# Classical versus Machine-Learning Forecasting in a High-Dimensional, Noisy Environment: A Monte Carlo Study

Pedro José Piqueras Martínez*

September 20, 2026

**Abstract**

Can machine-learning models use broader indicator panels to improve quarterly GDP nowcasts relative to a dynamic factor model (DFM)? Before the pandemic, DFMs were central to short-term forecasting, but the arrival of machine learning to macroeconomic nowcasting changed that consensus. Recent studies report that flexible learners can beat the classics. Most studies use autoregressions as benchmarks, while those that include a DFM often evaluate it on specific samples. This makes it unclear whether the reported gains stem from the forecasting method, the benchmark's specification, or the data. We run the experiment the literature has skipped, in a controlled laboratory we sweep the one dimension every empirical comparison holds fixed by accident — whether the classical benchmark is correctly specified — and compare 6 models in two Monte Carlo environments, with 100 replications per design, and in a US GDP nowcasting exercise. Each family competes as its own literature prescribes: the Machine-learning models receive the full high-dimensional panel, while the DFM is estimated on a curated 15 indicator panel, the scale that the factor-model literature identifies as sufficient. There is no uniform winner: the ranking is a property of the environment rather than of the estimators. But the balance tilts towards the classics. When the DFM matches the generating process, none of its 5 competitors achieves a lower mean relative RMSE. Under a neural-generated process, 5 do, although the gains are small relative to Monte Carlo variation. In US data, the DFM ranks first on average and at the nowcast horizon, with no statistically significant gain for any competitor. Model rankings therefore depend on the forecasting environment.

**Keywords:** Forecasting, Dynamic Factor Models, mixed-frequency data, Machine Learning, Deep Learning, Monte Carlo simulation.

**JEL codes:** C32, C45, C53, E37.

*Universidad de Alicante.

# Contents

# 1 Introduction

Just before the pandemic, short-term GDP forecasting had something close to a consensus. Dynamic factor models (DFMs) had become a workhorse of nowcasting, and central banks, international organisations and private investors had adopted them as core infrastructure (Giannone et al., 2008; Bańbura and Modugno, 2014). The practical debate had shifted toward the model's scale: how many factors to retain and which indicators to include. Research on factor dimension, targeted panels, and release timing made those choices more disciplined showing that a compact, curated panel was not a compromise but the efficient choice (J. Bai and Ng, 2002; Boivin and Ng, 2006; J. Bai and Ng, 2008; Bańbura and Rünstler, 2011; Alvarez et al., 2016). The profession had, in other words, a structured guide: a model class, a criterion for choosing its size, and a rough consensus on what that size should be.

The pandemic drew attention to the difficulty of forecasting abrupt changes. At the same time, machine-learning methods offered more flexible ways to use large indicator panels at the cost of interpretability. Studies report gains over traditional models in GDP nowcasting (Richardson et al., 2021), global trade nowcasting (Hopp, 2022), and inflation forecasting (Medeiros et al., 2021). However, interpreting those gains is harder than the comparison may suggest. Empirical comparisons in the literature give a mixed picture because they vary along too many axes at once: countries, samples, target definitions, indicator coverage, release calendars, and treatment of outlier episodes. Most comparisons use an autoregression as a benchmark. Because it cannot use the monthly indicator panel, outperforming it does not establish an advantage over a DFM. Where a DFM is included, a result from one country and period may depend on that sample, its release calendar, or how the DFM was specified. Dauphin et al. (2022) find that DFMs tend to perform better in normal periods, while several machine-learning methods do well at turning points. In simulations and six country applications, Klakow Akepanidtaworn and Korkrid Akepanidtaworn (2025) find that traditional methods lead in most cases, although simpler machine-learning methods sometimes win. The relevant question is whether machine-learning models using a broader indicator panels deliver reliable accuracy gains over a carefully specified, curated-panel DFM, and how do those gains vary across forecasting environments.

We address this question by comparing the models with predictor sets motivated by their respective literatures. The machine-learning models receive the full high-dimensional panel, while the DFM is estimated on a curated 15-indicator panel. This choice matters because indicators differ in predictive value, noise, and release timing: a broader panel may contain useful signals, but it may also bring in weak or noisy series. We keep these panel assignments fixed across the analysis. To assess how the ranking depends on the forecasting environment, we evaluate the models under two data-generating processes using the same protocol. The first is a linear factor economy with 100 monthly indicators. Here the DFM matches the generating structure, and we vary the level and distribution of measurement noise across indicators. The second process is estimated from a real macroeconomic panel using a neural network that handles missing observations. We simulate from that fitted process with bootstrapped shocks; none of the competing forecast models is correctly specified for it. The two environments differ in more than the DFM's specification status, so their comparison tests whether a ranking is robust across

different settings rather than isolating a single causal effect.

