BANCO CENTRAL DE RESERVA DEL PERÚ

Nowcasting Peruvian GDP with Machine Learning Methods

Jairo Flores*, Bruno Gonzaga*, Walter Ruelas-Huanca* and Juan Tang*

* Banco Central de Reserva del Perú.

DT. N°. 2024-019
Serie de Documentos de Trabajo
Working Paper series
Diciembre 2024

Nowcasting Peruvian GDP with Machine Learning Methods*

Jairo Flores
jairo.flores@bcrp.gob.pe

Bruno Gonzaga
bruno.gonzaga@bcrp.gob.pe

Walter Ruelas-Huanca
walter.ruelas@bcrp.gob.pe

Juan Tang
juan.tang@bcrp.gob.pe

Abstract

This paper explores the application of machine learning (ML) techniques to nowcast the monthly year-over-year growth rate of both total and non-primary GDP in Peru. Using a comprehensive dataset that includes over 170 domestic and international predictors, we assess the predictive performance of 12 ML models, including Lasso, Ridge, Elastic Net, Support Vector Regression, Random Forest, XGBoost, and Neural Networks. The study compares these ML approaches against the traditional Dynamic Factor Model (DFM), which serves as the benchmark for nowcasting in economic research. We treat specific configurations, such as the feature matrix rotations and the dimensionality reduction technique, as hyperparameters that are optimized iteratively by the Tree-Structured Parzen Estimator. Our results show that ML models outperformed DFM in nowcasting total GDP, and that they achieve similar performance to this benchmark in nowcasting non-primary GDP. Furthermore, the bottom-up approach appears to be the most effective practice for nowcasting economic activity, as aggregating sectoral predictions improves the precision of ML methods. The findings indicate that ML models offer a viable and competitive alternative to traditional nowcasting methods.
JEL classification: C14, C32, E32, E52.
Keywords: GDP, Machine Learning, nowcasting.

*We are grateful to Cedric Tille, Dalibor Stevanovic, Andrew Garcia, as well as to the participants at the XLII Meeting of Economist of the Central Reserve Bank of Peru (BCRP) 2024, Economic and Social Research Consortium (CIES) 2024 and the Bilateral Assistance and Capacity Building for Central Banks participants for useful comments and suggestions. This research took place through the coaching program under the Bilateral Assistance and Capacity Building for Central Banks (BCC), financed by SECO, and the Graduate Institute in Geneva. Any opinions expressed in this research are those of the authors and do not necessarily represent the official views of the BCRP.

1 Introduction

Nowcasting, which involves estimating the current state of an economy in near real-time, plays a critical role in economic analysis, particularly for decision making and policy formulation. The practice has been predominantly applied in meteorology and more recently in economics, as emphasized by the seminal works of Giannone, Reichlin and Small (2008), and Bańbura et al. (2013). In recent years, machine learning (ML) techniques have shown great potential in macroeconomic nowcasting, particularly due to their ability to capture nonlinear relationships that traditional models might miss. Although several studies have evaluated ML methods for nowcasting GDP in advanced economies, limited but growing research has been conducted on developing economies such as Peru.

We seek to build upon the growing body of research that examines how ML techniques can improve the accuracy of GDP nowcasts. The challenge of nowcasting consists of i) identifying a robust set of high-frequency indicators that can capture real-time signals of economic activity, and ii) selecting an optimal predictive model to process and transform these indicators into reliable estimates. We tackle these challenges by leveraging a comprehensive dataset, comprising over 170 domestic and international predictors that cover the period from April 2015 to August 2024, combining structured macroeconomic variables with unstructured data sources, including Google Trends, to offer a more holistic approach to nowcasting.

The primary goal of this research is to compare the performance of different ML algorithms in nowcasting the monthly year-over-year growth rate of total and non-primary Peru's GDP. The models tested include regularization methods such as Ridge and Lasso, tree-based models like Random Forest and XGBoost, and advanced techniques like Support Vector Regression and Neural Networks. A further goal is to assess the performance of these models in comparison to a Dynamic Factor Model (DFM), which serves as the benchmark for time-series nowcasting in economic research and practice. We tested various model specifications, focusing on several key aspects of the nowcasting process. Specifically, we explored the use of expanding versus rolling windows for the training sample, compared the performance of K-Fold cross-validation against Walk-Forward cross-validation, and evaluated different strategies for dimensionality reduction and feature matrix rotations to optimize model performance. These comparisons allowed us to identify the most effective techniques for enhancing the accuracy of our ML models in nowcasting GDP.

A recent study closely related to our research is the work by Tenorio and Perez (2023), which explores the application of ML techniques for nowcasting Peruvian GDP. While their study offers valuable insights, our research distinguishes itself in several important ways. First, we employ a broader and more diverse set of predictors that allow for a more comprehensive analysis of the economic indicators influencing GDP growth. Second, our study extends the analysis to examine the performance of ML methods in nowcasting non-primary GDP, a critical component of the economy that often lacks timely real-time information. Third, we perform a bottom-up analysis by predicting sectoral GDP growth and then aggregating these predictions to obtain the overall GDP growth estimate. This approach allows for a detailed comparison of ML precision at both sectoral and aggregate levels. Finally, we conduct a recursive approach for hyperparameter optimization using the Tree-Structured Parzen Estimator, and explore the usefulness of dimensionality reduction techniques and feature rotations for enhancing models' performance.

The remainder of this paper is structured as follows: Section 2 reviews the relevant literature on GDP nowcasting and ML applications. Section 3 describes the dataset and the estimation strategy, detailing the preprocessing steps, sample splits, and hyperparameter optimization procedures. Section 4 presents the results, comparing the performance of the models using the expanding and rolling window techniques. Finally, Section 5 concludes the paper by discussing the implications of the findings for future research.

2 Literature Review

This paper contributes to the expanding literature that evaluates the relative effectiveness of ML models for nowcasting GDP, as compared to traditional time-series methods. Numerous studies have implemented a wide range of ML techniques to improve GDP nowcasting accuracy, with many finding that these approaches either surpass or rival standard statistical methods.

Recent research has increasingly focused on assessing the performance of ML models in nowcasting GDP, particularly in advanced economies. For instance, Soybilgen and Yazgan (2021) and Hopp (2024) applied ML techniques to nowcast U.S. GDP. Soybilgen and Yazgan (2021) employed bagged decision trees, random forests, and stochastic gradient tree boosting models, while Hopp (2024) expanded on this by incorporating long short-term memory (LSTM) networks and XGBoost in addition to the aforementioned models. Similarly, Richardson, van Florenstein Mulder and Vehbi (2021) applied a range of ML techniques, including ridge, Lasso, elastic net, and support vector machine (SVM) regression, to nowcast New Zealand's GDP, emphasizing methods beyond gradient boosting and neural networks. Kant, Pick and de Winter (2022) conducted a comparable exercise for the Dutch economy, demonstrating the versatility of ML models in various economic contexts.

For developing economies, several studies have also explored the potential of ML techniques in nowcasting GDP. Zhang, Ni and Xu (2023) compared the performance of various ML algorithms against DFMs, static factor models, and MIDAS for short-term forecasting of China's annualized real GDP. Ghosh and Ranjan (2023) evaluated these approaches for a group of emerging economies, while Dauphin et al. (2022) applied them to European countries, Muchisha et al. (2021) focused on Indonesia, Fornaro and Luomaranta (2020) on Finland, and Tiffin (2016) on Lebanon. Additionally, Buell et al. (2021) examined the application of ML models to nowcasting GDP in Sub-Saharan Africa.

In Latin America, several studies have implemented ML models for nowcasting GDP. For instance, Bolivar (2024) applied these methods to Bolivia, while León and Ortega (2018) utilized artificial neural networks (ANNs) to nowcast monthly economic activity in Colombia using electronic payments data. Bravo Higuera et al. (2024) explored regularization techniques to produce early estimates of agricultural sector GDP in Colombia. De Oliveira (2023) compared traditional statistical methods and ML techniques to nowcast Brazilian GDP, finding that the best results were obtained when ML predictions were combined with traditional models. Miranda (2021) applied Lasso and deep neural networks to nowcast economic activity in Mexico.

In the case of Peru, the application of ML techniques to nowcasting GDP is more limited. However, one of the few recent studies in this area is by Tenorio and Perez (2023), who applied ML methods to nowcast monthly Peruvian GDP using a large dataset that includes both structured and unstructured data sources.

This growing body of literature illustrates the increasing relevance and success of ML models in improving the accuracy and timeliness of GDP nowcasting across both advanced and developing economies.

3 Methodology

3.1 Models

In this section, we provide a brief description of the various ML and benchmark models we considered for nowcasting Peruvian GDP. Please refer to the original citations for greater detail. Our primary focus is on the methodologies to choose key hyperparameters, and on the details of the out-of-sample exercise to assess nowcast performance.

Dynamic Factor Models: As a benchmark, we employ a standard DFM following the framework of Bańbura et al. (2013) and Mariano and Murasawa (2010), and a modified version of the implementation in Python made by Fulton (2020). In a DFM, it is assumed that a reduced number of unobserved latent factors can explain a substantial portion of the variability in a large set of observable macroeconomic and financial variables. This model is particularly useful for nowcasting because it allows the extraction of common information from a high-dimensional dataset, which can then be used to predict key economic indicators such as GDP.

The DFM can be expressed as follows:

$$y_t = Af_t + e_t$$

$$f_t = A_1 f_{t-1} + A_2 f_{t-2} + ... + U_t$$

where y_t is a $N \times 1$ vector of observable variables at time t, f_t is a $r \times 1$ vector of unobserved common factors. A is the $N \times r$ matrix of factor loadings. $A_1, A_2, ...$ are $r \times r$ autoregressive coefficient matrices. e_t is the $N \times 1$ vector of idiosyncratic disturbances and U_t is the $r \times 1$ vector of factor innovations. For more details on the estimation procedure, please refer to the Appendix.

Ridge, Lasso, and Elastic Net: These models are widely used regularization techniques aimed at reducing model complexity, particularly when dealing with datasets that contain a large number of features. These methods help mitigate overfitting by introducing a penalty term in the regression model, which effectively constrains the size of the estimated coefficients.

Ridge regression, also known as $L2$ regularization, works by adding a penalty to the sum of the squared magnitudes of the coefficients, effectively shrinking them towards zero. This shrinkage helps reduce the influence of less important features while maintaining all features in the model. The mathematical formulation for ridge regression is as follows:

$$\beta = \arg\min \left[\sum_{i=1}^{T} (y_i - \beta_0 - \sum_{j=1}^{k} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{k} \beta_j^2 \right]$$

Here, λ is a regularization parameter that controls the amount of shrinkage applied. As λ increases, the model imposes stronger penalties on the coefficients, shrinking those with minimal contribution to the response variable closer to zero. However, ridge regression does not set any coefficients exactly to zero, so all variables remain in the model.

Lasso regression, or $L1$ regularization, takes a different approach by applying a penalty on the absolute values of the coefficients rather than their squared values. This encourages sparsity in the model, allowing Lasso to shrink some coefficients entirely to zero, thus performing feature selection. Lasso is particularly useful when only a subset of the features are expected to be truly relevant. The formulation of Lasso regression is as follows:

$$\beta = \arg\min \left[\sum_{i=1}^{T} (y_i - \beta_0 - \sum_{j=1}^{k} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{k} |\beta_j| \right]$$

Like ridge regression, λ is the hyperparameter controlling the degree of regularization. A higher value of λ increases the likelihood that coefficients will be reduced to zero, effectively excluding some features from the model.

Elastic Net combines the strengths of both ridge and Lasso by incorporating both $L2$ (Ridge) and $L1$ (Lasso) penalties. This allows elastic net to both shrink coefficients and perform feature selection by driving some coefficients to zero. The method introduces a mixing parameter, α, which controls the balance between the $L1$ and $L2$ penalties. The elastic net objective function is given by

$$\beta = \arg\min \left[\sum_{i=1}^{T} (y_i - \beta_0 - \sum_{j=1}^{k} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{k} ((1-\alpha)\beta_j^2 + \alpha|\beta_j|) \right]$$

In this formulation, λ controls the overall strength of regularization, while α (ranging between 0 and 1) determines the trade-off between Ridge ($L2$) and Lasso ($L1$) regularization. When $\alpha = 0$, Elastic Net behaves like Ridge regression, and when $\alpha = 1$, it behaves like Lasso.

Support Vector Machine: The Support Vector Machine (SVM) algorithm for regression, also known as Support Vector Regression (SVR), was introduced by Cortes and Vapnik (1995) and seeks to find a function $f(x)$ that predicts the target value y_i with a maximum deviation of ϵ from the actual targets, while keeping the function as flat as possible. In the simplest case, when the function is linear, the model can be expressed as:

$$f(x) = (w, x) + b$$

where w is the weight vector, x is the input feature vector, and b is the bias term. The concept of "flatness" here refers to minimizing the magnitude of the weight vector w, which corresponds to finding a decision function that has the smallest possible slope. This helps ensure that the model generalizes well to unseen data, avoiding overfitting.

However, in many cases, it may not be possible to find a function that perfectly satisfies the condition of keeping all deviations within ϵ. To handle this, the algorithm introduces slack variables ξ_i and ξ_i^*, which allow for some errors, permitting deviations from the ϵ-tube. These slack variables measure the extent to which the predictions $f(x)$ fall outside the allowable error margin. The optimization problem for SVM regression is then defined as:

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{T}(\xi_i + \xi_i^*)$$

subject to:

$$\begin{aligned} y_i - (w, x) - b \leq & \quad \epsilon + \xi_i \\ (w, x) + b - y_i \leq & \quad \epsilon + \xi_i^* \end{aligned}$$

The constant C, known as the regularization parameter, plays a crucial role in controlling the model's complexity. A larger value of C allows for fewer slack variables, enforcing stricter adherence to the ϵ-tube and potentially leading to overfitting if the model becomes too complex. Conversely, a smaller C value allows for more slack, which can lead to underfitting but encourages better generalization by permitting more errors.

K-Nearest Neighbors: K-Nearest Neighbors (KNN), introduced by Fix and Hodges (1951) and extended by Cover and Hart (1967), is a simple, yet powerful non-parametric ML algorithm widely used for both classification and regression tasks. Unlike parametric models, KNN does not assume any specific form for the relationship between the predictors and the dependent variable. Instead, it relies on the proximity of data points in the feature space, making predictions based on the values of the k-nearest neighbors.

In regression tasks, KNN predicts the outcome for a given query point X_q by averaging the outcomes of the k-nearest observations from the training data. The key concept in KNN is the calculation of the distance between the query point X_q and each observation X_i in the training dataset, typically using the Euclidean distance:

$$d(X_q, X_i) = \sqrt{\sum_{j=1}^{d} (X_{q,j} - X_{i,j})^2}$$

Where d is the number of dimensions (or features) in the dataset, $X_{q,j}$ and $X_{i,j}$ are the values of the j-th feature for the query point and the i-th observation, respectively. Once the distances are computed, the k-nearest neighbors are identified based on their proximity to the query point. The predicted value $\hat{y}_q$ is then calculated as the average of the dependent variable y_i values of these nearest neighbors:

$$\hat{y}_q = \frac{1}{K} \sum_{i \in N_k(x_q)} y_i$$

Where $N_k(X_q)$ represents the set of indexes corresponding to the k-nearest neighbors of X_q and y_i is the observed value of the dependent variable for the i-th neighbor.

