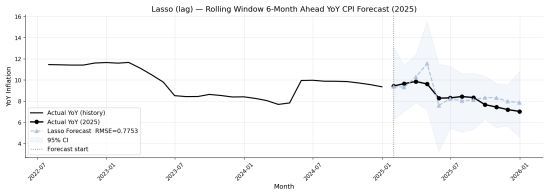
(b) **VAR**



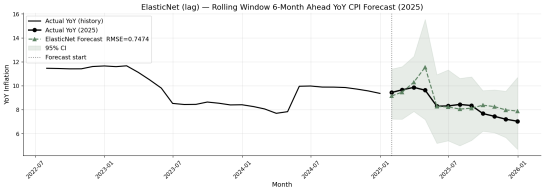
(c) **BVAR**



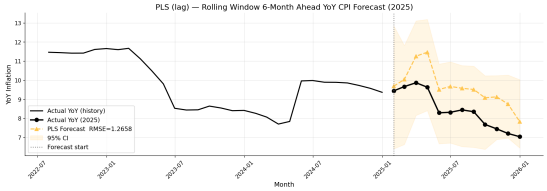
(d) **Ridge**



(e) **Lasso**

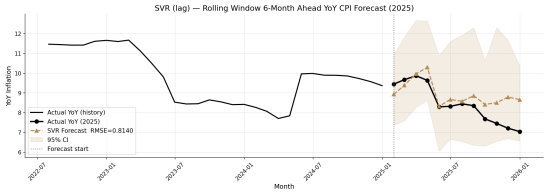


(f) **Elastic Net**

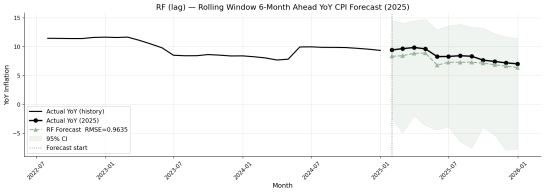


(g) **PLS**

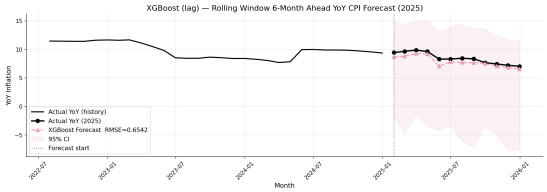
(h) **PCR**



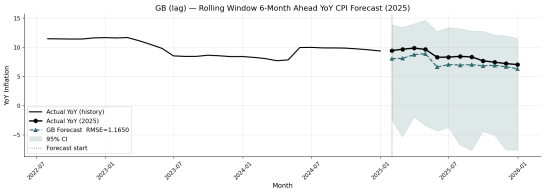
(i) **SVR**



(j) **Random Forest**



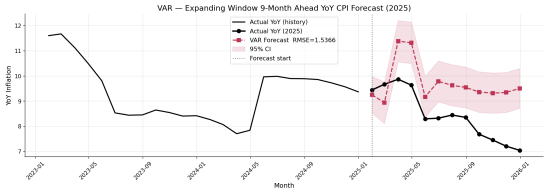
(k) **XGBoost**



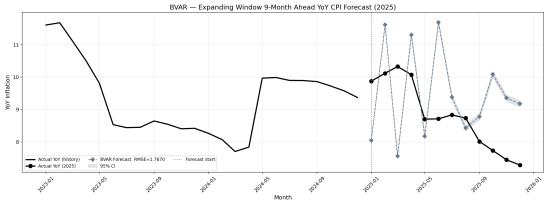
(l) **Gradient Boosting**

Figure 22: Out-of-sample YoY CPI inflation forecasts with 95% confidence intervals, $h = 6$ months ahead.