We compare 6 models—2 classical and 4 machine-learning models—at horizons $h \in \{0, 1, 2, 3, 4\}$, with 100 replications per simulation design. The machine-learning models receive the full indicator panel, while the DFM uses a selected 15-indicator subset. This reflects how each approach is intended to work: the learners can use breadth, and the DFM benefits from indicator selection. We also evaluate the models at 116 US forecast origins. That exercise uses publication lags measured from historical vintages, although the indicator values themselves are revised.

The results do not support a uniform ranking. In the factor economy, none of the 5 competitors has a lower mean relative RMSE than the DFM. In the neural-generated economy, 5 do, but their gains are small relative to Monte Carlo variation. In the US exercise, the DFM ranks first on average and at the nowcast horizon, and no competitor achieves a statistically significant gain. The simulation reversal shows why a result from one generating process cannot settle a comparison of methods. For an institution with an established DFM, these findings call for reliable gains against a well-specified benchmark before replacing its forecasting system.

Section 2 reviews the literature. Section 3 describes the two generating processes and the US data exercise; Sections 4 and 5 present the models and shared evaluation protocol. Sections 6, 7, and 8 report and interpret the results.

## 2 Related Literature

Nowcasting uses information released before the target variable is published. For quarterly GDP, that means working with monthly indicators that arrive at different times and leave the current quarter only partly observed. A useful model must handle both the difference in frequency and this changing pattern of missing observations. Giannone et al. (2008) show how individual releases update a GDP nowcast; Bańbura et al. (2013) and Bok et al. (2018) describe the wider real-time data problem. Bridge equations, MIDAS regressions, mixed-frequency VARs, and factor models address these problems in different ways (Camacho et al., 2013).

Dynamic factor models (DFMs) summarise many indicators with a small number of common factors (Stock and Watson, 2002; Forni et al., 2000). J. Bai and Ng (2002) provide criteria for choosing how many factors to retain. In a state-space DFM, the Kalman filter can update the estimated factors as new observations arrive, even when the latest months are incomplete (Doz et al., 2011; Bańbura and Modugno, 2014). Panel size and composition affect how well those factors are estimated. Adding series may provide more measurements of the common signal, but Boivin and Ng (2006) show that forecasts can deteriorate when idiosyncratic errors are strongly correlated across series. J. Bai and Ng (2008) find gains from estimating factors with predictors selected for the target series. In simulations and US data, Alvarez et al. (2016) find settings in which a small panel representing distinct economic categories matches or outperforms a larger, more disaggregated one. Bańbura and Rünstler (2011) show that the value of timely survey and financial indicators for GDP forecasts depends on their publication lags. The Euro-STING and Spain-STING applications illustrate how a small group of timely indicators can support operational GDP nowcasts (Camacho and Pérez-Quirós, 2010; Camacho and Pérez-Quirós, 2011). Together, these results motivate selecting indicators for the DFM.

Machine-learning methods offer different ways to use a broad indicator set while dealing with its high dimensionality. The elastic net regularizes a linear regression by shrinking coefficients and setting some effectively to zero, which stabilizes estimation when predictors are numerous and correlated and performs variable selection (Zou and Hastie, 2005). Random forests accommodate nonlinearities, interactions, and threshold effects through recursive partitioning, while random subsampling of predictors at each split reduces reliance on any single subset of a high-dimensional feature space (Breiman, 2001; Athey et al., 2019). Recurrent neural networks process predictors sequentially and compress past information into a hidden state, whereas temporal convolutional networks use shared convolutional filters over local time windows, allowing both architectures to extract lower-dimensional temporal representations from many predictors and lags (Hochreiter and Schmidhuber, 1997; S. Bai et al., 2018). These methods may therefore capture relationships missed by a linear factor model, although estimating their additional flexibility reliably from a short quarterly GDP series remains difficult.

The source of any machine-learning gain matters as much as the model label. In macroeconomic forecasting experiments, Goulet Coulombe et al. (2022) separate the effects of nonlinearity, regularization, cross-validation, and loss functions; they find nonlinearity especially important while the standard factor model remains an effective form of regularization. Kock and Teräsvirta (2014) compare automated methods for fitting autoregressive neural networks to monthly in-

dustrial production and unemployment during the 2007–09 crisis. They find no method that dominates across the series, reinforcing the need to evaluate flexible methods across forecasting conditions rather than infer an advantage from model capacity.