Decision Tree: A decision tree (DT), introduced by Breiman et al. (1984), is a non-parametric model that recursively partitions the data space into subsets based on the values of the predictors. Each split is chosen to best separate the observations according to some criterion, such as minimizing the variance in regression tasks. The result is a tree structure where internal nodes represent the decision rules based on the predictors, and the terminal nodes (also known as leaves) represent regions of the data where a simple prediction model, typically a constant value, is applied.

In regression decision trees, the model divides the predictor space into a number of disjoint regions R_m and the predicted value for any new observation falling into region R_m is given by a constant c_m, which is the mean of the y-values in that region. This can be represented mathematically as:

$$E(Y|X_1, X_2) = \sum_{m=1}^{p} c_m 1_{(X_1, X_2) \in R_m}$$

where p is the total number of regions.

Gradient Boosting, Extreme Gradient Boosting and Adaptive Boosting: Gradient boosting

(GBoostig) is an ensemble learning technique that builds a robust predictor by combining multiple weak learners, usually decision trees. Each weak learner is trained sequentially, with the goal of improving the predictive accuracy of the model by focusing on the errors made by the previous learners. This process iteratively minimizes a predefined loss function, such as least squares or least absolute deviation, to refine the model's predictions.

The algorithm starts by fitting an initial model to the target variable and calculating the residuals. A new model is then added that fits these residuals, with each subsequent model improving upon the errors of the previous one. The gradient boosting algorithm updates the overall model as follows:

$$F_m(x) = F_{m-1}(x) + \upsilon\Delta_m(x)$$

Here, $F_m(x)$ represents the updated model at iteration m, $F_{m-1}(x)$ is the previous model, $\Delta_m(x)$ is the weak learner (often a decision tree), and υ is the learning rate, a key hyperparameter that controls the contribution of each new learner. The learning rate υ acts as a shrinkage parameter, helping to prevent overfitting by limiting the influence of each model update. A value less than 1 for υ effectively slows down the learning process, improving generalization beyond the training data.

Extreme Gradient Boosting (XGBoost) is an advanced implementation of GBoosting that introduces several optimizations to improve performance and scalability (Chen and Guestrin (2016)). It includes additional features like regularization terms (L1 and L2), which help control model complexity and prevent overfitting. XGBoost also supports efficient handling of missing data, parallelized tree construction, and early stopping, where the model halts further iterations when no significant improvement is detected. These features make XGBoost highly efficient and scalable for large datasets, which has contributed to its widespread adoption in ML competitions and practical applications.

Adaptive Boosting (AdaBoost) is another popular boosting algorithm, but it differs from GBoosting in its approach. Instead of fitting to residuals, AdaBoost adjusts the weights of the training instances at each iteration, giving more focus to observations that were poorly predicted by the previous model (Freund and Schapire (1997)). By increasing the influence of these "difficult" cases, AdaBoost forces subsequent models to correct the mistakes made earlier in the process. One limitation of AdaBoost is its sensitivity to noisy data and outliers, which can be addressed by tuning the hyperparameters carefully.

Bagging: Short for bootstrap aggregating, is an ensemble learning technique that enhances model robustness and accuracy by reducing variance and mitigating overfitting. It involves training multiple models independently on a unique bootstrap sample. By exposing each model to a distinct variation of the training data, bagging reduces the dependency on any single dataset and diminishes the variance associated with individual models.

Once trained, the predictions from all models are aggregated to produce a final output. This combination leverages the strengths of the individual models while counteracting their individual errors. Bagging is particularly effective when applied to high-variance models, such as decision trees, where it curtails overfitting by ensuring that each constituent model learns from different portions of the dataset. Consequently, the ensemble model delivers improved predictive performance, especially in scenarios where variability in individual models could otherwise compromise accuracy.

Random Forest: Random Forest (RF), introduced by Breiman (2001), is an ensemble learning method that combines multiple decision trees to produce a more robust and accurate model. It is based on the concept of Bagging, but RF introduces another layer of randomness by selecting a random subset of predictors (features) at each split within the decision trees. This random selection prevents individual trees from relying too heavily on any one feature and encourages diversity across the trees.

The final prediction of the RF model is obtained by averaging the predictions of all individual trees in regression tasks. This averaging reduces the variance of the model and improves its generalization ability, making it less prone to overfitting compared to a single decision tree.

Mathematically, the Random Forest prediction for a regression task is given by:

$$\hat{y} = \frac{1}{M} \sum_{m=1}^{M} \hat{y}_m(x)$$

where $\hat{y}_m(x)$ is the prediction from the m-th decision tree and M is the total numbers of trees in the forest.

Artificial Neural Network: A Multilayer Perceptron (MLP) is a type of artificial neural network designed to map a set of input features to a target output through multiple layers of nodes, or neurons. The network consists of an input layer, one or more hidden layers, and an output layer. Each layer is made up of nodes, which represent learned weights, and biases that adjust the input data during the learning process.

The basic structure begins with an input layer, where the raw data is fed into the network. These inputs are then passed to the first hidden layer, where each node applies a linear combination of the input values through a series of weights and biases, followed by a non-linear activation function (e.g., the sigmoid function, ReLU). The activation function allows the network to learn complex patterns and relationships within the data. In a deeper network, additional hidden layers can be added, with each layer using the output from the previous layer as its input, allowing the model to learn increasingly abstract representations of the data.

The final layer is the output layer, which combines the learned features from the hidden layers to produce the prediction. The output layer's activation function depends on the type of problem: for regression tasks, a linear activation is often used, while classification tasks might use a softmax function to output probabilities for different classes. Mathematically, the MLP can be represented as:

$$\hat{y} = f(W_2 \cdot g(W_1 \cdot x + b_1) + b_2)$$

Where: x is the input vector, W_1 and W_2 are weight matrices for the first and second layers, b_1 and b_2 are bias terms, g is the activation function for the hidden layer (e.g., sigmoid or ReLU), f is the activation function for the output layer (e.g., linear or softmax), $\hat{y}$ is the predicted output.

The goal of training the MLP is to adjust the weights W_1, W_2, and biases b_1, b_2 to minimize a chosen loss function, typically the mean squared error (MSE) for regression. This is achieved through backpropagation, an optimization algorithm that computes the gradient of the loss function with respect to the network's weights and biases, allowing for the iterative improvement of model parameters using a method such as stochastic gradient descent (SGD), see Rumelhart, Hinton and Williams (1986) for more details.

3.2 Data

The dataset comprises a diverse range of macroeconomic and financial market variables, providing a comprehensive foundation for nowcasting Peruvian GDP. These variables include domestic activity indicators, such as electricity and oil production, cement consumption, baby chicken placements and goods' supply to wholesale markets; consumer and producer price indices, offering insight into inflationary pressures; as well as Central Reserve Bank of Peru surveys, which capture key sentiment and expectations from various sectors. The dataset also incorporates domestic trade statistics and a variety of international macroeconomic variables, allowing for the assessment of global influences on Peru's economy. Furthermore, domestic financial market variables are also included, capturing essential information about asset prices, interest rates, and currency fluctuations. As specific sectors of Peruvian GDP are vulnerable to natural phenomena, climate data is also included.

In line with Tenorio and Perez (2023), we enhance the dataset by integrating unstructured data from Google Trends. This source provides real-time insights into consumer behavior and economic sentiment. We utilize a variety of keywords, including: "economy", "visa", "huaicos", "toyota", among others. (see section B in the Appendix).

The dataset spans the period from April 2015 to August 2024. This long timespan allows the models to capture both short-term fluctuations and long-term trends in economic activity, providing a robust basis for nowcasting. The inclusion of both structured and unstructured data ensures that the model can draw from a wide variety of information sources, enhancing its predictive power and adaptability to different economic conditions.

3.3 Estimation Strategy

In this section we outline the estimation strategy adopted in this study. As highlighted by Coulombe et al. (2021), the choice of data transformations can significantly influence the accuracy of ML methods.

In light of this, we explored various specifications to assess the performance of ML techniques for nowcasting Peruvian GDP. Specifically, we investigated several key aspects: using variables in YoY percent change or in MoM percent change (seasonally adjusted[1]) for fitting, using expanding versus rolling windows for the training sample[2], comparing K-Fold cross-validation with Walk-Forward cross-validation and evaluating the proper lag structure. Specifications such as the feature matrix rotations and the dimensionality reduction technique were treated as hyperparameters themselves, and were optimized in each nowcast iteration.

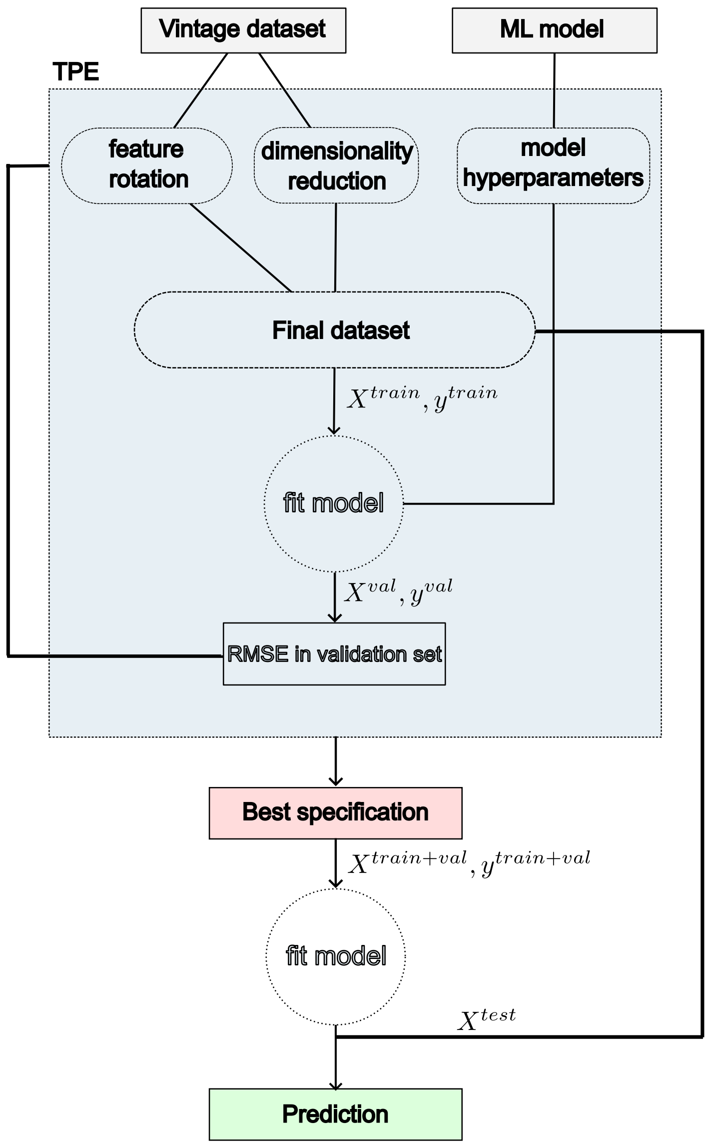
Besides, we evaluated the performance of a bottom-up approach, in which each sector that accounts for the total and non-primary GDP was nowcasted and the projections were assembled externally. Specifically, we nowcasted the sectors of services, commerce, primary industry, non-primary industry and agriculture. We didn't need to nowcast construction, mining and fishing sectors, because their information is known prior to total and non-primary GDP publication. For electricity, water and gas distribution, we assumed the electricity production growth, which accounts for approximately 80 percent of this sector.

The diagram in Figure 1 illustrates the estimation strategy we adopted for a point prediction, building on the detailed methodology described below. This comprehensive approach ensures the deployment of the most effective model configuration, enabling precise and robust GDP nowcasting while effectively capturing its inherent complexities.

[1]We used the TRAMO-SEATS algorithm in the RJDemetra package, seasonally adjusting features and target variable if seasonality was present. As the objective variable is expressed in YoY percent change, we used forecasted factors to transform every models' prediction to this desired form.

[2]In an expanding window approach, the training set starts with an initial set of observations, and, as time progresses, new data points are added to the training set with each nowcast iteration. In contrast, a rolling window approach uses a fixed-size window of observations for training. As time progresses, the oldest data points are dropped from the training set and replaced with the most recent ones

Figure 1: Estimation Strategy



Note: This procedure is employed to obtain point predictions using a given dataset and an ML model. Feature rotations and the dimensionality reduction technique are treated as hyperparameters and are optimized alongside the ML model hyperparameters within the Tree-Structured Parzen Estimator framework.

Data preprocessing: The data used in the analysis is sampled on a monthly basis, and each time series is individually evaluated to determine the most appropriate transformation that maximizes its predictive power for GDP growth. Firstly, series undergo seasonal adjustment[3]

[3]In our analysis, we found that fitting models' with seasonally adjusted variables ensures better out-of-sample performance than doing so with MoM percent change transformations.

to remove recurrent seasonal patterns that could introduce noise or distort the underlying trends relevant for GDP nowcasting. Then, depending on the characteristics of each series, we either transform it into monthly percentage changes, take first differences, or retain it in its level form. These transformations are selected based on their ability to enhance the signal related to GDP dynamics, thereby improving model performance. To further optimize the model, we assess (i) the appropriate lag structure and (ii) feature matrix rotations, to identify the configuration that maximizes predictive accuracy in the validation set. Given the temporal nature of economic data, lagged values of predictors often contain important information about future GDP behavior, and determining the optimal number of lags is crucial for capturing this temporal dependence effectively. Besides, Coulombe et al. (2021) have shown that some feature rotations can also be helpful for predictive performance. In our setting, we introduce eight lags for each variable in the dataset, and let the Tree-Structured Parzen Estimator choose the optimal feature matrix rotations from the space detailed in Table 1:

Table 1: Rotations - Hyperparameters search space

Rotation	Hyperparameter	Prior Distribution
X	Use	{True,False}
MARX	Use	{True,False}
	Order	DiscreteUniform(3,6)
MAF	Use	{True,False}
	N° Components	DiscreteUniform(1,3)

Note: X denotes features in levels, while MARX and MAF represent the moving average rotation of X and moving average factors, respectively. See Coulombe et al. (2021) for more details.

COVID-19 Pandemic : In addressing the impact of the COVID-19 pandemic, and following the recommendations of Schorfheide and Song (2021) and Lenza and Primiceri (2022), we exclude the observations from March 2020 to December 2021 from the estimation of all models. This exclusion is based on the significant economic disruptions caused by the pandemic, which introduced unprecedented volatility and structural breaks in the data. Including these periods could skew the models' predictive accuracy, as the atypical patterns observed during the pandemic are not representative of the usual economic dynamics. In our analysis, we found that removing these observations ensures more reliable and stable model estimates.

Finally, all time series are standardized to ensure that variables with different scales and units are comparable, preventing certain predictors from dominating the ML algorithms due to larger magnitudes.

Dimensionality reduction: Dimensionality reduction techniques allow us to address high-dimensionality and multicollinearity issues, enabling a more robust model for the analysis. This step is potentially important in our setting as many of the features may be highly correlated, specially if we consider the inclusion of feature matrix rotations.