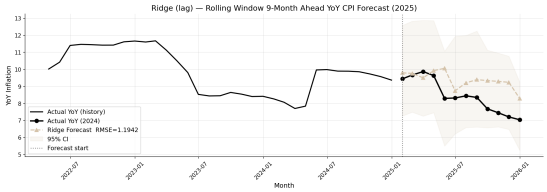
(a) **ARIMA**



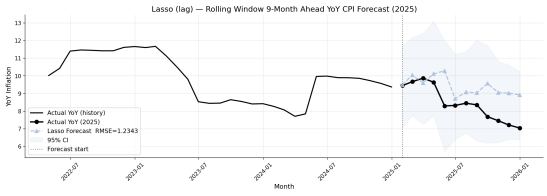
(b) **VAR**



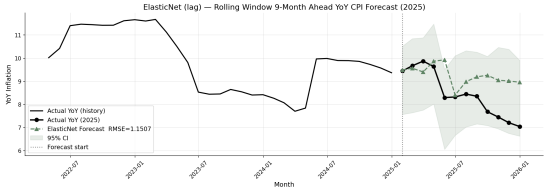
(c) **BVAR**



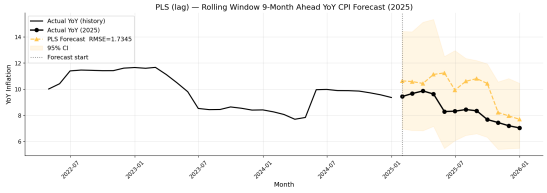
(d) **Ridge**



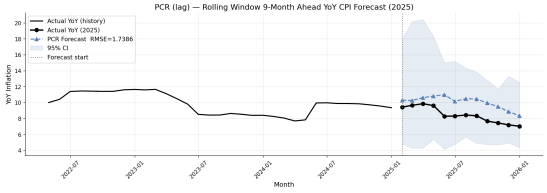
(e) **Lasso**



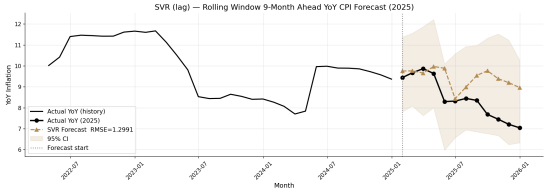
(f) **Elastic Net**



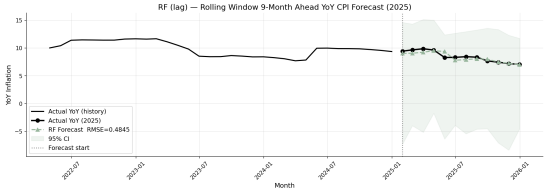
(g) **PLS**



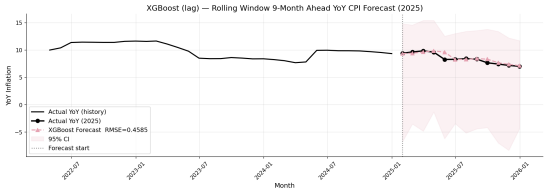
(h) **PCR**



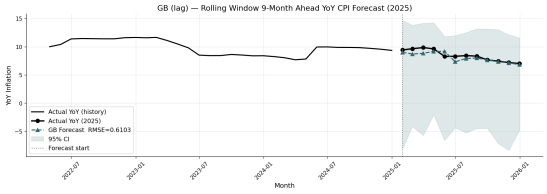
(i) **SVR**



(j) **Random Forest**

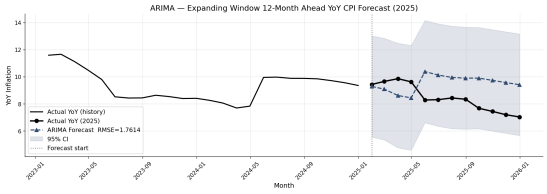


(k) **XGBoost**

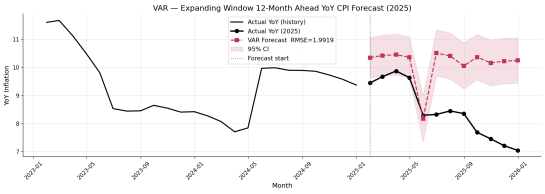


(l) **Gradient Boosting**

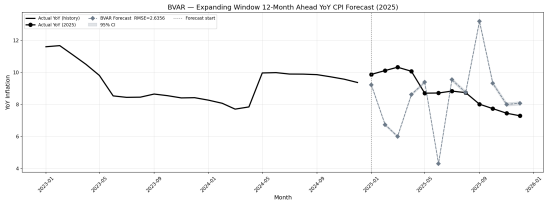
Figure 23: Out-of-sample YoY CPI inflation forecasts with 95% confidence intervals, $h = 9$ months ahead.



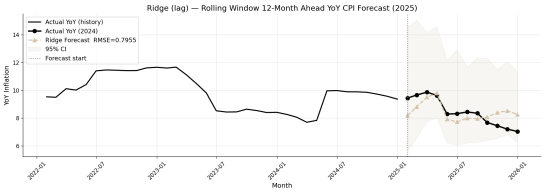
(a) **ARIMA**



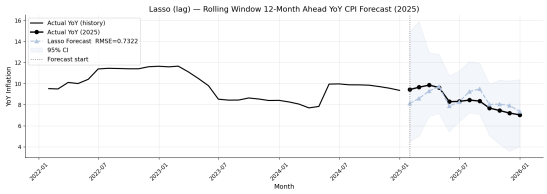
(b) **VAR**



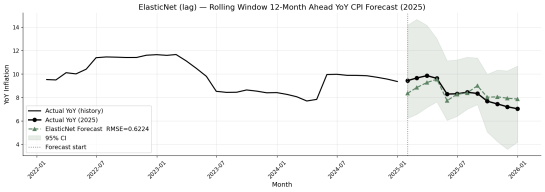
(c) **BVAR**



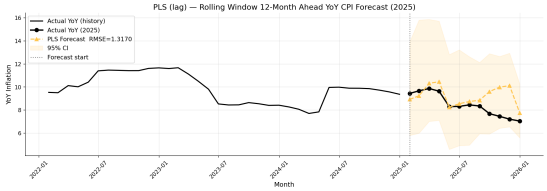
(d) **Ridge**



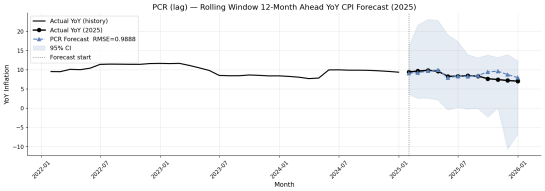
(e) **Lasso**



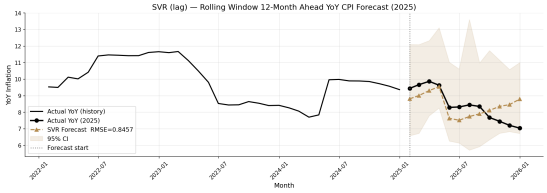
(f) **Elastic Net**



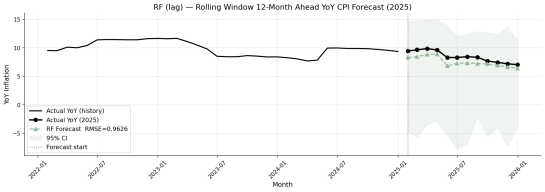
(g) **PLS**



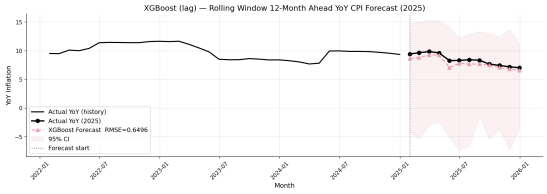
(h) **PCR**



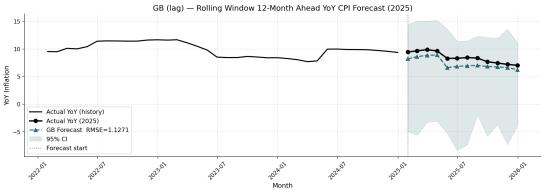
(i) **SVR**



(j) **Random Forest**



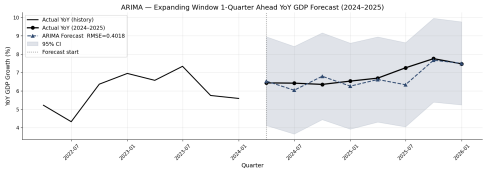
(k) **XGBoost**



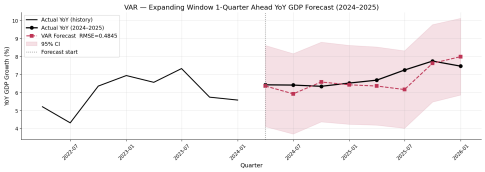
(l) **Gradient Boosting**

Figure 24: Out-of-sample YoY CPI inflation forecasts with 95% confidence intervals, $h = 12$ months ahead.