The empirical evidence tends to give the machine learning a consistent advantage. However, it varies among many dimensions including countries, samples, target definitions, indicator sets, etc. Using real-time New Zealand data, Richardson et al. (2021) find that several machine-learning methods improve on both an autoregression and a DFM for GDP nowcasting. Other positive results concern different targets: Hopp (2022) finds gains from LSTMs over DFMs for three global trade measures, while Medeiros et al. (2021) study inflation forecasting. In European GDP applications, Dauphin et al. (2022) find that DFMs tend to do better in normal periods, whereas several machine-learning methods identify turning points well. The target, sample, and evaluation period therefore matter when interpreting a reported gain.

The closest broad comparison to our work is Klakow Akepanidtaworn and Korkrid Akepanidtaworn (2025). They compare traditional and machine-learning methods in simulations with linear, quadratic, and interaction terms and in GDP nowcasts for six countries. Their simulated indicators retain a common factor structure. Traditional methods, especially bridge models and DFMs, lead in most of the country cases. Lasso and elastic net are generally the strongest machine-learning methods and sometimes outperform the traditional models; more complex nonlinear methods often fare worse in their samples. The authors link that pattern to the limited length of quarterly GDP series and note that their conclusions depend on the countries, indicators, and simulation design. Their paper therefore does not support a general claim that machine learning outperforms classical GDP nowcasting methods.

Comparisons also depend on what information each model receives. A study using historical data vintages measures a different forecasting exercise from one that reconstructs release dates but uses revised indicator values (Camacho and Pérez-Quirós, 2010; Richardson et al., 2021). Our study evaluates a curated-panel DFM and full-panel machine-learning models at the same targets, origins, and horizons within each environment. It repeats that comparison in a factor-generated economy, a neural-generated economy, and a US data exercise. Because the predictor sets differ by design, the results compare these complete forecasting strategies, not the estimators in isolation.

# 3 Data-Generating Processes

## 3.1 Why two data-generating processes

A Monte Carlo comparison of forecasting models is only as informative as the process that generates its data, and any single process encodes a strong assumption about which competitor is favoured. Write the target as

$$y_\tau \;=\; m(\mathcal{I}_\tau) + \epsilon_\tau, \qquad m(\mathcal{I}_\tau) \;=\; \mathbb{E}[y_\tau \mid \mathcal{I}_\tau]\,, \tag{1}$$

with $\mathcal{I}_\tau$ the information available at the forecast origin. By construction,

$$\mathbb{E}[\epsilon_\tau \mid \mathcal{I}_\tau] = 0, \tag{2}$$

so that $m(\mathcal{I}_\tau)$ is the optimal forecast under squared-error loss.

Define the population mean-squared-error risk of a forecasting model $f$ as

$$R(f) = \mathbb{E}\Big[(y_\tau - f(\mathcal{I}_\tau))^2\Big]\,. \tag{3}$$

Substituting (1) gives

$$\begin{aligned} R(f) &= \mathbb{E}\Big[(m(\mathcal{I}_\tau) - f(\mathcal{I}_\tau) + \epsilon_\tau)^2\Big] \\ &= \mathbb{E}\Big[\epsilon_\tau^2\Big] + \mathbb{E}\Big[(m(\mathcal{I}_\tau) - f(\mathcal{I}_\tau))^2\Big] \\ &\quad + 2\mathbb{E}[\epsilon_\tau\,(m(\mathcal{I}_\tau) - f(\mathcal{I}_\tau))]\,. \end{aligned} \tag{4}$$

The cross term vanishes by iterated expectations hence,

$$R(f) = \underbrace{\mathbb{E}\Big[\epsilon_\tau^2\Big]}_{\text{irreducible risk}} + \underbrace{\mathbb{E}\Big[(m(\mathcal{I}_\tau) - f(\mathcal{I}_\tau))^2\Big]}_{\text{error from using } f}\,. \tag{5}$$

Thus, a model's attainable loss contains an irreducible component determined by the innovation $\epsilon_\tau$ and a model-dependent component measuring the distance between its forecasting rule and the conditional mean. Choosing a DGP fixes both $m$ and the distribution of $\mathcal{I}_\tau$, and therefore fixes the approximation error of every competitor before a single replication is drawn. If $m$ lies inside the hypothesis class of one model, that model can attain zero approximation error and the experiment measures primarily estimation efficiency; if $m$ lies far outside all competing classes, the experiment instead places greater weight on their relative flexibility. Neither reading generalises on its own.