We assess the use of dimensionality reduction techniques as hyperparameters that must be optimized. In particular, following Ng (2013), we set Least Angle Regression (LARs) and Principal Component Analysis (PCA) as dimensionality reduction techniques that can be selected by the Tree-Structured Parzen Estimator from the space detailed in Table 2:

Table 2: Dimensionality reduction - Hyperparameters search space

Model	Hyperparameter	Prior Distribution
Dimensionality reduction	Use	{None, LARS, PCA}
LARS	N° non-zero coefficients	DiscreteUniform$(30, 150)$
PCA	N° Components	Uniform$(0.5, 0.99)$

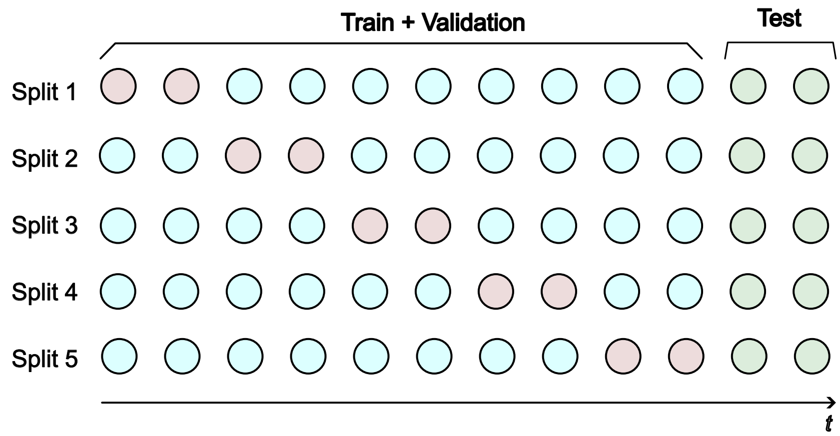
LARs is a regression algorithm that is particularly effective when the number of predictors is large relative to the number of observations. It builds a parsimonious model by iteratively selecting variables based on their correlation with the response variable. At each step, LARs moves the estimated coefficients toward the least-squares solution, but in contrast to traditional stepwise selection, it stops when a new predictor becomes as correlated with the residual as the current predictor. This process continues until all predictors have been incorporated or the model reaches a specified level of sparsity (for further details see Efron et al. (2004)). Meanwhile, PCA is a widely-used method for dimension reduction, which transforms a set of correlated variables into a smaller set of uncorrelated variables known as principal components. Each principal component is a linear combination of the original variables and captures the maximum variance in the data, thereby reducing the dimensionality of the dataset while retaining the most critical information. For a detailed mathematical treatment of PCA, Jolliffe (2002) provides a comprehensive reference.

Sample Split: We divided the dataset into training and test sets, experimenting with various split combinations to find the most appropriate configuration. Ultimately, we established the training set to cover the period from April 2015 to February 2020, while the test set spans from January 2022 to August 2024, deliberately excluding the pandemic period (March 2020 to December 2021) as mentioned before. This ensures that the model is trained on stable, pre-pandemic data and tested on post-pandemic economic conditions, providing a more reliable measure of performance without the distortions caused by the unprecedented volatility during the COVID-19 pandemic.

Cross Validation: To fine-tune the model's hyperparameters, we conducted a cross-validation exercise, using two different methods: K-Fold cross-validation and Walk-Forward cross-validation. Each method has distinct characteristics suited to different types of data, especially in the context of time-series prediction.

K-Fold cross-validation is a widely adopted technique that divides the dataset into K equal-sized subsets or "folds". The model is trained on $K-1$ folds and tested on the remaining fold. This process is repeated K times, with each fold serving as the test set once, and the performance is averaged across all K iterations to obtain a robust estimate of the model's accuracy. For $K=5$, the dataset is split into five parts (see Figure 2). In each iteration, four parts are used for training, and the remaining part is used for validation. This ensures that every observation in the dataset is used both for training and validation, providing a comprehensive assessment of the model's generalization ability.

Figure 2: K-Fold Cross Validation (5 folds)



Walk-forward cross-validation, also known as time-series cross-validation, is specifically designed for time-series data where the temporal order of the observations must be preserved. In this method, the training set begins with a small subset of data, and the model is trained sequentially as more data becomes available. After each training iteration, the model is tested on the next time period. This iterative process simulates real-world nowcasting conditions, where models are always fitted using past data to nowcast future outcomes. However, Walk-Forward cross-validation can be less efficient in terms of data usage because each iteration uses only a portion of the dataset for training and another small portion for testing. This often results in higher variance in performance estimates, particularly in the early iterations when less data is available for model training.

Figure 3: Walk Forward Cross Validation (5 folds)

In our analysis, we found that 5-Fold cross-validation provided better performance than Walk-Forward cross-validation, which is consistent with the findings of Goulet Coulombe et al. (2022). The K-Fold method delivered more stable and reliable results in terms of hyperparameter tuning, likely because it fully utilizes the data in a balanced manner, and is less affected by the temporal dependencies that Walk-Forward cross-validation seeks to preserve.

Hyperparameter Optimization: We explored two of the most widely used techniques for hyperparameter optimization: Grid Search and the Tree-Structured Parzen Estimator. Each of these methods has distinct characteristics and advantages when tuning the hyperparameters of ML models, and they play a critical role in enhancing model performance by identifying the most effective parameter configurations.

Grid Search is an exhaustive search technique that systematically evaluates all possible combinations of hyperparameters from a predefined set of values. For each hyperparameter, a range of values is defined, and the algorithm explores every possible combination of these values, training and validating the model for each case. Although Grid Search guarantees that the best combination within the specified grid is found, it can be computationally expensive, especially when the number of hyperparameters and their possible values is large.

The Tree-Structured Parzen Estimator is a Bayesian optimization technique that uses probabilistic models to guide the search for optimal hyperparameters. Unlike Grid Search, which explore the hyperparameter space without considering previous results, it builds a model of the objective function based on past evaluations and uses this model to predict the most promising areas of the hyperparameter space to explore next. It works by constructing a probability model for the objective function and selecting hyperparameter values that are most likely to improve performance, thus allowing for a more targeted and efficient search.

Tree-Structured Parzen Estimator is particularly advantageous for complex models, such as deep learning architectures (see Watanabe (2023) for more details), where the hyperparameter space is vast and nonlinear. By focusing on promising regions of the hyperparameter space, it significantly reduces the number of evaluations needed to find the optimal configuration. As a result, it is often more effective than traditional methods in scenarios where hyperparameter interactions are complex and non-intuitive. This technique consists of the following steps:

1. Given an initial number of evaluations (N^{init}) to an objective function, define $D := \{(\mathbf{x}_n, y_n)\}_{n=1}^{N^{init}}$ as a set that stores every pair of hyperparameter configuration-objective function value.

2. Divide set D in two subsets: a good (D^l) and a bad (D^g) group of evaluations, given a quantile $y^\gamma, \gamma \in [0,1)$.

3. Construct probability density functions (PDF's) $p(\mathbf{x}|D^l), p(\mathbf{x}|D^g)$ from a prior distribution for each hyperparameter and the use of kernel density estimators (KDE's).

4. Get a sample from the PDF of the good group of evaluations $S := \{\mathbf{x}_s\}_{s=1}^{N_s} \in p(\mathbf{x}|D^l)$.

5. Optimize a surrogate function to obtain an optimal candidate to evaluate the objective function:

$$\mathbf{x}^*_{\mathbf{N+1}} = \underset{\mathbf{x}\in S}{\operatorname{argmax}}\, r(\mathbf{x}|D)$$

$$r(\mathbf{x}|D) = p(\mathbf{x}|D^l)/p(\mathbf{x}|D^g)$$

6. Evaluate the objective function with the optimal candidate: $\mathbf{y}^*_{\mathbf{N+1}} = f(\mathbf{x}^*_{\mathbf{N+1}})$

7. Update set $D \leftarrow D \cup \{(\mathbf{x}^*_{\mathbf{N+1}}, \mathbf{y}^*_{\mathbf{N+1}})\}$.

8. Iterate over 1-7 until maximum number of evaluations to the objective function has been reached.

9. Select the hyperparameter configuration that optimized the objective function.

After evaluating the performance and efficiency of all three methods, we selected Tree-Structured Parzen Estimator as our preferred hyperparameter optimization technique. Its ability to iteratively refine its search based on previous results makes it far more efficient than Grid Search, particularly in complex models. The models' hyperparameters search space is defined in Table 3:

Table 3: ML Models - Hyperparameters search space

Model	Hyperparameter	Prior Distribution
Lasso	Alpha	Uniform$(10^{-3}, 3*10^{-1})$
Ridge	Alpha	Uniform$(10^{-1}, 10^{2})$
Elastic Net	Alpha	Uniform$(10^{-3}, 3*10^{-1})$
	L1 ratio	Uniform$(10^{-4}, 1)$
SVR	Gamma	Uniform$(10^{-7}, 0.5)$
	C	Uniform$(1, 10^{5})$
Decision Tree	Max. Depth	Discrete Uniform(3,100)
	Min. samples for leaf	DiscreteUniform(1,20)
	Min. samples for split	DiscreteUniform(2,20)
	Max. leaf nodes	DiscreteUniform(5,20)
KNN	N Neighbors	DiscreteUniform(2,30)
	Weights	{uniform,distance}
Random Forest	Max. Depth	DiscreteUniform(3,100)
	Min. samples for leaf	DiscreteUniform(4,10)
	Min. samples for split	DiscreteUniform(4,10)
	N. Estimators	DiscreteUniform(30,200)
AdaBoost	Learning Rate	Uniform$(10^{-4}, 1)$
	Loss	{linear,squared,exponential}
	N. Estimators	DiscreteUniform(30,200)
GBoost	Learning Rate	Uniform$(10^{-4}, 1)$
	Max. Depth	DiscreteUniform(3,100)
	Min. samples for leaf	DiscreteUniform(4,10)
	Min. samples for split	DiscreteUniform(4,10)
	N. Estimators	DiscreteUniform(30,200)
XGBoost	Columns per tree	Uniform$(10^{-2}, 0.99)$
	Gamma	Uniform$(10^{-3}, 0.99)$
	Learning Rate	Uniform$(10^{-4}, 1)$
	Subsample	Uniform$(0.5, 0.99)$
	Max. Depth	DiscreteUniform(3,100)
	N. Estimators	DiscreteUniform(30,200)
Bagging	N. Estimators	DiscreteUniform(5,20)
	Max. Samples	Uniform$(10^{-2}, 1)$
	Max. Features	Uniform$(3*10^{-2}, 1)$
	Bootstrap	{True,False}
	Bootstrap Features	{True,False}
MLP	N. Layers	DiscreteUniform(1,7)
	Neurons per layer	DiscreteUniform(1,15)
	Activation	{Identity, Logistic, Tanh, ReLU}
	Alpha	Uniform$(10^{-8}, 0.99)$
	Batch size	DiscreteUniform(10,20)
	Beta 1	Uniform$(10^{-2}, 0.99)$
	Beta 2	Uniform$(10^{-2}, 0.99)$

Model Evaluation: Following the established practices in the GDP nowcasting literature, we assess the predictive performance of our models using the Root Mean Square Error (RMSE), a widely used metric for evaluating the accuracy of predictions. RMSE provides a direct measure of the average magnitude of the prediction error, penalizing larger errors more heavily, thus making it an effective tool for comparing model performance.

For a given model m, the RMSE is computed by taking the square root of the mean of the squared differences between the actual values y_t and the predicted values $\hat{y}_t$. The RMSE for each model is calculated as follows:

$$RMSE^m = \sqrt{\frac{1}{32} \sum_{t=T-32}^{T} (y_t - \hat{y}_t^m)^2}$$

where y_t is the actual observed value of GDP at time t, $\hat{y}_t^m$ is the nowcast generated by model m at time t, T represents August 2024 and the window of 32 observations refers to the test period (e.g., the last 32 months).

4 Results

In order to evaluate and compare the performance of various ML methods, we performed an out-of-sample nowcasting exercise from January 2022 to August 2024. We conducted four separate analyses: for each target variable (total and non-primary GDP) and for each method (direct versus bottom-up).

Tables 4 and 5 display the point nowcasts for YoY total GDP growth for the bottom-up and direct approach, respectively. The first column represents the actual GDP values, while the subsequent columns display the predictions for each ML model, the simple average of all ML models (Mean ML), and the DFM, which serves as the benchmark.

From Table 4, we observe that all ML models report lower RMSE values than the DFM. Models such as XGBoost and Elastic Net consistently deliver robust predictions, with RMSE values around 0,6, showing their strength in handling non-linear patterns in GDP data.

Comparing tables 4 and 5, we observe that the bottom-up approach delivered lower RMSE values for every ML model. By assembling sectorial nowcasts, it is possible to capture episodes of high volatility more precisely. For instance, in the second quarter of 2024, the agricultural and fishery sectors experienced an important bounce that could not be captured well by the direct approach. Intuitively, it may be easier for ML models to identify the idiosyncratic movements of each sector than the movement of a series (total GDP) that is an aggregation of all sectors.