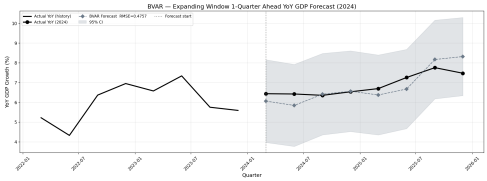
GDP forecast results



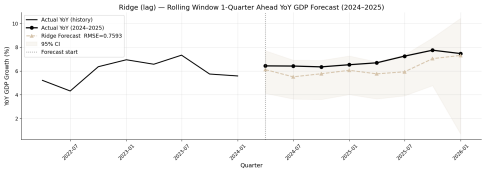
(a) **ARIMA**



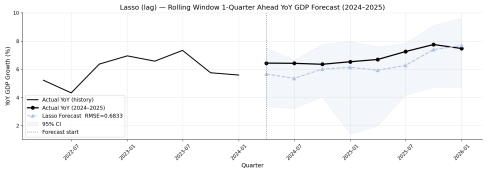
(b) **VAR**



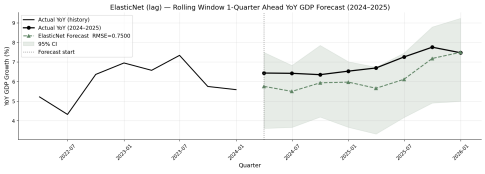
(c) **BVAR**



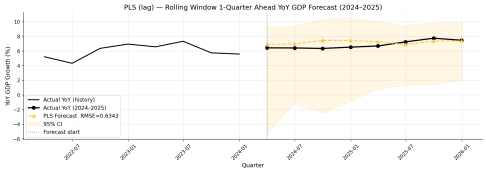
(d) **Ridge**



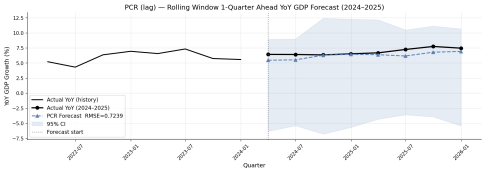
(e) **Lasso**



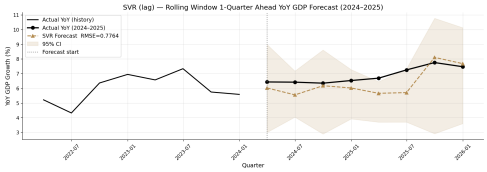
(f) **Elastic Net**



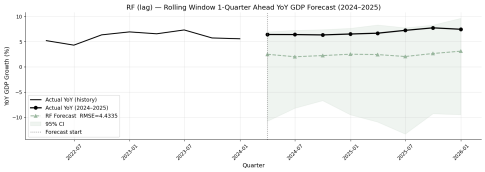
(g) **PLS**



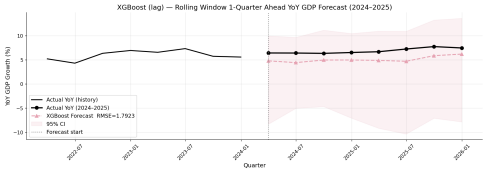
(h) **PCR**



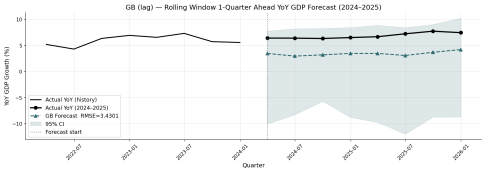
(i) **SVR**



(j) **Random Forest**



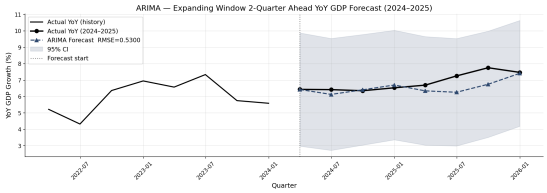
(k) **XGBoost**



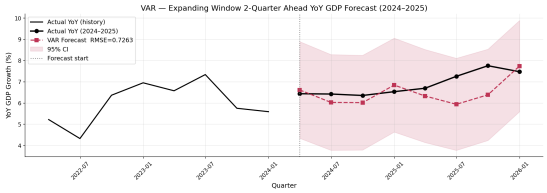
(l) **Gradient Boosting**

Figure 25: Out-of-sample GDP growth forecasts with 95% confidence intervals, $h = 1$ quarter ahead.

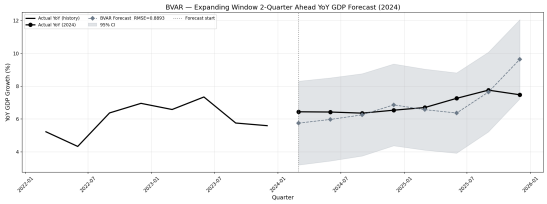
Note: Confidence intervals are constructed using a moving-block bootstrap. See Appendix D for details of the methodology.