Table 4: Nowcast of YoY GDP (bottom-up)

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	2,8	4,4	4,0	4,1	3,6	3,1	4,1	3,9	3,8	3,4	3,6	3,6	3,9	3,8	4,0
Feb-22	4,7	5,2	5,4	5,3	5,4	5,2	5,5	5,4	5,6	5,3	5,6	5,3	5,3	5,4	5,9
Mar-22	3,8	3,8	3,8	3,7	3,9	3,5	3,7	3,7	3,6	3,6	3,5	3,6	3,9	3,7	4,0
Apr-22	4,0	3,6	3,6	3,7	3,5	3,4	3,9	3,6	3,5	3,2	3,7	3,5	3,5	3,6	4,3
May-22	2,6	2,4	2,5	2,5	2,4	2,5	2,1	2,4	2,4	2,3	2,3	2,5	2,3	2,4	1,7
Jun-22	3,5	3,1	3,2	3,3	3,0	3,6	3,4	3,5	3,4	3,7	3,5	3,3	3,2	3,4	3,6
Jul-22	1,8	2,4	2,3	2,3	2,5	3,3	2,5	2,4	2,6	2,5	2,6	2,5	2,5	2,5	2,8
Aug-22	2,0	1,8	1,9	1,8	2,1	2,3	1,8	2,0	1,8	1,8	2,1	1,8	1,8	1,9	1,9
Sep-22	2,1	1,5	1,7	1,6	1,2	1,4	1,5	1,6	1,6	1,9	1,5	1,6	1,6	1,6	1,7
Oct-22	2,3	3,0	2,7	2,8	2,5	3,3	2,8	3,0	3,3	3,1	3,0	3,3	2,7	3,0	3,5
Nov-22	2,1	2,5	2,4	2,5	2,3	2,4	2,9	2,6	2,6	2,2	2,3	2,5	2,3	2,4	2,7
Dec-22	1,0	2,2	2,3	2,3	2,1	2,9	2,1	2,1	2,3	2,1	2,4	2,3	2,4	2,3	1,8
Jan-23	-0,9	0,3	0,3	0,1	0,5	0,2	0,8	0,4	0,5	0,5	0,4	0,4	0,3	0,4	0,4
Feb-23	-0,6	-0,8	-0,9	-0,8	-0,8	-1,1	-0,5	-0,8	-0,9	-1,0	-0,9	-1,0	-1,0	-0,9	-1,0
Mar-23	0,3	0,6	-0,4	0,6	0,1	0,2	0,4	0,4	0,1	0,3	0,1	0,3	0,3	0,3	-0,2
Apr-23	0,4	0,5	0,4	0,5	0,7	0,9	1,3	1,2	0,9	0,9	0,9	0,6	0,4	0,8	1,1
May-23	-1,3	-1,1	-0,7	-0,9	-1,0	-0,8	-0,4	-0,4	-0,9	-0,6	-0,7	-0,5	-0,7	-0,7	-0,9
Jun-23	-0,6	-0,7	-0,5	-0,9	-0,5	-0,7	-0,8	-0,7	-0,9	-0,9	-0,7	-0,8	-0,5	-0,7	-1,5
Jul-23	-1,2	-0,3	-0,3	-0,3	-0,1	-0,1	-0,3	-0,4	-0,2	0,1	0,1	0,1	-0,5	-0,2	-0,4
Aug-23	-0,4	-0,8	-0,8	-0,6	-0,8	-0,4	-0,9	-1,0	-0,5	-0,5	-0,5	-0,6	-0,7	-0,7	-1,2
Sep-23	-1,2	-1,0	-1,1	-1,1	-1,4	-0,4	-1,0	-0,7	-0,8	-0,8	-0,8	-0,6	-1,0	-0,9	-0,3
Oct-23	-0,7	-0,4	-0,4	-0,5	-0,3	-0,5	-0,4	-0,5	-0,4	-0,5	-0,6	-0,5	-0,4	-0,5	-0,4
Nov-23	0,3	-0,5	-0,6	-0,5	-0,7	0,1	-0,4	-0,2	-0,2	-0,2	-0,2	-0,1	-0,7	-0,4	0,4
Dec-23	-0,7	-1,2	-0,8	-0,9	-0,7	-0,8	-0,7	-0,6	-1,2	-1,0	-0,7	-1,0	-1,0	-0,9	-0,5
Jan-24	1,5	1,3	1,3	1,3	1,3	1,1	1,1	1,1	1,2	1,1	1,0	1,1	1,2	1,2	0,8
Feb-24	3,2	3,4	3,2	3,2	3,6	2,7	3,4	3,2	2,8	2,9	3,3	3,1	3,4	3,2	3,6
Mar-24	-0,4	-0,7	-0,5	-0,8	-0,6	-0,7	-0,9	-0,8	-0,8	-0,3	-0,6	-0,7	-0,8	-0,7	-0,7
Apr-24	5,4	4,7	4,4	4,5	4,3	4,6	3,6	3,9	4,6	4,9	4,8	4,5	4,2	4,4	3,4
May-24	5,3	5,1	5,0	5,1	5,5	5,8	4,9	5,5	5,4	5,2	5,4	5,5	4,9	5,3	6,2
Jun-24	0,3	1,5	1,4	1,5	1,4	1,3	1,9	1,6	1,1	1,2	1,0	1,0	1,4	1,4	1,8
Jul-24	4,6	3,5	3,5	3,4	3,6	3,5	3,7	3,5	3,2	3,4	3,8	3,7	3,5	3,5	3,7
Aug-24	3,7	3,8	3,6	3,5	3,4	3,3	3,4	3,4	3,4	3,4	3,5	3,3	3,6	3,5	2,9
RMSE		0,636	0,621	0,597	0,630	0,696	0,773	0,683	0,647	0,610	0,590	0,613	0,634	0,612	0,840

Table 5: Nowcast of YoY GDP (direct)

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	2,8	2,9	2,5	3,0	2,6	2,9	2,2	2,4	2,5	2,4	2,3	2,7	3,0	2,6	3,6
Feb-22	4,7	3,2	3,1	3,1	3,9	2,6	3,4	3,3	3,3	3,3	3,4	3,4	3,2	3,3	4,6
Mar-22	3,8	5,2	4,6	5,1	5,0	4,1	4,8	4,0	4,2	3,9	4,1	4,7	5,1	4,6	3,9
Apr-22	4,0	3,6	4,2	4,1	3,7	3,0	4,0	4,0	4,0	3,8	4,1	4,2	3,8	3,9	5,2
May-22	2,6	3,5	3,4	3,2	3,2	3,7	3,2	3,1	3,1	3,7	3,4	3,1	3,2	3,3	2,0
Jun-22	3,5	2,4	2,7	2,6	2,7	2,3	2,5	2,5	2,7	2,8	2,8	2,7	2,6	2,6	3,6
Jul-22	1,8	3,1	3,1	3,3	3,0	3,3	3,1	3,2	3,6	3,3	3,3	3,6	3,1	3,3	4,0
Aug-22	2,0	2,0	2,1	2,2	2,0	1,9	2,1	1,9	1,9	2,1	1,6	1,9	2,0	2,0	1,1
Sep-22	2,1	1,6	1,6	1,6	1,6	1,3	1,3	1,4	1,4	1,0	1,5	1,7	1,5	1,4	1,1
Oct-22	2,3	2,5	2,5	2,5	2,4	3,3	2,6	2,8	2,8	2,9	2,7	2,9	2,5	2,7	3,7
Nov-22	2,1	2,5	2,6	2,6	2,2	2,2	2,7	2,7	2,6	2,6	2,9	2,6	2,5	2,6	2,7
Dec-22	1,0	2,1	2,1	1,9	2,3	1,7	1,9	1,9	2,2	2,3	1,8	1,8	1,9	2,0	1,0
Jan-23	-0,9	0,6	0,5	0,7	1,2	1,2	0,7	1,7	1,7	1,6	1,5	1,6	0,9	1,1	2,4
Feb-23	-0,6	-1,9	-2,2	-2,2	-2,0	-1,5	-1,0	-0,8	-1,2	-1,4	-0,7	-0,9	-2,3	-1,5	-1,3
Mar-23	0,3	0,6	0,5	0,6	0,6	-0,8	0,2	0,2	0,4	0,6	0,5	0,5	0,5	0,4	-1,3
Apr-23	0,4	0,2	0,0	0,0	-0,3	0,1	0,4	0,0	-0,3	-0,5	0,0	-0,1	0,2	0,0	-0,1
May-23	-1,3	0,0	0,2	0,0	0,3	0,4	0,7	0,6	0,3	0,3	0,2	0,4	0,1	0,3	0,2
Jun-23	-0,6	-1,3	-1,4	-1,0	-1,1	-1,3	-1,6	-1,3	-1,1	-1,3	-1,4	-1,1	-1,4	-1,3	-1,5
Jul-23	-1,2	-0,1	-0,1	-0,1	0,2	-0,6	-0,3	-0,3	-0,5	-0,3	-0,3	-0,4	0,0	-0,2	1,4
Aug-23	-0,4	-0,8	-0,8	-0,2	-0,8	0,1	-1,0	-0,7	-0,8	-0,1	-0,7	-0,7	-0,7	-0,6	-1,5
Sep-23	-1,2	-0,7	-0,6	-0,6	-0,8	-0,4	-0,7	-0,5	-0,6	-0,6	-0,9	-0,6	-0,7	-0,6	-0,4
Oct-23	-0,7	-0,7	-0,9	-0,9	-0,6	-0,2	-0,8	-0,5	-0,6	-0,5	-0,3	-0,5	-0,9	-0,6	-0,8
Nov-23	0,3	-1,0	-0,9	-1,1	-0,9	0,0	-0,3	-0,7	-0,6	-0,5	-0,8	-0,7	-0,9	-0,7	-0,1
Dec-23	-0,7	0,3	0,4	0,4	0,1	0,1	0,5	0,3	0,5	0,6	0,3	0,1	0,4	0,3	-0,8
Jan-24	1,5	2,2	1,5	2,7	2,5	1,1	1,8	0,9	1,0	1,1	0,6	0,7	1,9	1,5	1,3
Feb-24	3,2	2,3	1,9	2,3	1,9	2,1	2,3	1,9	1,5	1,5	2,0	1,9	2,8	2,0	3,3
Mar-24	-0,4	-1,0	-1,1	-1,0	-1,0	0,4	-0,8	-0,3	-1,1	-1,1	-0,1	-1,1	-1,1	-0,8	0,1
Apr-24	5,4	3,0	3,0	3,1	3,3	2,6	2,9	2,6	2,7	2,8	2,7	2,6	3,1	2,9	2,8
May-24	5,3	3,1	3,3	3,3	3,1	3,8	3,8	3,9	4,1	3,9	3,2	3,9	3,3	3,6	4,9
Jun-24	0,3	2,5	2,7	2,6	2,9	3,9	3,4	3,1	2,8	2,4	3,1	3,2	2,6	2,9	2,6
Jul-24	4,6	3,4	3,1	3,3	3,1	2,5	3,1	2,9	3,2	2,6	3,4	2,7	3,1	3,0	4,4
Aug-24	3,7	3,0	3,2	3,1	3,2	2,6	3,3	3,1	3,2	3,2	3,5	3,4	3,2	3,2	3,0
RMSE		1,100	1,123	1,120	1,142	1,302	1,127	1,178	1,154	1,187	1,150	1,181	1,108	1,114	1,245

Tables 6 and 7 present the monthly nowcasts of YoY non-primary GDP growth for the bottom-up and direct approach, respectively. Non-primary GDP is particularly important because it typically has less real-time information compared to total GDP, making its nowcasting crucial for policymakers and economic analysts.

We notice that ML models produced accurate predictions for non-primary GDP, with RMSE values consistently around 0,7 and 0,8. These models perform relatively well in economic downturns (second half of 2023) or in volatile periods (e.g., first half of 2024). DFM performed similarly to ML models, achieving an RMSE of 0,7, indicating that the ML models are on par with the more traditional DFM in terms of prediction accuracy.

Nevertheless, models struggled to nowcast the rapid slowdown observed in January 2023. Although the magnitude of the error was lower for the bottom-up approach, in each of our four exercises these models failed to account for the economic deterioration caused by social conflicts. In this period, the DFM did a better job in capturing non-primary GDP movements.

Table 6: Nowcast of YoY Non-Primary GDP (bottom-up)

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	2,9	4,3	4,0	4,0	3,9	4,0	4,0	4,1	4,1	4,0	4,0	4,1	4,1	4,1	3,9
Feb-22	6,1	6,5	6,5	6,5	6,5	6,5	6,5	6,4	6,6	6,3	6,5	6,4	6,5	6,5	7,1
Mar-22	5,3	5,1	4,9	4,8	5,0	4,9	4,7	4,8	4,7	4,8	4,7	4,7	4,9	4,8	5,3
Apr-22	5,1	4,7	4,6	4,6	4,5	4,6	4,7	4,7	4,6	4,7	4,7	4,7	4,6	4,6	4,9
May-22	4,4	4,6	4,3	4,4	4,4	4,5	4,4	4,4	4,6	4,5	4,5	4,5	4,4	4,5	4,3
Jun-22	3,7	3,8	3,8	3,8	3,9	3,9	3,8	4,0	3,9	4,0	4,1	3,9	3,8	3,9	4,0
Jul-22	2,1	3,0	3,0	2,9	3,0	3,5	3,1	3,0	3,1	3,0	3,0	2,9	3,1	3,0	2,9
Aug-22	2,8	2,4	2,3	2,4	2,5	3,0	2,4	2,6	2,5	2,4	2,6	2,5	2,2	2,5	2,6
Sep-22	2,7	2,0	2,2	2,0	2,0	1,8	2,0	1,9	2,0	2,3	1,9	2,0	2,1	2,0	2,0
Oct-22	2,1	3,0	2,9	3,0	2,9	2,8	2,9	3,0	3,0	2,9	2,9	3,2	2,9	3,0	2,8
Nov-22	2,1	2,4	2,3	2,4	2,3	2,6	2,6	2,4	2,5	2,5	2,5	2,7	2,4	2,5	2,3
Dec-22	-0,2	1,4	1,4	1,2	1,3	1,3	1,4	1,4	1,3	1,4	1,4	1,4	1,4	1,4	1,3
Jan-23	-1,9	-0,1	0,0	0,0	0,0	-0,3	0,2	0,0	0,1	0,1	-0,1	-0,1	0,1	0,0	-0,7
Feb-23	-1,6	-1,8	-1,9	-1,7	-1,9	-2,0	-1,8	-1,9	-1,9	-1,9	-1,8	-1,9	-1,8	-1,9	-2,2
Mar-23	-1,9	-2,1	-2,1	-2,0	-2,1	-2,2	-2,1	-2,0	-2,3	-2,2	-2,1	-2,1	-2,1	-2,1	-2,3
Apr-23	-1,1	-2,1	-2,0	-2,0	-1,9	-1,4	-1,7	-1,7	-1,7	-1,8	-1,8	-1,7	-2,0	-1,8	-1,8
May-23	-1,4	-1,2	-1,1	-1,2	-1,2	-0,9	-1,0	-1,1	-1,2	-1,1	-1,1	-1,2	-1,1	-1,1	-1,1
Jun-23	-0,7	-1,4	-1,2	-1,3	-1,4	-1,4	-1,2	-1,2	-1,4	-1,3	-1,4	-1,2	-1,2	-1,3	-1,2
Jul-23	-2,0	-0,5	-0,5	-0,5	-0,4	-0,7	-0,6	-0,5	-0,5	-0,3	-0,5	-0,5	-0,7	-0,5	-0,6
Aug-23	-1,8	-2,1	-2,0	-1,8	-2,0	-1,8	-1,6	-1,8	-1,9	-1,8	-1,8	-1,9	-2,0	-1,9	-1,8
Sep-23	-2,5	-2,7	-2,7	-2,7	-2,9	-2,2	-2,5	-2,4	-2,4	-2,5	-2,3	-2,3	-2,8	-2,5	-2,1
Oct-23	-1,5	-1,1	-1,2	-1,2	-1,1	-1,3	-1,1	-1,4	-1,4	-1,4	-1,4	-1,4	-1,2	-1,3	-1,1
Nov-23	-1,6	-1,5	-1,6	-1,6	-1,6	-1,5	-1,5	-1,5	-1,5	-1,6	-1,6	-1,5	-1,6	-1,6	-1,0
Dec-23	0,3	-1,1	-0,8	-0,7	-0,6	-0,7	-0,8	-0,8	-0,8	-1,0	-0,7	-0,8	-1,0	-0,8	-0,2
Jan-24	2,3	2,3	2,2	2,3	2,3	2,4	2,2	2,1	2,2	2,2	2,2	2,2	2,2	2,2	1,9
Feb-24	2,7	2,2	2,1	2,3	2,4	1,9	2,3	2,2	2,0	2,0	2,3	2,1	2,3	2,2	2,5
Mar-24	-0,3	-0,7	-0,7	-0,7	-0,6	-0,5	-0,6	-0,6	-0,7	-0,5	-0,5	-0,5	-0,7	-0,6	-0,6
Apr-24	3,9	4,0	3,8	4,0	3,8	3,5	3,7	3,6	3,7	3,8	3,9	3,7	3,8	3,8	3,8
May-24	2,5	2,3	2,2	2,3	2,5	2,7	2,5	2,4	2,3	2,3	2,3	2,3	2,3	2,4	3,1
Jun-24	1,1	1,4	1,3	1,3	1,3	1,3	1,0	1,0	1,0	1,0	1,0	1,0	1,4	1,1	1,2
Jul-24	5,1	3,8	3,8	3,8	3,9	4,0	3,9	3,8	3,6	3,9	3,9	3,8	3,8	3,8	3,8
Aug-24	3,6	3,3	3,1	3,0	3,0	3,1	3,4	3,3	3,2	3,3	3,3	3,2	3,2	3,2	3,0
RMSE		0,763	0,737	0,720	0,718	0,719	0,742	0,722	0,768	0,742	0,721	0,731	0,743	0,726	0,672

Table 7: Nowcast of YoY Non-Primary GDP (direct)

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	2,9	3,0	2,9	3,0	2,9	2,9	3,1	3,2	3,2	3,0	3,2	3,3	3,0	3,1	2,8
Feb-22	6,1	6,3	6,3	6,4	6,5	6,7	6,8	6,8	7,0	6,7	7,1	6,7	6,3	6,6	7,1
Mar-22	5,3	5,6	5,7	5,6	5,5	5,3	5,5	5,5	5,8	6,0	5,7	5,7	5,7	5,6	5,6
Apr-22	5,1	4,5	4,6	4,4	4,4	4,6	4,8	4,6	4,8	4,7	4,4	4,4	4,6	4,6	5,4
May-22	4,4	5,1	5,1	4,9	5,1	5,0	5,0	5,0	5,2	5,2	5,0	5,1	5,1	5,1	4,3
Jun-22	3,7	4,0	4,1	4,2	4,3	4,6	4,0	4,2	4,2	4,1	4,3	4,2	4,3	4,2	4,1
Jul-22	2,1	3,1	3,4	3,0	3,2	3,6	3,2	3,3	3,4	3,4	3,2	3,3	3,1	3,3	3,0
Aug-22	2,8	2,4	2,3	2,2	2,5	2,4	2,4	2,5	2,7	2,9	2,6	2,5	2,2	2,5	2,3
Sep-22	2,7	1,8	1,6	1,8	2,1	1,9	1,8	1,9	1,8	1,7	1,8	1,8	1,8	1,8	1,9
Oct-22	2,1	3,1	3,0	3,0	3,0	3,0	2,8	2,7	2,8	2,8	2,5	2,9	3,0	2,9	2,6
Nov-22	2,1	1,8	1,8	2,1	1,9	2,1	2,0	2,1	2,1	2,1	1,9	2,0	2,0	2,0	1,6
Dec-22	-0,2	1,0	0,9	1,0	1,3	1,5	1,5	1,4	1,3	1,2	1,3	1,5	1,1	1,2	0,5
Jan-23	-1,9	0,6	0,5	0,5	0,6	1,3	0,9	1,0	1,0	1,2	1,4	1,0	0,5	0,9	0,3
Feb-23	-1,6	-1,4	-1,5	-1,5	-1,7	-1,5	-1,5	-1,6	-1,5	-1,5	-2,0	-1,8	-1,6	-1,6	-2,2
Mar-23	-1,9	-1,5	-1,6	-1,6	-1,8	-0,7	-1,5	-1,4	-1,3	-1,2	-1,2	-1,2	-1,5	-1,4	-2,2
Apr-23	-1,1	-2,3	-2,4	-2,4	-2,3	-2,2	-2,3	-2,2	-2,2	-2,1	-2,2	-2,3	-2,3	-2,3	-2,2
May-23	-1,4	-0,3	-0,5	-0,5	-0,4	-0,1	-0,2	-0,2	-0,2	-0,3	-0,3	0,0	-0,5	-0,3	-0,4
Jun-23	-0,7	-1,3	-1,4	-1,4	-1,4	-1,0	-1,4	-1,4	-1,4	-1,3	-1,4	-1,6	-1,3	-1,4	-1,5
Jul-23	-2,0	-0,5	-0,5	-0,5	-0,4	-0,7	-0,4	-0,7	-0,7	-0,8	-0,8	-0,8	-0,4	-0,6	-0,3
Aug-23	-1,8	-1,6	-1,5	-1,4	-1,5	-1,1	-1,5	-1,5	-1,6	-1,4	-1,6	-1,5	-1,6	-1,5	-1,9
Sep-23	-2,5	-2,2	-2,1	-2,1	-2,1	-1,9	-2,1	-2,0	-2,1	-1,9	-2,1	-1,9	-2,1	-2,1	-1,8
Oct-23	-1,5	-1,6	-1,5	-1,4	-1,3	-1,6	-1,5	-1,4	-1,4	-1,6	-1,5	-1,4	-1,5	-1,5	-1,2
Nov-23	-1,6	-1,6	-1,9	-1,8	-1,8	-1,3	-1,7	-1,7	-1,8	-1,6	-1,5	-1,7	-1,9	-1,7	-1,1
Dec-23	0,3	-1,5	-1,5	-1,5	-1,6	-1,4	-1,5	-1,4	-1,6	-1,5	-1,4	-1,3	-1,4	-1,5	-1,0
Jan-24	2,3	2,6	2,6	2,5	2,4	1,6	1,7	1,6	1,5	1,8	1,8	1,6	2,6	2,0	1,3
Feb-24	2,7	2,3	2,3	2,3	2,1	2,6	2,2	2,1	2,2	2,2	2,2	2,2	2,3	2,3	2,5
Mar-24	-0,3	-0,2	-0,3	-0,3	-0,4	-0,2	-0,2	0,0	-0,1	0,0	-0,2	-0,2	-0,4	-0,2	0,1
Apr-24	3,9	3,4	3,3	3,3	3,4	3,5	3,5	3,5	3,6	3,6	3,4	3,5	3,2	3,4	3,6
May-24	2,5	2,0	1,9	2,3	2,1	2,4	2,0	2,0	2,0	2,3	1,8	2,1	2,2	2,1	3,0
Jun-24	1,1	1,7	1,6	1,8	1,7	1,1	1,3	1,4	1,4	1,6	1,5	1,3	1,7	1,5	1,6
Jul-24	5,1	3,0	3,1	3,0	3,1	3,0	3,1	3,1	3,1	3,3	3,2	3,0	3,2	3,1	3,4
Aug-24	3,6	3,6	3,5	3,6	3,5	3,0	3,5	3,5	3,4	3,3	3,6	3,5	3,5	3,5	2,9
RMSE		0,907	0,907	0,885	0,919	1,035	0,952	0,951	0,966	0,950	0,977	0,979	0,885	0,927	0,841

Regarding hyperparameter values, some interesting evidence arises. Dimensionality reduction techniques are preferred by Tree-Structured Parzen Estimator across target variables and ML models. Furthermore, LARs stands out as a better strategy than PCA for all ML models, except Decision Tree, that sometimes performs better with PCA. On the other hand, we find mixed evidence regarding feature matrix rotations. While the Tree-Structured Parzen Estimator tends to use MARX more often that MAF or X for most out-of-sample nowcasts, the frequency of this election is not as absolute as LARs election for the dimensionality reduction techniques. While LARs was chosen 96 percent of the time (2581 out of 2688 times), MARX was chosen around 70 percent of the time. In Table 8, we observe the relative frequency of use of every hyperparameter in the out-of-sample exercises:

Table 8: Hyperparameters - Frequency of usage in Out-of-Sample exercises
(Percentage points)

Dimensionality reduction			**Rotations**		
LARs	PCA	None	MARX	X	MAF
96,0	3,6	0,3	69,6	44,4	31,8

5 Conclusion

In this paper, we applied ML techniques to nowcast total and non-primary peruvian GDP. By testing a wide range of model specifications, we compared the accuracy of ML methods against the DFM, a standard benchmark for nowcasting. Our findings show that ML models performed relatively well in this task, delivering lower RMSE values than the DFM for total GDP, and similar values for non-primary GDP.

One of the key contributions of this study is the application of a bottom-up approach for nowcasting, a procedure that involved nowcasting disaggregated sectorial components of total and non-primary GDP. By doing this, we were able to reduce the RMSE by around 45 and 22 percent for each target variable, respectively. Besides that, we explored the usefulness of adding new features and reducing the dimensionality of the feature matrix for enhancing predictability. The inclusion of moving averages of features and the use of LARs for dimensionality reduction were the preferred procedures most of the time.

An important avenue for future work involves extending the scope from nowcasting to forecasting. While nowcasting focuses on the current state of the economy, incorporating forecasting elements could provide insights into future economic trends. This would further enhance the utility of ML techniques in economic decision-making.

Besides, we plan to explore methods for improving the interpretability of ML models, such as SHAP (Shapley Additive Explanations) or LIME (Local Interpretable Model-agnostic Explanations). Making ML results more interpretable will not only boost their reliability but also increase their acceptance among policymakers and economists, who often require a clear understanding of the underlying drivers of the predictions.

In conclusion, while ML techniques have proven to be highly effective in the nowcasting of Peruvian GDP, there are still exciting opportunities to expand and refine their application. By extending into forecasting and improving interpretability, future research in ML for economic predictions can continue to advance.

A Dynamic Factor Model for Nowcasting Monthly GDP in Peru

The methodology of this document follows the theory defined by Bańbura et al. (2013) and Mariano and Murasawa (2010), and a modified version of the implementation in Python made by Fulton (2020). The DFM posits that a small number of unobservable factors can be used to explain a substantial portion of the variation and dynamics of a large number of observable variables. These observable variables can consist of dozens or even hundreds of series, so estimating dynamic factors serves as a dimension reduction technique. The estimated factors can then be used for forecasting and nowcasting. The model is specified as follows:

$$
\begin{aligned}
y_t &= \Lambda f_t + \epsilon_t \\
f_t &= A_1 f_{t-1} + \ldots + A_p f_{t-p} + u_t,
\end{aligned}
$$

where $y_t = (y_{1,t}, y_{2,t}, \ldots, y_{n,t})'$ denote monthly series, transformed to meet stationarity. ϵ_t are the idiosyncratic disturbances at time t, f_t is an $r \times 1$ vector containing unobservable common factors, modeled as a VAR process of order p. $u_t \sim N(0, Q)$ are the disturbances associated with the dynamic factors. Additionally, Λ is the matrix of factor loadings, and A_i are the autoregressive coefficient $r \times r$ matrices. Lastly, we allow the idiosyncratic component of the monthly series to follow an AR(1) process:

$$
\epsilon_{i,t} = \alpha_i \epsilon_{i,t-1} + e_{i,t}, \text{ with } e_{i,t} \sim i.i.d.N(0, \sigma_i^2) \text{ and } \mathbb{E}[e_{i,t}, e_{j,t}] = 0, \forall i \neq j
$$

The model specification allows for the separation of unobserved factors into two categories: (i) global factors, which exhibit cross-sectional comovement across all groups of explanatory variables (such as coincident/leading economic indicators, employment, credit, fiscal accounts, among others), and (ii) group-specific factors for each group of variables.

To see this, we restrict Λ, $A_1, A_2, \ldots, A_p$ and Q, in order to partition f_t into mutually independent global (g) and m group-specific (s) factors:

$$\Lambda = \begin{pmatrix} \Lambda_{g,s_1} & A_{s_1,s_1} & 0 & \dots & 0 \\ \Lambda_{g,s_2} & 0 & A_{s_2,s_2} & \dots & 0 \\ \vdots & \vdots & \dots & \ddots & \vdots \\ \Lambda_{g,s_m} & 0 & \dots & 0 & A_{s_m,s_m} \end{pmatrix}$$

$$f_t = \begin{pmatrix} f_t^g \\ f_t^{s_1} \\ \vdots \\ f_t^{s_m} \end{pmatrix}, \quad A_i = \begin{pmatrix} A_{i,g} & 0 & \dots & 0 \\ 0 & A_{i,s_1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_{i,s_m} \end{pmatrix}, \quad Q = \begin{pmatrix} Q_g & 0 & \dots & 0 \\ 0 & Q_{s_1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q_{s_m} \end{pmatrix}$$

This framework accounts for cross-sectional correlation within the group-specific blocks, allowing for a more efficient estimation of the global factor(s). Moreover, A_i, g is not necessarily a diagonal matrix, and thus it may allow the existence of multiple correlated global factors.

A.1 Data

The initial set of variables includes up to 182 series, spanning from January 2006 to September 2024. This set consists of structured series such as industrial activity indicators, prices, fiscal accounts, trade balance, terms of trade, employment statistics, expectations survey indices, financial system data, and equity market information. On the other hand, non-structured series are represented by the number of relevant searches, gathered through Google Trends, related to crises, expenditure, government transfers, among other topics.

To reduce the number of series to a more relevant subset, we apply a LASSO model, where hyperparameter optimization within the period 2011-2019 resulted in a preselection of 155 variables.

Lastly, we transform the variables as follows. First, we apply seasonal adjustment to the entire reduced dataset, using the software JDemetra+. Next, variables originally expressed in levels are converted to month-on-month variations to meet the stationarity assumption.

Finally, we remove observations that are more than 10 times the Interquartile Range (IQR = $Q_3 - Q_1$, where Q_i is the i-th quartile) from the mean. This outlier removal implicitly excludes most of the COVID-19 period.

A.2 Estimation

The details of the state-space representation can be found in Mariano and Murasawa (2010) and Bańbura et al. (2013). The literature on nowcasting with DFM estimates $\theta = (\Lambda, A, Q, \sigma^2)$ using maximum likelihood, through the Expectation-Maximization (EM) algorithm, which allows for the handling of missing observations. This algorithm roughly consists of iterating a two-step approach, while treating the unobserved factors as latent variables: (i) given a current estimate of θ, we compute the expected value of the log-likelihood function of the complete data (observed data and latent factors); and then (ii) we maximize the expected log-likelihood with respect to the parameters θ, yielding new estimates of the parameters.

The algorithm works as follows: we begin with an initial guess for θ. In the first step (Expectation Step), we use a Kalman Filter and Smoother to compute the conditional expectation of the latent factors f_t and their covariances, in order to calculate the expected value of the complete data log-likelihood, which depends on the latent factors. In the second step (Maximization Step), the expected log-likelihood from the previous step is maximized with respect to the parameters. We repeat the process until convergence, i.e., until the changes in the log-likelihood between iterations become sufficiently small.

The algorithm is implemented via the DynamicFactorMQ class included in the statsmodels library in Python. We compared the performance of various specifications for the DFM and determined that the best model so far had 2 global factors that follow a VAR process of order 4. Additionally, we identified three group-specific factors for the indicators of economic activity, employment, and expectations, that follow a VAR process with 3 lags. The remaining four groups of variables each have 1 factor that follow AR(1) processes.