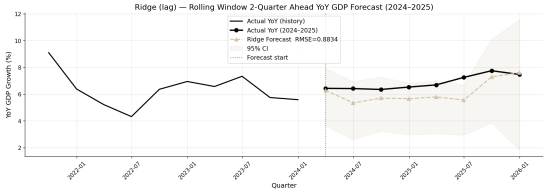
(a) **ARIMA**



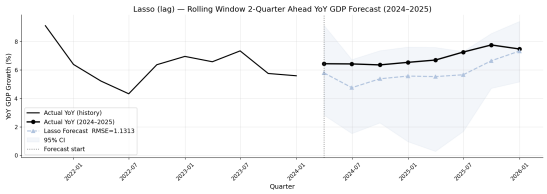
(b) **VAR**



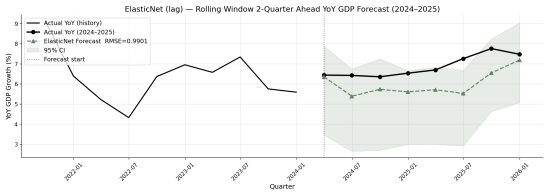
(c) **BVAR**



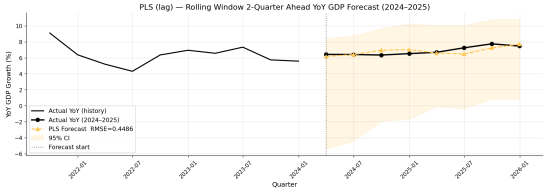
(d) **Ridge**



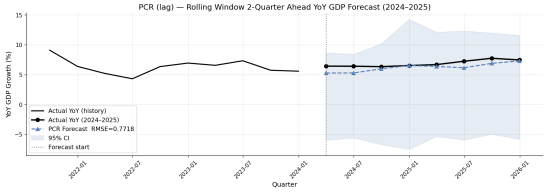
(e) **Lasso**



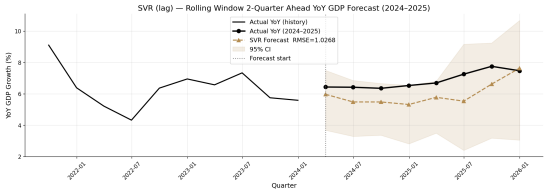
(f) **Elastic Net**



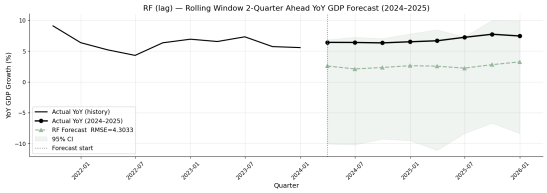
(g) **PLS**



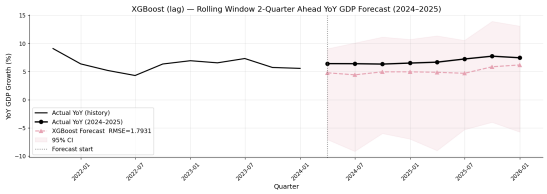
(h) **PCR**



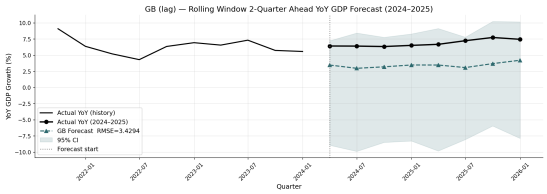
(i) **SVR**



(j) **Random Forest**

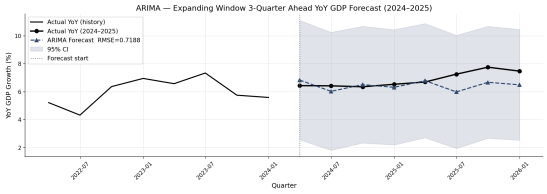


(k) **XGBoost**

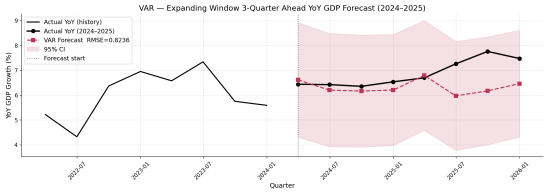


(l) **Gradient Boosting**

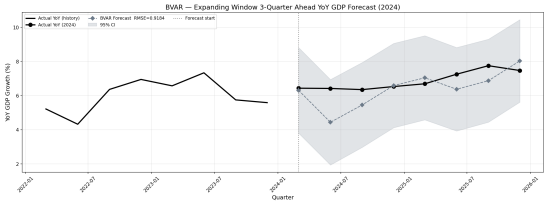
Figure 26: Out-of-sample GDP growth forecasts with 95% confidence intervals, $h = 2$ quarter ahead.



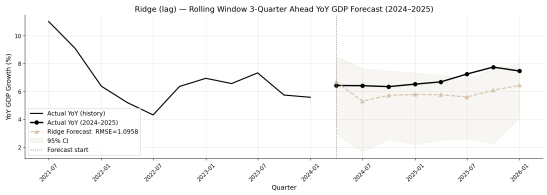
(a) **ARIMA**



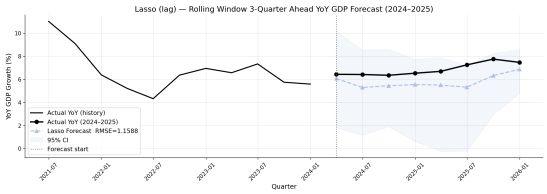
(b) **VAR**



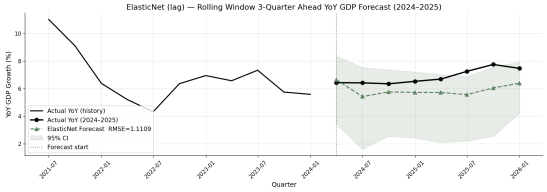
(c) **BVAR**



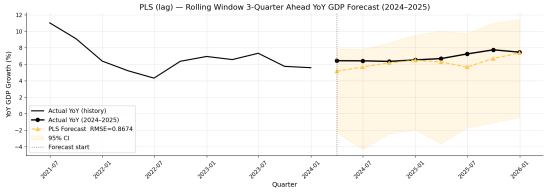
(d) **Ridge**



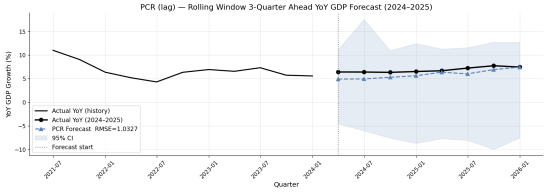
(e) **Lasso**



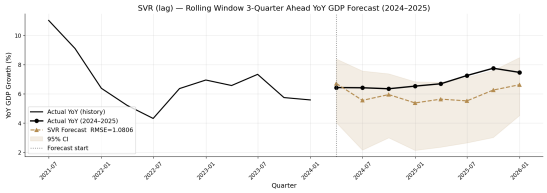
(f) **Elastic Net**



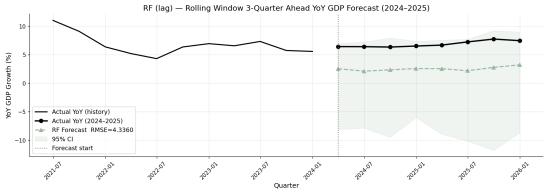
(g) **PLS**



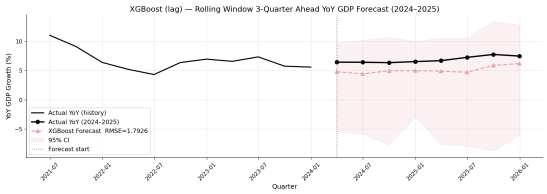
(h) **PCR**



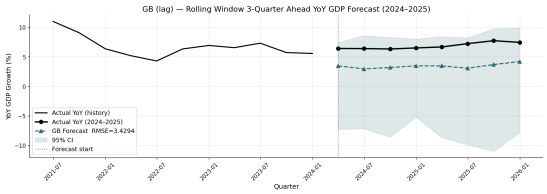
(i) **SVR**



(j) **Random Forest**

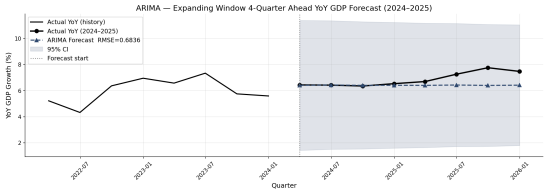


(k) **XGBoost**

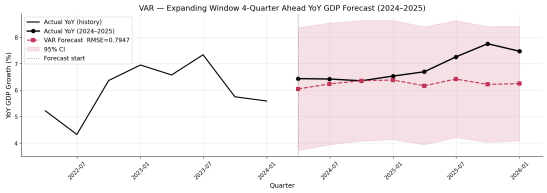


(l) **Gradient Boosting**

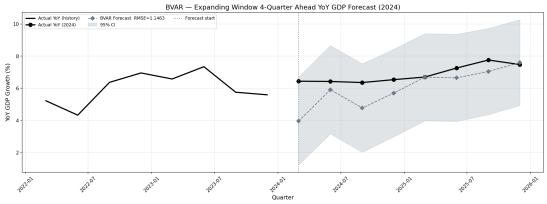
Figure 27: Out-of-sample GDP growth forecasts with 95% confidence intervals, $h = 3$ quarter ahead.



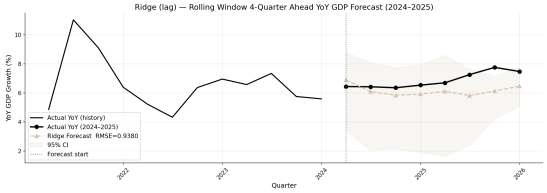
(a) **ARIMA**



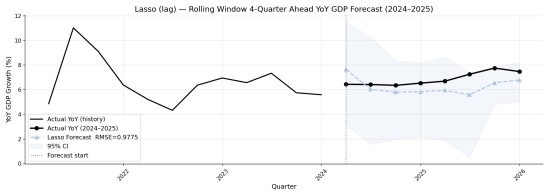
(b) **VAR**



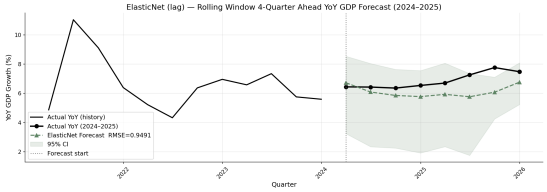
(c) **BVAR**



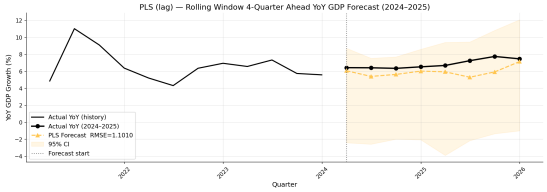
(d) **Ridge**



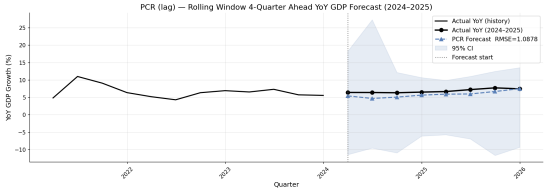
(e) **Lasso**



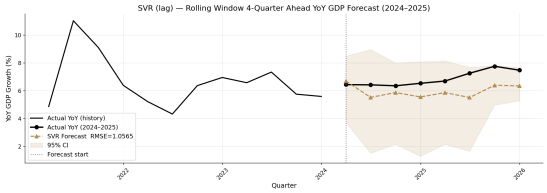
(f) **Elastic Net**



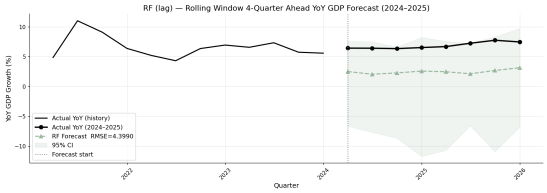
(g) **PLS**



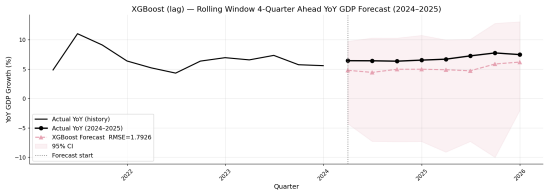
(h) **PCR**



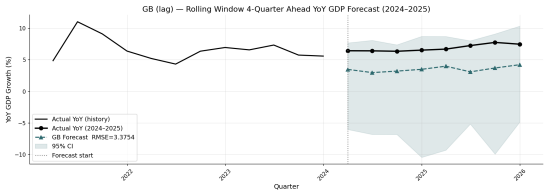
(i) **SVR**



(j) **Random Forest**



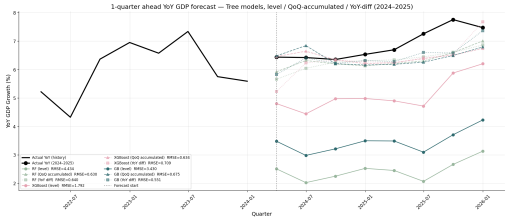
(k) **XGBoost**



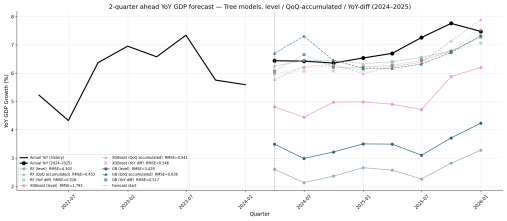
(l) **Gradient Boosting**

Figure 28: Out-of-sample GDP growth forecasts with 95% confidence intervals, $h = 4$ quarter ahead.