B Dataset Description

Variable	Definition	Frequency	Aggregation
PBI_pesca	Fishery GDP	Monthly	-
PBI_mineria	Mining GDP	Monthly	-
elec	Total electricity generation	Monthly	-
elec_sm	Total electricity generation excluding demand from mining companies	Monthly	-
elec_resto	Electricity generation excluding demand from mining and manufacturing companies	Monthly	-
elec_manuf	Electricity demand from manufacturing companies	Monthly	-
cic	Domestic cement consumption	Monthly	-
unacem	UNACEM cement shipments	Monthly	-
afo	Public Construction	Monthly	-
anchoveta	Anchovy landings	Monthly	-
pet	Oil production	Daily	Sum
lgn	Liquefied natural gas production	Daily	Sum
gn	Natural gas production	Daily	Sum
colocac	Baby chicken placements	Monthly	-
ipc_tot	Consumer Price Index	Monthly	-
ipc_sae	CPI without Food and Energy	Monthly	-
ipm	Wholesale Price Index	Monthly	-
ipc_aa	Food CPI	Monthly	-
ipc_comb	Fuel CPI	Monthly	-
ipc_ele	Electricity CPI	Monthly	-
ipc_core	Core CPI	Monthly	-
precio_pollo	Wholesale chicken price	Monthly	-
arroz	Rice supply to wholesale markets	Monthly	-
papa	Potato supply to wholesale markets	Monthly	-
cebolla	Onion supply to wholesale markets	Monthly	-
igv_int_real	Real Domestic VAT	Monthly	-
ir	Real Income Tax	Monthly	-
fbk	Gross Capital Formation	Monthly	-
ingtrib	Real tax revenues	Monthly	-
ingnotrib	Real non-tax revenues	Monthly	-
volexp_trad	Traditional export volume	Monthly	-
volexp_notrad	Non-traditional export volume	Monthly	-
volimp_insum_plast	Import volume of plastic inputs	Monthly	-
volimp_insum_hierro	Import volume of iron	Monthly	-
volimp_insum_text	Import volume of textiles	Monthly	-
volimp_insum_papel	Import volume of paper	Monthly	-
volimp_insum_pquim	Import volume of chemical products	Monthly	-
volimp_insum_qorg	Import volume of organic chemicals	Monthly	-
volimp_bbk	Import volume of capital goods without construction materials	Monthly	-
volimp_cons	Import volume of durable consumer goods	Monthly	-
tdi	Terms of Trade	Monthly	-
desempleo	Unemployment Rate	Monthly	-
peao	Employed Economic Active Population	Monthly	-
expti_12m	BTS: Inflation expectations 12 months ahead	Monthly	-
exppbi_12m	BTS: GDP expectations 12 months ahead	Monthly	-
sitactneg_indice	BTS: Current business situation	Monthly	-
ventasn_indice	BTS: Sales index with respect to the previous month	Monthly	-
producn_indice	BTS: Production index with respect to the previous month	Monthly	-
nivdda_indice	BTS: Demand level with respect to expected	Monthly	-
ordcompran_indice	BTS: Purchase orders	Monthly	-
inv_nd	BTS: Unwanted inventories	Monthly	-
eco3prox_indice	BTS: Economy 3 months ahead	Monthly	-
ecoañoprox_indice	BTS: Economy 12 months ahead	Monthly	-

sec3prox_indice	BTS: Sector 3 months ahead	Monthly	-
secañoprox_indice	BTS: Sector 12 months months ahead	Monthly	-
emp3prox_indice	BTS: Company situation 3 months ahead	Monthly	-
empañoprox_indice	BTS: Company situation 12 months ahead	Monthly	-
dda3prox_indice	BTS: Demand 3 months ahead	Monthly	-
ddaañoprox_indice	BTS: Demand 12 months ahead	Monthly	-
cont3prox_indice	BTS: Hiring 3 months ahead	Monthly	-
contañoprox_indice	BTS: Hiring 12 months ahead	Monthly	-
invr3prox_indice	BTS: Investment 3 months ahead	Monthly	-
invrañoprox_indice	BTS: Investment 12 months ahead	Monthly	-
preinstresfor_indice	BTS: Price of inputs 3 months ahead	Monthly	-
prevtatresfor_indice	BTS: Sales price 3 months ahead	Monthly	-
indicca_p	COS: Consumer confidence - Present	Monthly	-
indicca_f	COS: Consumer confidence - Future	Monthly	-
credito_cons	Private sector credit balance - Consumption	Monthly	-
credito_hipo	Private sector credit balance - Mortgage	Monthly	-
credito_emp	Private sector credit balance - Companies	Monthly	-
circulante	Currency in circulation	Monthly	-
banc_actextlp	Long-Term Net External Assets of banking companies	Monthly	-
banc_liqmn	Liquidity in domestic currency of banking companies	Monthly	-
banc_obligvistamn	Demand deposits in domestic currency of banking companies	Monthly	-
banc_obligahorrmn	Savings deposits in domestic currency of banking companies	Monthly	-
banc_obligplazomn	Fixed-Term Liabilities in domestic currency of banking companies	Monthly	-
banc_liqme	Foreign currency liquidity of banking companies	Monthly	-
emisionprim	Monetary base - End of Period	Monthly	-
credito_mn	Credit in domestic currency (millions S/)	Monthly	-
credito_me	Credit in foreign currency (millions US$)	Monthly	-
banc_creditomn	Credit in domestic currency to the Private Sector of banking companies	Monthly	-
banc_cajamn	Cash in domestic currency of banking companies	Monthly	-
banc_cdbcrpmn	BCRP Certificates of deposit of banking companies	Monthly	-
banc_creditome	Credit in foreign currency to the Private Sector of banking companies	Monthly	-
banc_depbcrpme	Deposits in foreign currency in the BCRP of banking companies	Monthly	-
banc_pasextcp	Short-Term External Liabilities in foreign currency of banking companies	Monthly	-
banc_pasextlp	Long-Term External Liabilities in foreign currency of banking companies	Monthly	-
banc_obligme	Liabilities in foreign currency with the Private Sector of banking companies	Monthly	-
rin	Net International Reserves	Monthly	-
lbtr_mn_tot	Payments through LBTR in domestic currency	Monthly	-
lbtr_mn_cheq	Payments through LBTR in domestic currency with Checks	Monthly	-
lbtr_mn_transf	Payments through LBTR in domestic currency with Credit Transfers	Monthly	-
lbtr_me_tot	Payments through LBTR in foreign currency	Monthly	-
lbtr_me_cheq	Payments through LBTR in foreign currency with Checks	Monthly	-
lbtr_me_transf	Payments through LBTR in foreign currency with Credit Transfers	Monthly	-
tasa_pm	Monetary Policy Reference Rate	Monthly	-
tasa_over	National currency overnight deposit rate	Monthly	-
tasa_bonosper10mn	Peruvian 10-year Government Bond Yield in S/	Monthly	-
tasa_bonosper10me	Peruvian 10-year Government Bond Yield in US$	Monthly	-
tasa_encaje	Reserve ratio	Monthly	-
tasa_amn	Average lending interest rate in domestic currency	Monthly	-
tasa_pmn	Average borrowing interest rate in domestic currency	Monthly	-
tasa_interbanc	Interbank Average Interest Rate in domestic currency	Monthly	-
tasa_ame	Average lending interest rate in foreign currency	Monthly	-
tasa_ipme	Average borrowing interest rate in foreign currency	Monthly	-
tasa_hipomn	Mortgage Loan Interest Rate in domestic currency	Monthly	-
mcdocap_bonos	Bonds - Private sector	Monthly	-
mdocap_bonos_fin	Bonds - Financial Institutions	Monthly	-
mdocap_bonos_nofin	Bonds - Non-Financial Institutions	Monthly	-
mdocap_valpub	Public Sector Securities	Monthly	-

igbvl	General BVL Index	Monthly	-
isbvl	Selective BVL Index	Monthly	-
valorafp	Value of AFP Funds	Monthly	-
tc_us_interbvta	Interbank Exchange Rate	Monthly	-
tc_euro	Euro Exchange Rate	Monthly	-
tc_realbil	Bilateral Real Exchange Rate Index	Monthly	-
tc_realmult	Multilateral Real Exchange Rate Index	Monthly	-
embig_peru	EMBIG Peru	Monthly	-
gt_recesion	Google Trends: recession	Monthly	-
gt_kia	Google Trends: kia	Monthly	-
gt_toyota	Google Trends: toyota	Monthly	-
gt_cine	Google Trends: movies	Monthly	-
gt_restaurantes	Google Trends: restaurants	Monthly	-
gt_creditos	Google Trends: credits	Monthly	-
gt_prestamos	Google Trends: loans	Monthly	-
gt_casas	Google Trends: houses	Monthly	-
gt_departamentos	Google Trends: apartments	Monthly	-
gt_ofertas	Google Trends: offers	Monthly	-
gt_empleo	Google Trends: employment	Monthly	-
gt_trabajo	Google Trends: work	Monthly	-
gt_bloqueos	Google Trends: blockades	Monthly	-
gt_crisis_peru	Google Trends: peru crisis	Monthly	-
gt_quiebra	Google Trends: bankruptcy	Monthly	-
gt_economia	Google Trends: economy	Monthly	-
gt_crisis_economica	Google Trends: economic crisis	Monthly	-
gt_terrenos	Google Trends: land	Monthly	-
gt_inmuebles	Google Trends: real estate	Monthly	-
gt_elecciones	Google Trends: elections	Monthly	-
gt_viajes	Google Trends: travel	Monthly	-
gt_vuelos	Google Trends: flights	Monthly	-
gt_visa	Google Trends: visa	Monthly	-
gt_machu_picchu	Google Trends: machu picchu	Monthly	-
gt_hoteles	Google Trends: hotels	Monthly	-
gt_alojamientos	Google Trends: lodging	Monthly	-
gt_vacaciones	Google Trends: vacations	Monthly	-
gt_bonos	Google Trends: bonds	Monthly	-
gt_cts	Google Trends: cts	Monthly	-
gt_afp	Google Trends: afp	Monthly	-
gt_lluvias	Google Trends: rains	Monthly	-
gt_el_niño	Google Trends: el niño	Monthly	-
gt_sequias	Google Trends: droughts	Monthly	-
gt_heladas	Google Trends: frosts	Monthly	-
gt_huaicos	Google Trends: huaicos	Monthly	-
gt_inflacion	Google Trends: inflation	Monthly	-
gt_delivery	Google Trends: delivery	Monthly	-
gt_pollo_a_la_brasa	Google Trends: pollo a la brasa	Monthly	-
atsm	Sea surface temperature (dummy variable)	Monthly	-
pmi_usa	PMI USA	Monthly	-
pmi_china	PMI China	Monthly	-
sentimiento	Sentiment indicator	Daily	Average
vehiculos_livianos	Number of Light Vehicules sold	Monthly	-
vehiculos_pesados	Number of Heavy Vehicules sold	Monthly	-
vehiculos_menores	Number of Small Vehicules sold	Monthly	-
temp_media	Average national temperature	Daily	Average

C Primary Industry

Table 9: Nowcast of YoY Primary Industry

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	-6,6	4,9	4,2	4,1	-5,7	-17,5	4,1	-1,7	-7,1	-11,4	-7,9	-11,2	0,8	-3,7	2,2
Feb-22	-7,1	-1,4	4,1	0,4	-0,4	-2,1	6,0	3,0	4,4	4,5	5,0	0,8	1,4	2,1	-0,1
Mar-22	-14,9	-1,1	-1,4	-2,6	1,3	-11,5	-1,5	-7,3	-4,6	-9,6	-10,1	-7,0	1,2	-4,5	-12,7
Apr-22	-9,7	-1,0	3,4	1,6	2,3	-8,4	4,2	-1,9	-5,6	-15,4	-2,2	-6,9	-2,3	-2,7	6,2
May-22	-11,4	-12,9	-8,4	-8,7	-8,9	-14,5	-17,9	-13,6	-14,3	-15,6	-13,7	-11,8	-12,1	-12,7	-25,6
Jun-22	5,2	-1,1	-3,2	-1,5	-3,2	4,8	0,9	-0,2	0,4	2,9	1,5	-1,1	-3,7	-0,3	0,4
Jul-22	12,7	14,9	11,9	12,6	20,2	21,1	16,0	17,9	19,0	17,6	20,7	21,9	14,8	17,4	32,8
Aug-22	-2,1	10,8	14,3	11,3	14,9	4,8	5,2	6,6	4,1	4,1	10,6	5,6	14,0	8,9	0,7
Sep-22	-1,1	4,3	7,1	4,9	-3,9	4,2	-2,6	6,9	3,5	8,2	3,9	7,0	4,9	4,0	3,3
Oct-22	2,3	-2,8	-12,0	-7,7	-18,2	15,4	-4,8	-0,7	8,5	6,3	2,9	5,4	-7,8	-1,3	17,1
Nov-22	-1,7	-4,7	-6,9	-4,5	-13,1	-16,6	0,1	-4,6	-8,3	-17,7	-14,8	-14,6	-10,5	-9,7	1,9
Dec-22	4,8	3,4	3,8	5,6	0,5	18,9	-2,8	-2,2	2,6	-1,5	3,5	2,6	4,0	3,2	-9,1
Jan-23	12,7	6,0	2,9	-1,2	10,6	7,9	12,7	7,5	9,4	9,9	9,4	7,0	1,3	7,0	25,3
Feb-23	23,0	14,3	12,5	13,1	16,4	12,1	26,2	16,6	12,8	10,9	14,6	7,6	9,7	13,9	17,5
Mar-23	29,3	46,7	9,2	48,1	22,5	32,9	39,1	35,4	32,2	35,6	28,1	31,1	31,7	32,7	23,1
Apr-23	12,4	17,5	12,0	11,3	15,2	12,3	31,0	28,1	17,3	21,4	18,6	8,5	9,5	16,9	28,5
May-23	-28,1	-15,5	-11,7	-15,1	-17,3	-19,2	-5,4	-4,7	-15,9	-7,8	-11,8	-8,5	-12,5	-12,1	-2,2
Jun-23	-29,9	-23,9	-22,0	-22,5	-24,1	-21,9	-24,4	-22,3	-25,1	-24,5	-23,9	-25,4	-23,9	-23,7	-41,1
Jul-23	-18,1	-21,0	-20,7	-22,0	-20,6	-8,2	-17,9	-23,7	-18,4	-18,0	-17,4	-9,6	-22,5	-18,3	-23,9
Aug-23	16,4	7,9	8,3	7,0	7,5	5,2	-9,2	-10,3	9,5	16,7	6,3	5,9	11,2	5,5	-21,6
Sep-23	8,9	8,0	8,2	8,3	3,2	7,6	11,1	13,0	6,8	4,5	6,5	7,6	11,0	8,0	18,6
Oct-23	9,6	15,8	15,0	13,0	12,3	3,2	7,8	12,5	13,6	13,3	8,2	15,3	13,1	11,9	4,0
Nov-23	10,9	-5,2	-5,8	-5,1	-7,1	3,8	-4,0	-1,4	2,3	0,7	-0,1	4,9	-7,8	-2,0	6,8
Dec-23	-28,1	-18,5	-12,4	-16,8	-12,1	-9,2	-5,9	-6,6	-22,7	-14,4	-10,9	-16,8	-12,2	-13,2	-11,8
Jan-24	-16,0	-24,9	-23,8	-22,9	-25,0	-31,2	-29,5	-27,2	-24,4	-28,1	-27,0	-26,8	-26,4	-26,4	-30,0
Feb-24	-22,7	-10,4	-11,3	-13,7	-10,5	-20,5	-12,8	-13,6	-18,6	-16,7	-12,6	-13,7	-11,4	-13,8	-12,1
Mar-24	-13,6	-15,6	-9,8	-18,4	-11,1	-17,0	-21,2	-19,0	-15,9	-11,6	-15,9	-15,8	-16,9	-15,7	-17,6
Apr-24	30,7	26,3	25,6	20,8	21,8	41,8	1,1	12,2	34,9	40,4	41,1	31,3	18,2	26,3	-6,0
May-24	68,5	64,6	64,5	65,1	74,5	64,7	44,5	67,0	67,6	70,0	67,8	66,2	56,4	64,4	75,9
Jun-24	12,0	38,0	34,7	35,7	34,6	33,2	59,9	48,7	33,2	33,9	25,2	25,3	33,4	36,3	47,4
Jul-24	12,6	5,6	7,9	5,4	7,7	1,2	13,1	10,4	0,0	5,7	10,9	3,1	6,9	6,5	13,1
Aug-24	-1,0	2,1	2,2	-1,1	-0,9	-5,4	-8,8	-5,3	-3,3	-5,3	-3,2	-6,1	1,9	-2,8	-11,0
RMSE		9,410	10,627	9,800	10,142	9,469	15,232	12,256	7,272	8,971	8,189	8,233	10,185	8,574	15,526

D Agriculture

Table 10: Nowcast of YoY Agriculture

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	6,2	7,6	6,3	7,7	7,2	5,5	7,9	7,7	7,4	6,8	7,5	7,9	6,0	7,1	8,5
Feb-22	3,1	3,1	4,3	3,8	5,0	2,6	6,0	5,3	5,6	4,2	6,1	5,2	3,7	4,6	7,9
Mar-22	4,7	-0,7	3,0	2,4	1,4	3,2	3,8	4,1	4,2	4,7	3,9	4,2	1,8	3,0	7,4
Apr-22	7,5	1,5	1,7	3,2	1,9	3,6	4,3	3,2	3,8	2,8	3,8	3,6	1,6	2,9	6,8
May-22	8,2	4,9	5,3	5,0	4,2	7,4	5,8	5,8	5,3	4,9	4,4	6,3	4,7	5,3	5,0
Jun-22	-0,7	-2,0	-0,4	-0,3	-3,0	0,2	-0,5	-0,6	-0,5	1,4	-1,1	-1,1	-0,2	-0,7	0,1
Jul-22	4,8	2,1	2,0	2,3	1,5	7,6	1,8	1,1	1,7	2,2	2,0	0,6	2,5	2,3	2,5
Aug-22	7,4	3,9	5,0	4,0	4,6	6,9	5,6	6,2	5,7	5,6	5,8	4,0	4,1	5,1	6,2
Sep-22	4,3	1,4	0,7	1,0	-0,8	3,0	4,9	2,4	2,6	2,4	2,1	2,3	1,4	1,9	4,1
Oct-22	6,0	6,7	5,9	6,2	5,8	6,0	5,8	5,7	6,2	4,7	5,9	6,4	6,3	6,0	8,8
Nov-22	3,1	7,0	7,2	7,4	9,7	9,9	9,2	7,9	8,7	8,1	8,4	8,2	6,8	8,2	9,6
Dec-22	0,3	-0,2	1,9	1,6	0,2	2,4	2,1	2,1	1,5	0,8	1,6	1,7	2,0	1,5	1,0
Jan-23	3,5	3,5	2,8	2,5	3,0	2,5	3,2	2,4	2,4	2,8	2,3	3,0	3,8	2,8	2,1
Feb-23	0,3	4,6	4,8	4,7	4,1	3,4	4,7	5,3	5,0	5,8	3,4	5,4	4,5	4,6	4,9
Mar-23	0,4	1,6	-1,2	-1,4	1,6	2,2	0,3	0,7	0,1	1,8	1,2	1,2	2,5	0,9	1,0
Apr-23	-11,0	-0,5	-0,8	-0,1	0,5	-0,6	1,1	0,0	-0,2	-0,4	0,7	0,0	0,0	0,0	-1,1
May-23	-4,3	-9,1	-7,6	-7,8	-7,4	-7,4	-7,4	-7,5	-7,6	-8,0	-7,7	-6,8	-7,3	-7,7	-15,0
Jun-23	-2,1	0,5	0,9	-3,6	3,5	-0,1	-2,5	-2,1	-2,5	-2,3	0,3	-1,7	1,6	-0,7	-2,7
Jul-23	-0,2	-2,8	-3,6	-3,5	-2,5	-4,6	-3,6	-2,8	-3,4	-0,6	0,2	-2,2	-3,2	-2,7	-2,2
Aug-23	-2,5	-3,2	-4,0	-3,3	-4,0	1,4	-3,4	-1,6	-2,0	-5,3	-1,7	-1,7	-4,0	-2,7	-0,5
Sep-23	-7,5	-0,4	-1,8	-1,9	-2,3	2,2	-3,9	-1,9	-2,0	0,9	-1,6	1,1	-0,8	-1,0	-0,6
Oct-23	-5,3	-7,8	-6,3	-8,3	-5,6	-2,8	-5,0	-4,8	-4,4	-4,2	-4,7	-5,2	-6,5	-5,5	-3,2
Nov-23	3,2	-4,6	-4,3	-3,2	-6,5	0,8	-3,6	-2,3	-2,9	-2,2	-1,1	-2,8	-5,0	-3,1	-2,0
Dec-23	0,7	5,4	4,7	4,8	4,9	2,1	2,9	4,5	4,7	5,1	4,0	3,7	4,5	4,3	1,5
Jan-24	-2,4	1,4	1,4	-0,5	-0,2	-0,1	0,2	-0,4	0,2	-0,4	-1,3	-1,4	0,3	-0,1	0,5
Feb-24	-0,2	2,3	1,6	0,7	4,1	1,0	2,4	1,3	0,5	-0,5	2,2	1,3	0,9	1,5	2,8
Mar-24	1,2	1,5	1,1	1,9	0,2	-0,1	-0,2	0,8	0,7	4,3	1,9	0,6	1,1	1,1	0,7
Apr-24	24,0	14,0	13,0	13,0	12,9	12,3	12,6	12,5	12,3	12,4	11,0	12,6	12,5	12,6	11,9
May-24	4,8	6,0	5,6	5,7	5,5	10,4	9,6	9,2	9,1	6,3	9,0	10,3	5,9	7,7	7,9
Jun-24	-0,8	2,6	3,6	4,5	3,4	2,8	2,6	3,7	3,4	4,0	4,2	4,9	2,9	3,5	3,5
Jul-24	-3,4	-2,0	-1,8	-1,8	-1,8	-1,8	-1,7	-2,8	-1,7	-3,3	0,6	2,0	-1,8	-1,5	-1,2
Aug-24	-1,8	2,9	1,8	2,0	1,3	0,5	0,2	-0,8	-0,4	-1,2	-1,0	-0,2	-0,4	0,4	-0,9
RMSE		4,277	4,084	4,043	4,567	4,158	4,026	3,937	3,977	4,104	4,146	4,298	4,180	3,978	4,414

E Retail Trade

Table 11: Nowcast of YoY Retail Trade

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	2,3	3,2	3,1	3,2	3,2	3,0	3,0	2,9	3,0	2,9	2,8	3,0	3,2	3,0	3,4
Feb-22	7,5	8,5	8,5	8,2	8,3	8,2	8,4	8,3	8,3	8,2	8,3	8,3	8,5	8,3	9,0
Mar-22	8,1	5,0	5,2	5,2	5,1	5,1	4,9	4,8	4,8	4,9	4,6	4,7	5,3	5,0	6,0
Apr-22	2,6	3,8	3,7	3,7	3,8	3,8	3,9	3,9	3,9	3,9	3,9	3,9	3,8	3,8	3,9
May-22	2,8	2,8	3,0	2,9	3,2	3,2	2,9	3,0	3,1	3,0	3,0	3,1	2,8	3,0	3,6
Jun-22	2,5	2,1	2,0	2,0	1,9	1,8	2,1	2,0	2,1	2,1	2,1	2,1	2,0	2,0	2,8
Jul-22	2,8	1,6	1,7	1,6	1,6	1,8	1,6	1,7	1,7	1,7	1,7	1,7	1,7	1,7	2,3
Aug-22	2,3	2,2	2,1	2,2	2,2	1,9	2,3	2,2	2,2	2,2	2,1	2,1	2,3	2,2	2,2
Sep-22	2,1	2,2	2,3	2,4	1,9	2,2	2,1	2,1	2,1	2,1	2,1	2,1	2,2	2,1	2,4
Oct-22	2,8	2,0	2,0	2,0	2,1	2,0	2,0	2,0	2,1	2,1	2,0	1,9	2,0	2,0	2,5
Nov-22	3,0	2,3	2,3	2,4	2,3	2,5	2,2	2,3	2,3	2,2	2,4	2,3	2,4	2,3	2,4
Dec-22	1,8	1,9	2,2	2,2	2,1	2,2	2,0	2,3	2,1	2,2	2,1	2,3	2,3	2,2	2,3
Jan-23	1,2	2,7	2,9	2,9	2,8	2,9	2,6	2,6	2,6	2,8	2,6	2,7	2,9	2,8	2,6
Feb-23	2,4	1,5	1,6	1,6	1,6	1,7	1,6	1,7	1,7	1,7	1,6	1,6	1,6	1,6	2,0
Mar-23	3,0	1,7	1,7	2,0	2,0	2,1	2,3	2,3	2,1	2,1	2,2	2,2	2,0	2,0	2,3
Apr-23	3,2	2,3	2,2	2,2	2,2	2,4	2,3	2,2	2,3	2,4	2,2	2,2	2,3	2,3	2,4
May-23	3,2	2,9	2,8	2,7	2,5	2,8	2,7	2,6	2,6	2,6	2,8	2,6	2,5	2,7	2,6
Jun-23	3,1	3,0	3,0	3,0	3,0	3,2	3,0	2,9	2,9	3,0	3,1	2,9	3,0	3,0	2,8
Jul-23	3,0	3,1	3,1	3,0	3,1	3,0	3,1	3,0	3,0	2,9	2,9	3,0	3,1	3,0	2,7
Aug-23	2,8	2,8	2,8	2,9	2,9	2,9	2,9	2,9	2,9	2,9	2,8	2,8	2,9	2,9	2,3
Sep-23	1,9	2,7	2,9	2,9	2,7	2,8	2,9	2,8	2,8	2,7	2,8	2,8	2,9	2,8	2,5
Oct-23	1,4	2,4	2,4	2,4	2,3	2,2	2,4	2,4	2,4	2,4	2,3	2,4	2,5	2,4	2,2
Nov-23	1,3	2,0	1,9	2,0	2,0	1,9	2,0	2,0	2,0	1,9	2,0	2,0	1,9	2,0	2,3
Dec-23	2,0	1,2	1,3	1,3	1,4	1,1	1,2	1,3	1,4	1,3	1,4	1,3	1,3	1,3	1,7
Jan-24	2,4	1,9	2,0	2,0	2,0	2,0	2,1	2,0	2,0	2,0	1,8	2,0	2,1	2,0	2,2
Feb-24	3,0	2,3	2,3	2,4	2,3	2,2	2,4	2,3	2,4	2,3	2,3	2,3	2,4	2,3	2,6
Mar-24	1,8	1,9	1,7	1,7	1,8	1,9	2,0	1,9	2,0	2,0	1,8	1,9	1,8	1,9	2,0
Apr-24	3,1	3,6	3,7	3,7	3,6	3,9	3,6	3,7	3,7	3,6	3,7	3,6	3,8	3,7	3,7
May-24	2,1	2,4	2,2	2,4	2,3	2,3	2,3	2,2	2,2	2,3	2,3	2,2	2,4	2,3	2,5
Jun-24	2,3	1,9	1,9	1,9	1,9	2,1	2,0	2,0	2,0	2,0	2,0	2,0	1,9	2,0	2,2
Jul-24	3,4	2,1	2,1	2,2	2,2	2,3	2,1	2,2	2,1	2,3	2,3	2,2	2,1	2,2	2,5
Aug-24	2,9	2,9	2,9	2,9	2,9	2,6	2,9	2,8	2,8	2,9	2,9	2,8	2,9	2,8	2,6
RMSE		0,911	0,893	0,880	0,883	0,871	0,893	0,901	0,894	0,878	0,909	0,912	0,895	0,888	0,768

F Services

Table 12: Nowcast of YoY Services

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	4,0	4,5	4,4	4,4	4,5	4,5	4,6	4,6	4,6	4,5	4,5	4,4	4,5	4,5	4,2
Feb-22	7,0	8,3	8,3	8,4	8,3	7,9	8,3	8,2	8,3	8,1	8,1	8,2	8,3	8,2	8,6
Mar-22	4,5	5,5	5,5	5,4	5,5	5,0	5,2	5,3	5,1	5,2	5,2	5,2	5,5	5,3	5,9
Apr-22	5,2	4,2	4,1	4,1	4,1	3,8	4,1	4,1	4,0	4,0	4,3	4,0	4,1	4,1	4,8
May-22	4,6	4,7	4,5	4,6	4,7	4,6	4,6	4,6	4,7	4,6	4,6	4,7	4,6	4,6	4,5
Jun-22	3,6	3,8	3,7	3,7	3,8	3,8	3,9	3,9	3,9	4,0	3,9	3,8	3,7	3,8	3,9
Jul-22	2,4	3,3	3,3	3,3	3,3	3,2	3,1	3,1	3,2	3,1	3,1	3,0	3,3	3,2	2,8
Aug-22	2,9	2,4	2,3	2,4	2,5	3,0	2,4	2,6	2,7	2,6	2,6	2,5	2,2	2,5	2,6
Sep-22	3,0	2,3	2,4	2,3	2,3	2,1	2,4	2,3	2,4	2,6	2,2	2,2	2,4	2,3	2,3
Oct-22	2,1	3,0	2,8	2,9	2,9	2,9	2,9	2,9	2,8	2,8	2,8	3,0	2,8	2,9	2,6
Nov-22	1,7	2,6	2,4	2,4	2,4	2,4	2,4	2,3	2,5	2,5	2,3	2,5	2,4	2,4	2,0
Dec-22	-0,1	1,5	1,5	1,3	1,5	1,4	1,6	1,5	1,5	1,4	1,5	1,4	1,5	1,5	1,6
Jan-23	-1,2	1,1	1,1	1,1	1,0	1,0	1,5	1,2	1,2	1,3	1,1	1,1	1,1	1,2	0,1
Feb-23	-0,3	-0,8	-0,8	-0,7	-0,8	-0,6	-0,6	-0,7	-0,7	-0,9	-0,6	-0,8	-0,7	-0,7	-1,0
Mar-23	-0,6	-0,1	-0,1	-0,1	-0,2	-0,5	-0,1	-0,2	-0,4	-0,3	-0,2	-0,3	-0,1	-0,2	-0,5
Apr-23	-0,5	-1,4	-1,3	-1,2	-1,3	-0,9	-1,1	-1,1	-1,1	-1,1	-1,1	-1,2	-1,2	-1,2	-1,1
May-23	0,2	0,0	0,1	0,1	0,1	0,4	0,3	0,1	0,0	0,1	0,3	0,1	0,2	0,2	0,0
Jun-23	0,3	-0,1	-0,1	-0,1	-0,2	0,0	0,1	0,1	-0,1	0,0	-0,1	0,1	-0,1	0,0	0,0
Jul-23	-0,6	0,4	0,5	0,5	0,6	0,2	0,3	0,4	0,3	0,5	0,3	0,4	0,5	0,4	0,3
Aug-23	-0,8	-0,6	-0,5	-0,2	-0,6	-0,5	-0,2	-0,5	-0,5	-0,6	-0,3	-0,6	-0,4	-0,5	-0,4
Sep-23	-0,7	-0,9	-0,9	-0,9	-1,0	-0,5	-1,0	-0,9	-0,8	-0,7	-0,8	-0,8	-1,1	-0,9	-0,6
Oct-23	-0,3	0,1	0,1	0,1	0,2	-0,1	0,2	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,4
Nov-23	-1,2	-0,4	-0,5	-0,4	-0,5	-0,2	-0,4	-0,3	-0,3	-0,3	-0,3	-0,3	-0,5	-0,4	0,4
Dec-23	0,8	-1,1	-0,8	-0,7	-0,6	-0,9	-0,9	-0,9	-1,0	-1,1	-0,9	-1,0	-1,1	-0,9	0,2
Jan-24	1,5	1,7	1,6	1,7	1,7	1,8	1,7	1,6	1,6	1,5	1,7	1,7	1,6	1,7	1,1
Feb-24	1,9	1,8	1,6	1,8	1,8	1,4	1,7	1,6	1,4	1,5	1,7	1,4	1,8	1,6	2,0
Mar-24	1,3	0,5	0,5	0,4	0,5	0,7	0,4	0,5	0,5	0,4	0,5	0,4	0,5	0,5	0,5
Apr-24	3,5	3,5	3,1	3,5	3,1	3,2	3,3	3,3	3,4	3,5	3,4	3,3	3,2	3,3	3,5
May-24	2,5	2,1	2,0	2,0	2,4	2,2	2,2	2,2	2,1	2,0	1,9	2,1	2,1	2,1	3,1
Jun-24	2,2	2,4	2,3	2,3	2,3	2,4	2,1	2,0	2,1	2,1	2,0	2,0	2,3	2,2	2,3
Jul-24	4,5	3,4	3,3	3,2	3,5	3,6	3,3	3,4	3,2	3,5	3,4	3,4	3,3	3,4	3,2
Aug-24	3,6	3,3	3,3	3,2	3,2	3,2	3,3	3,3	3,3	3,1	3,3	3,3	3,3	3,3	2,7
RMSE		0,878	0,843	0,842	0,827	0,796	0,869	0,828	0,852	0,850	0,809	0,835	0,864	0,833	0,784