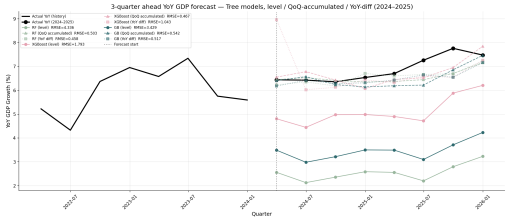
H Sensitivity of tree-based models to target transformation



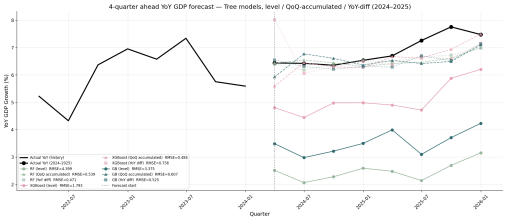
(a) $h = 1$ quarter



(b) $h = 2$ quarter

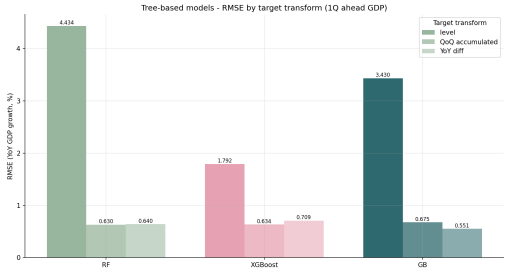


(c) $h = 3$ quarter



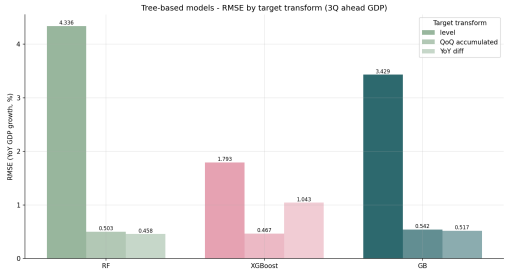
(d) $h = 4$ quarter

Figure 29: Out-of-sample YoY GDP growth forecasts for tree-based models under three target transformations (level, QoQ, and YoY), $h = 1$-4 quarters ahead.



(a) $h = 1$ quarter

(b) $h = 2$ quarter



(c) $h = 3$ quarter

(d) $h = 4$ quarter

Figure 30: Tree-based models: RMSE by target transformation and forecast horizon.

I Data description

The full dataset comprises 178 variables (111 monthly and 67 quarterly series) drawn from domestic, international, financial, and alternative data sources. Table 7 lists each variable together with its title, unit of measurement, frequency (M = Monthly, Q = Quarterly), available sample period, and source.

Table 7: Full List of Variables

Var.	Title	Units	Freq.	Available Period	Source
MACT1	Volume of industrial production	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT10	Amount of interbank transactions	Billions of UZS	M	2017M1-2025M12	National Statistics Office Uzbekistan
MACT11	Total revenue from sales and services	Billions of UZS	M	2018M1-2025M12	CBU
MACT12	Job search activity	index	M	2010M1-2025M12	Google trends
MACT13	Number of vacancies from olx.uz	number	M	2024M1-2025M12	CBU-OLX
MACT14	Economic activity index	index	M	2020M6-2025M12	CBU
MACT15	Business condition index	index	M	2020M6-2025M12	CBU
MACT2	Mining and quarrying	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT3	Manufacturing	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT4	Electricity, gas, steam, and air conditioning	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT5	Water supply, sewerage, waste management and remediation activities	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT6	Volume of construction and installation works	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT7	Services sector volume	Billions of UZS	M	2018M1-2025M12	National Statistics Office Uzbekistan
MACT8	Retail trade turnover	Billions of UZS	M	2018M9-2025M12	National Statistics Office Uzbekistan
MACT9	Total quantity of interbank transactions	Billions of UZS	M	2017M1-2025M12	National Statistics Office Uzbekistan
MBOP1	Exports	Millions of USD	M	2016M1-2025M12	Central Bank of Uzbekistan
MBOP10	Purchase of forex from households	Millions of USD	M	2017M9-2025M12	Central Bank of Uzbekistan
MBOP4	Imports	Millions of USD	M	2016M1-2025M12	Central Bank of Uzbekistan
MBOP8	Remittance	Millions of USD	M	2010M1-2025M12	Central Bank of Uzbekistan
MBOP9	Sales of forex to households	Millions of USD	M	2017M9-2025M12	Central Bank of Uzbekistan
MCOM1	Brent oil, futures price	USD per barrel	M	2010M1-2025M12	IMF
MCOM2	Uranium, futures price	USD per lbs	M	2010M1-2025M12	IMF
MCOM3	Cotton, futures price	USD per lbs	M	2010M1-2025M12	IMF
MCOM4	Copper, futures price	USD per lbs	M	2010M1-2025M12	IMF
MCOM5	Gold, futures price	USD per lbs	M	2010M1-2025M12	IMF
MCPI1	Consumer price Index	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI10	CPI Fruits and Vegetables	Index, 100 = December 2023	M	2008M12-2025M12	National Statistics Committee
MCPI11	CBU CPI	Index, 100 = December 2020	M	2020M12-2025M12	National Statistics Committee
MCPI12	CBU CPI Food	Index, 100 = December 2020	M	2020M12-2025M12	National Statistics Committee
MCPI13	CBU CPI Non-Food	Index, 100 = December 2020	M	2020M12-2025M12	National Statistics Committee
MCPI14	CBU CPI Services	Index, 100 = December 2020	M	2020M12-2025M12	National Statistics Committee