G Non-Primary Industry

Table 13: Nowcast of YoY Non-Primary Industry

	GDP	Lasso	Ridge	Elastic Net	SVR	DT	KNN	RF	AdaBoost	GBoost	XGBoost	Bagging	MLP	Mean ML	DFM
Jan-22	0,1	8,1	6,0	5,5	4,6	5,0	5,0	5,6	6,0	5,5	5,7	6,8	6,1	5,8	5,8
Feb-22	5,8	1,2	1,0	1,3	1,7	3,8	0,7	1,1	2,0	1,2	2,6	0,9	1,3	1,6	3,6
Mar-22	10,4	6,5	4,4	4,1	5,2	6,9	4,9	5,7	5,0	5,7	5,3	5,2	4,8	5,3	5,2
Apr-22	7,7	9,3	8,5	8,4	7,8	10,0	9,6	9,2	9,1	9,6	8,4	10,1	8,5	9,1	7,7
May-22	8,9	9,9	8,5	8,5	8,0	9,1	8,6	9,0	9,7	9,6	9,3	8,8	8,4	8,9	7,9
Jun-22	5,2	4,8	6,2	5,8	5,8	5,9	4,9	6,0	5,4	5,9	6,7	6,2	5,9	5,8	5,5
Jul-22	-1,1	2,5	2,9	2,5	2,8	7,2	4,3	4,1	4,4	4,1	4,1	3,9	3,5	3,9	3,8
Aug-22	1,5	1,0	0,8	0,8	1,2	2,8	0,9	1,3	0,3	0,0	1,5	1,3	0,3	1,0	1,9
Sep-22	1,0	-1,1	-0,6	-1,2	-1,4	-2,3	-1,9	-1,9	-1,7	-0,3	-1,4	-1,1	-1,0	-1,3	-1,4
Oct-22	-0,9	2,9	3,1	2,9	2,5	1,5	2,4	3,2	3,8	3,0	2,7	4,0	2,8	2,9	2,7
Nov-22	-1,6	-2,5	-2,3	-2,0	-2,7	0,0	0,5	-1,0	-1,1	-1,6	-0,5	0,0	-1,9	-1,2	-0,8
Dec-22	-8,4	-3,7	-4,1	-4,1	-3,9	-3,6	-4,2	-4,1	-4,2	-3,1	-3,7	-3,3	-3,8	-3,8	-5,2
Jan-23	-4,2	-3,4	-2,8	-2,8	-2,7	-5,0	-2,9	-2,7	-2,7	-3,7	-3,3	-3,2	-2,6	-3,2	-2,9
Feb-23	-8,8	-7,5	-7,9	-7,2	-7,8	-10,2	-8,2	-8,2	-8,4	-8,0	-8,2	-7,8	-7,8	-8,1	-9,4
Mar-23	-7,2	-11,2	-10,7	-10,5	-10,8	-10,3	-11,5	-10,5	-11,1	-10,7	-11,1	-10,2	-11,4	-10,8	-11,5
Apr-23	-8,3	-11,2	-11,1	-11,0	-9,8	-8,1	-9,4	-9,1	-9,2	-10,4	-9,8	-9,1	-11,1	-9,9	-9,7
May-23	-10,2	-7,9	-7,5	-7,8	-7,7	-7,4	-7,2	-6,9	-7,3	-7,5	-7,8	-7,2	-7,6	-7,5	-6,0
Jun-23	-7,9	-11,3	-10,4	-10,5	-10,7	-12,1	-10,6	-10,8	-11,1	-11,1	-11,2	-10,6	-10,3	-10,9	-9,9
Jul-23	-11,1	-4,6	-4,4	-5,0	-4,5	-5,1	-4,1	-4,6	-3,9	-3,3	-3,5	-4,4	-6,7	-4,5	-4,7
Aug-23	-8,6	-11,5	-11,8	-11,0	-11,1	-10,3	-9,9	-10,1	-10,1	-9,6	-10,5	-10,0	-11,7	-10,6	-9,6
Sep-23	-12,9	-14,6	-14,5	-14,4	-15,2	-11,8	-12,5	-11,7	-11,8	-13,1	-11,9	-12,0	-14,5	-13,2	-10,9
Oct-23	-7,6	-7,7	-8,3	-8,2	-7,6	-7,3	-7,5	-9,4	-9,6	-9,8	-9,3	-9,7	-7,8	-8,5	-8,1
Nov-23	-4,4	-8,0	-8,0	-8,4	-8,5	-9,1	-8,2	-8,4	-8,8	-9,0	-9,4	-8,5	-8,3	-8,6	-8,6
Dec-23	-3,7	-4,5	-3,6	-3,1	-3,3	-1,8	-3,7	-3,2	-3,2	-3,8	-2,6	-2,5	-4,0	-3,3	-4,5
Jan-24	0,5	-0,3	-0,5	-0,4	-0,1	-0,5	-0,6	-1,1	-0,6	0,0	-1,1	-1,0	-0,5	-0,6	-0,8
Feb-24	3,1	1,1	0,5	1,0	2,0	0,4	1,7	1,3	1,0	0,8	1,6	1,9	1,9	1,3	1,7
Mar-24	-9,1	-7,9	-7,8	-7,6	-7,5	-7,1	-7,2	-6,8	-8,0	-6,4	-6,4	-6,2	-7,9	-7,2	-7,2
Apr-24	5,5	6,0	5,5	5,9	6,2	2,1	4,4	3,1	3,7	4,2	4,8	4,4	5,5	4,6	3,5
May-24	0,8	1,7	1,2	1,3	0,9	4,8	2,1	2,0	1,0	1,3	1,7	1,8	1,7	1,8	2,5
Jun-24	-4,1	-2,4	-2,8	-2,8	-2,7	-3,4	-3,9	-3,6	-4,3	-4,2	-4,0	-3,7	-2,1	-3,3	-3,6
Jul-24	10,3	6,7	6,9	6,9	6,8	6,8	7,5	6,5	5,8	6,4	6,4	6,7	6,6	6,7	6,8
Aug-24	4,2	3,7	1,7	1,7	1,3	2,7	3,9	3,6	3,1	4,1	3,4	2,1	2,5	2,8	3,8
RMSE		3,027	2,985	2,878	2,795	3,164	2,940	2,960	3,107	3,088	3,037	3,132	2,873	2,903	2,822

References

Bańbura, Marta, Domenico Giannone, Michele Modugno, and Lucrezia Reichlin. 2013. "Now-casting and the real-time data flow." In *Handbook of economic forecasting.* Vol. 2, 195–237. Elsevier.

Bolivar, Osmar. 2024. "GDP nowcasting: A machine learning and remote sensing data-based approach for Bolivia." *Latin American Journal of Central Banking,* 5(3): 100126.

Bravo Higuera, Diego Fernando, León Darío Parra Bernal, Milenka Linneth Argote Cusi, and Grace Andrea Torres Pineda. 2024. "Colombian Agricultural Sector's Early Estimator of Gross Domestic Production Using Nowcasting and Big Data Methods." *Journal of technology management & innovation,* 19(2): 54–66.

Breiman, Leo. 2001. "Random forests." *Machine learning,* 45: 5–32.

Breiman, L, JH Friedman, R Olshen, and CJ Stone. 1984. "Classification and Regression Trees."

Buell, Brandon, Reda Cherif, Carissa Chen, Karl Walentin, Jiawen Tang, and Nils Wendt. 2021. *Impact of COVID-19: Nowcasting and big data to track economic activity in Sub-Saharan Africa.* International Monetary Fund.

Chen, Tianqi, and Carlos Guestrin. 2016. "Xgboost: A scalable tree boosting system." 785–794.

Cortes, Corinna, and Vladimir Vapnik. 1995. "Support-Vector Networks." *Machine Learning.*

Coulombe, Philippe Goulet, Maxime Leroux, Dalibor Stevanovic, and Stéphane Surprenant. 2021. "Macroeconomic data transformations matter." *International Journal of Forecasting,* 37(4): 1338–1354.

Cover, Thomas, and Peter Hart. 1967. "Nearest neighbor pattern classification." *IEEE transactions on information theory*, 13(1): 21–27.

Dauphin, Mr Jean-Francois, Mr Kamil Dybczak, Morgan Maneely, Marzie Taheri Sanjani, Mrs Nujin Suphaphiphat, Yifei Wang, and Hanqi Zhang. 2022. *Nowcasting gdp-a scalable approach using dfm, machine learning and novel data, applied to european economies.* International Monetary Fund.

De Oliveira, Lucas Gabriel Martins. 2023. "Which One Predicts Better?: Comparing Different GDP Nowcasting Methods Using Brazilian Data."

Efron, Bradley, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. 2004. "Least angle regression."

Fix, Evelyn, and Joseph L Hodges. 1951. "Discriminatory analysis." *Nonparametric discrimination: Small sample performance. Report A*, 193008.

Fornaro, Paolo, and Henri Luomaranta. 2020. "Nowcasting Finnish real economic activity: a machine learning approach." *Empirical Economics*, 58(1): 55–71.

Freund, Yoav, and Robert E Schapire. 1997. "A decision-theoretic generalization of on-line learning and an application to boosting." *Journal of computer and system sciences*, 55(1): 119–139.

Fulton, Chad. 2020. "Large dynamic factor models, forecasting, and nowcasting."

Ghosh, Saurabh, and Abhishek Ranjan. 2023. "A machine learning approach to GDP nowcasting: An emerging market experience." *Bulletin of Monetary Economics and Banking*, 26: 33–54.

Giannone, Domenico, Lucrezia Reichlin, and David Small. 2008. "Nowcasting: The real-time informational content of macroeconomic data." *Journal of monetary economics*, 55(4): 665–676.

Goulet Coulombe, Philippe, Maxime Leroux, Dalibor Stevanovic, and Stéphane Surprenant. 2022. "How is machine learning useful for macroeconomic forecasting?" *Journal of Applied Econometrics*, 37(5): 920–964.

Hopp, Daniel. 2024. "Benchmarking econometric and machine learning methodologies in nowcasting GDP." *Empirical Economics*, 66(5): 2191–2247.

Jolliffe, Ian T. 2002. *Principal component analysis for special types of data.* Springer.

Kant, Dennis, Andreas Pick, and Jasper de Winter. 2022. "Nowcasting GDP using machine learning methods."

Lenza, Michele, and Giorgio E Primiceri. 2022. "How to estimate a vector autoregression after March 2020." *Journal of Applied Econometrics*, 37(4): 688–699.

León, Carlos, and Fabio Ortega. 2018. "Nowcasting economic activity with electronic payments data: A predictive modeling approach." *Revista de economia del Rosario*, 21(2): 381–407.

Mariano, Roberto S, and Yasutomo Murasawa. 2010. "A coincident index, common factors, and monthly real GDP." *Oxford Bulletin of economics and statistics*, 72(1): 27–46.

Miranda, Hairo Ulises. 2021. "Nowcasting Mexican economic activity by using deep learning approaches: A comparison with econometric models."

Muchisha, Nadya Dwi, Novian Tamara, Andriansyah Andriansyah, and Agus M Soleh. 2021. "Nowcasting Indonesia's GDP growth using machine learning algorithms." *Indonesian Journal of Statistics and Its Applications*, 5(2): 355–368.

Ng, Serena. 2013. "Variable selection in predictive regressions." *Handbook of economic forecasting*, 2: 752–789.

Richardson, Adam, Thomas van Florenstein Mulder, and Tuğrul Vehbi. 2021. "Nowcasting GDP using machine-learning algorithms: A real-time assessment." *International Journal of Forecasting*, 37(2): 941–948.

Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams. 1986. "Learning representations by back-propagating errors." *nature*, 323(6088): 533–536.

Schorfheide, Frank, and Dongho Song. 2021. "Real-time forecasting with a (standard) mixed-frequency VAR during a pandemic." National Bureau of Economic Research.

Soybilgen, Barış, and Ege Yazgan. 2021. "Nowcasting us gdp using tree-based ensemble models and dynamic factors." *Computational Economics*, 57(1): 387–417.

Tenorio, Juan, and Wilder Perez. 2023. "GDP nowcasting with machine learning and unstructured data to Peru."

Tiffin, Mr Andrew. 2016. *Seeing in the dark: A machine-learning approach to nowcasting in Lebanon.* International Monetary Fund.

Watanabe, Shuhei. 2023. "Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance." *arXiv preprint arXiv:2304.11127*.

Zhang, Qin, He Ni, and Hao Xu. 2023. "Nowcasting Chinese GDP in a data-rich environment: Lessons from machine learning algorithms." *Economic Modelling*, 122: 106204.