Continued on next page

Table 7 - continued from previous page

Var.	Title	Units	Freq.	Available Period	Source
MCPI15	CBU e-CPI	Index, 100 = August 2021	M	2021M8-2025M12	National Statistics Committee
MCPI16	CBU e-CPI Food	Index, 100 = August 2021	M	2021M8-2025M12	National Statistics Committee
MCPI17	CBU e-CPI Non-Food	Index, 100 = August 2021	M	2021M8-2025M12	National Statistics Committee
MCPI18	CBU e-CPI Services	Index, 100 = August 2021	M	2021M8-2025M12	National Statistics Committee
MCPI19	FAO Global Food Price Index	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI2	CPI Food	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI20	FAO Meat	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI21	FAO Dairy	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI22	FAO Cereals	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI23	FAO Oils	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI24	FAO Sugar	Index, 2014-2016=100	M	2006M1-2025M12	Food and Agriculture Organization
MCPI25	Household inflation expectations in Uzbekistan	Percent	M	2019M1-2025M12	CBU
MCPI26	Business inflation expectations in Uzbekistan	Percent	M	2019M1-2025M12	CBU
MCPI3	CPI Non-Food	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI4	CPI Services	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI5	CPI Core	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI6	CPI Core Food	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI7	CPI Core Non-Food	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI8	CPI Core Services	Index, 100 = December 2006	M	2006M1-2025M12	National Statistics Committee
MCPI9	CPI Regulated	Index, 100 = December 2022	M	2008M12-2025M12	National Statistics Committee
MDEF1	Headline inflation	MoM, %	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF2	Producer inflation	MoM, %	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF3	Service inflation	MoM, %	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF4	Import prices	MoM, %	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF5	Import prices index, 2010=100	index	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF6	Comes from weight of industry in GDP	weight	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF7	Comes from weight of service in GDP	weight	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF8	Proxy deflator based on ppi and service inflation	MoM, %	M	2010M1-2025M12	National Statistics Office Uzbekistan
MDEF9	Deflator index, 2010=100	index	M	2010M1-2025M12	National Statistics Office Uzbekistan
MEXR1	Exchange rate RUB/USD	RUB per USD	M	2010M1-2025M12	Trading Economics
MEXR2	Exchange rate CNY/USD	CNY per USD	M	2010M1-2025M12	Trading Economics
MEXR3	Exchange rate KAZ/USD	KZT per USD	M	2010M1-2025M12	Trading Economics
MEXR4	Exchange rate TRY/USD	TRL per USD	M	2010M1-2025M12	Trading Economics

Continued on next page

Table 7 - continued from previous page

Var.	Title	Units	Freq.	Available Period	Source
MEXR5	Exchange rate UZS/USD	UZS per USD	M	2010M1-2025M12	CBU
MEXT1	Consumer price index, Russia	index	M	2010M1-2025M12	CEIC
MEXT2	Consumer price index, China	index	M	2010M1-2025M12	CEIC
MEXT3	Consumer price index, Kazakhstan	index	M	2010M1-2025M12	CEIC
MEXT4	Consumer price index, Turkia	index	M	2010M1-2025M12	CEIC
MEXT5	Monetary policy rate, Russia	percent, p.a.	M	2010M1-2025M12	CEIC
MEXT6	Monetary policy rate, China	percent, p.a.	M	2013M11-2025M12	CEIC
MEXT7	Monetary policy rate, Kazakhstan	percent, p.a.	M	2010M1-2025M12	CEIC
MEXT8	Monetary policy rate, Turkia	percent, p.a.	M	2010M1-2025M12	CEIC
MFIS1	State budget revenues	Billions of UZS	M	2018M1-2025M12	Ministry of Economy and Finance
MFIS2	State budget expenditures	Billions of UZS	M	2018M1-2025M12	Ministry of Economy and Finance
MINR1	Central Bank Policy Rate (End of Period)	Percent, p.a.	M	2013M1-2025M12	Central Bank of Uzbekistan
MINR2	Money Market Rate	Percent, p.a.	M	2013M1-2025M12	Central Bank of Uzbekistan
MINR3	Treasury Bill Rate	Percentage points	M	2018M12-2025M10	Central Bank of Uzbekistan
MINR4	Deposit Rate	Percent, p.a.	M	2013M1-2025M12	Central Bank of Uzbekistan
MINR5	Prime Lending Rate	Percentage points	M	2013M1-2025M12	Central Bank of Uzbekistan
MMON1	Monetary base	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON10	Loans to economy, stock	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON11	Loans to households, stock	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON12	Loans to agricultural sector, stock	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON13	Loans to State owned enteprises, stock	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON14	Loans to private entities	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON15	Loans to other legal entities	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON2	Money supply, M2	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON3	Currency in circulation, M0	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON4	Demand deposits in domestic currency	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON5	Other deposits in national currency	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON6	Deposits in foreign currency	Billions of UZS	M	2012M1-2025M12	Central Bank of Uzbekistan
MMON7	New loans total	Billions of UZS	M	2013M1-2025M12	Central Bank of Uzbekistan
MMON8	New loans in national currency	Billions of UZS	M	2017M1-2025M12	Central Bank of Uzbekistan
MMON9	New loans in foreign currency	Billions of UZS	M	2017M1-2025M12	Central Bank of Uzbekistan
MPPI1	Producer Price Index	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI10	PPI Coke	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI11	PPI Non-Metal Minerals	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI12	PPI Metallurgy	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI13	PPI Fabricated Metals	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI14	PPI Electrical Equipment	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI15	PPI Vehicles	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI16	PPI Metal Ores	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee

Continued on next page

Table 7 - continued from previous page

Var.	Title	Units	Freq.	Available Period	Source
MPPI2	PPI Mining	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI3	PPI Manufacturing	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI4	PPI Electricity and Gas	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI5	PPI Water	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI6	PPI Food	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI7	PPI Beverages	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI8	PPI Textiles	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
MPPI9	PPI Chemicals	Index, 100 = December 2015	M	2016M1-2025M12	National Statistics Committee
QACT1	Job search activity	Index, 3 month average	Q	2010Q1-2025Q4	Google trend
QACT2	Industrial Production Index, Quarterly, ending "Mar, June, Sep, Dec"	Index, 2010=100	Q	2010Q1-2025Q4	International Monetary Fund, CEIC
QACT3	Market Capitalization: RSE, 3 month average	UZS bn	Q	2010Q1-2025Q4	Republican Stock Exchange "Toshkent", CEIC
QACT4	NN: EIA: Crude Oil Production: Non-OPEC Producers: Uzbekistan, 3 month average	Barrel/Day th	Q	2010Q1-2025Q4	U.S. Energy Information Administration, CEIC
QACT5	National Minimum Wage, 3 month average	UZS	Q	2010Q1-2025Q4	President of the Republic of Uzbekistan, CEIC
QACT6	Residential Buildings Commissioning (NET)	sq m th	Q	2010Q1-2025Q4	State Committee of the Republic of Uzbekistan on Statistics? CEIC
QBOP1	Current account balance	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP2	Export of goods and services	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP3	Import of goods and services	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP4	Net foreign assets	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP5	External debt	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP6	Forex reservs	Millions of USD	Q	2005Q1-2025Q4	CBU
QBOP7	Direct investment: liabilities	Million USD	Q	2005Q1-2025Q4	CBU
QBOP8	Remittance	Millions of USD, sum of 3 months	Q	2010Q1-2025Q4	n/a
QCOM1	Brent oil, futures price	USD per barrel, 3 month average	Q	2010Q1-2025Q4	IMF
QCOM2	Uranium, futures price	USD per lbs, 3 month average	Q	2010Q1-2025Q4	IMF
QCOM3	Cotton, futures price	USD per lbs, 3 month average	Q	2010Q1-2025Q4	IMF
QCOM4	Copper, futures price	USD per lbs, 3 month average	Q	2010Q1-2025Q4	IMF
QCOM5	Gold, futures price	USD per lbs, 3 month average	Q	2010Q1-2025Q4	IMF
QEXR1	Exchange rate RUB/USD	RUB per USD, 3 month average	Q	2010Q1-2025Q4	Trading Economics
QEXR2	Exchange rate CNY/USD	CNY per USD, 3 month average	Q	2010Q1-2025Q4	Trading Economics
QEXR3	Exchange rate KAZ/USD	KZT per USD, 3 month average	Q	2010Q1-2025Q4	Trading Economics
QEXR4	Exchange rate TRY/USD	TRL per USD, 3 month average	Q	2010Q1-2025Q4	Trading Economics
QEXR5	Exchange rate UZS/USD	UZS per USD, 3 month average	Q	2010Q1-2025Q4	CBU
QEXT1	Consumer price index, Russia	index, 3 month average	Q	2010Q1-2025Q4	CEIC

Continued on next page

Table 7 - continued from previous page

Var.	Title	Units	Freq.	Available Period	Source
QEXT2	Consumer price index, China	index, 3 month average	Q	2010Q1-2025Q4	CEIC
QEXT3	Consumer price index, Kazakhstan	index, 3 month average	Q	2010Q1-2025Q4	CEIC
QEXT4	Consumer price index, Turkia	index, 3 month average	Q	2010Q1-2025Q4	CEIC
QEXT5	Monetary policy rate, Russia	percent, p.a., 3 month average	Q	2010Q1-2025Q4	CEIC
QEXT6	Monetary policy rate, Kazakhstan	percent, p.a., 3 month average	Q	2010Q1-2025Q4	CEIC
QEXT7	Monetary policy rate, Turkia	percent, p.a., 3 month average	Q	2010Q1-2025Q4	CEIC
QFIS1	State budget revenues	Billions of UZS	Q	2012Q1-2025Q4	Ministry of Economy and Finance
QFIS2	State budget expenditures	Billions of UZS	Q	2012Q1-2025Q4	Ministry of Economy and Finance
QINR1	Central Bank Policy Rate (End of Period)	Percent, p.a., 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QINR2	Money Market Rate	Percent, p.a., 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QINR3	Deposit Rate	Percent, p.a., 3 month average	Q	2013Q1-2025Q4	Central Bank of Uzbekistan
QMON1	Monetary base	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON10	Loans to State owned enteprises, stock	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON11	Loans to private entities	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON12	Loans to other legal entities	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON13	Total Outstanding loan, stock	Billions of UZS, 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QMON14	Total Outstanding loan in national currency, stock	Billions of UZS, 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QMON15	Total Outstanding loan in foreign currency, stock	Billions of UZS, 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QMON2	Money supply, M2	Billions of UZS, 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QMON3	Currency in circulation, M0	Billions of UZS, 3 month average	Q	2010Q1-2025Q4	Central Bank of Uzbekistan
QMON4	Demand deposits in domestic currency	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON5	Other deposits in national currency	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON6	Deposits in foreign currency	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON7	Loans to economy, stock	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON8	Loans to households, stock	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QMON9	Loans to agricultural sector, stock	Billions of UZS, 3 month average	Q	2012Q1-2025Q4	Central Bank of Uzbekistan
QNAT1	Real GDP, NET, calculated using deflator, for more look at: Main-rGDP and Deflator.xlsx	Billions of UZS	Q	2010Q1-2025Q4	National Statistics Office Uzbekistan
QNAT2	Real GDP, Cumulative, calculated using deflator, for more look at: Main-rGDP and Deflator.xlsx	Billions of UZS	Q	2010Q1-2025Q4	National Statistics Office Uzbekistan
QNAT3	GDP growth (from STAT.uz)	Percent	Q	2010Q1-2025Q4	National Statistics Office Uzbekistan

Continued on next page

Table 7 - continued from previous page

Var.	Title	Units	Freq.	Available Period	Source
QPI1	Consumer price Index	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI10	FAO Dairy	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization
QPI11	FAO Cereals	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization
QPI12	FAO Oils	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization
QPI13	FAO Sugar	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization
QPI2	CPI Food	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI3	CPI Non-Food	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI4	CPI Services	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI5	CPI Core	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI6	Producer Price Index	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Committee
QPI7	Import prices index, 2010=100	Index, 2010=100	Q	2010Q1-2025Q4	National Statistics Office Uzbekistan
QPI8	FAO Global Food Price Index	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization
QPI9	FAO Meat	Index, 2014-2016=100, 3 month average	Q	2006Q1-2025Q4	Food and Agriculture Organization