This paper therefore runs the same models, the same origins and the same protocol against two processes placed deliberately at the two ends of that decomposition.

DGP-A is the correctly specified environment (Section 3.2). It is a parametric economy in which one latent AR(2) factor drives 100 monthly indicators and latent monthly GDP, and the quarterly target is its Mariano–Murasawa aggregate. Here $m$ is linear, Gaussian and exactly of

the form the Dynamic Factor Model estimates, so the DFM's approximation error is zero by construction. This makes the benchmark as strong as possible and tests whether flexible models can *match* it when there is no additional structure to extract. Because $m$ is known, the design also admits an exact stress axis — the per-indicator noise-to-signal ratio — which no real panel provides.

DGP-B is the misspecified environment (Section 3.3). It is *estimated* from a real mixed-frequency macro panel: a missing-aware recurrent network is fitted to the one-step conditional mean of 100 FRED monthly indicators plus quarterly GDP, then iterated recursively with shocks resampled from its own estimated residuals. Here $m$ is a trained network on the full panel, so no competitor contains it and every competitor carries a nonzero approximation error. DGP-B replaces imposed comovement and persistence with patterns estimated from data, replaces Gaussian innovations with resampled residuals, and removes the benchmark's specification advantage.

Table 1 states the two designs side by side, with the real economy of Section 3.4 in a third column for reference. Both processes are simulated at the same sample length, model specification and panel width and evaluated under the identical protocol of Section 5, so the comparison between them is a comparison of environments only.

**Table 1:** The two data-generating processes, and the real economy they approximate.

| | **DGP-A** | **DGP-B** | **Experiment C** |
|---|---|---|---|
| Conditional mean $m$ | parametric, linear | estimated network | unknown |
| Origin of parameters | fixed by design | fitted to real data | not chosen |
| Comovement | one imposed factor | whatever the panel has | whatever the economy has |
| Innovations | Gaussian | bootstrapped residuals | realised history |
| DFM approximation error | zero | nonzero | unknown |
| Stress axis | measurement noise | none (single cell) | none |
| Information set | full-information M3 | full-information M3 | measured ragged edge |
| Monthly periods | 600 | 600 | 504 |
| Quarters | 200 | 200 | 168 |
| Monthly indicators | 100 | 100 | 100 |
| Replications | 100 | 100 | 1 (structural) |
| Forecast origins | 1 per replication | 1 per replication | 116 |

### 3.2 DGP-A: a parametric measurement-error economy

The simulation generates a panel of 100 monthly indicators over 600 months (200 quarters) per replication, driven by a single latent factor. The parameters are fixed across replications, and measurement error is the only axis swept across designs. The construction follows the standard factor representation that is the empirical target of the DFM literature (Stock and Watson, 2002; Forni et al., 2000; Stock and Watson, 2016) and the dimension that machine-learning models must learn implicitly in order to match the canonical benchmark.

In DGP-A this predictor-set comparison is controlled exactly. The machine-learning models receive all 100 contaminated indicators, including the increasingly noisy end of the heterogeneous panel, while the DFM receives the 15 indicators with the lowest $\tau_i$ values. Sweeping $\tau_{\max}$ therefore

tests whether broad-panel learning and regularisation can realise the information advantage of additional available series while screening or down-weighting many low-signal variables, against a classical benchmark that is both correctly specified and supplied with the cleanest available subset.

### 3.2.1 Latent factor and indicator equation

Let $f_t$ denote a univariate latent factor that follows an AR(2) process,

$$f_t \;=\; \phi_1 f_{t-1} + \phi_2 f_{t-2} + u_t, \qquad u_t \sim \mathcal{N}(0,1), \tag{6}$$

with $(\phi_1, \phi_2) = (0.6, 0.2)$. The $n = 100$ monthly indicators load on the factor and carry an idiosyncratic AR(2) component,

$$x^*_{i,t} \;=\; \lambda_i f_t + e_{i,t}, \tag{7}$$

$$e_{i,t} \;=\; \psi_1 e_{i,t-1} + \psi_2 e_{i,t-2} + \varepsilon_{i,t}, \qquad \varepsilon_{i,t} \sim \mathcal{N}(0, \sigma^2_{\varepsilon,i}), \tag{8}$$

with $(\psi_1, \psi_2) = (0.4, 0.1)$. The starred symbol $x^*_{i,t}$ denotes the *latent signal* indicator; the observed indicator $x_{i,t}$ that enters every model is the contaminated version constructed in Section 3.2.3 below. Indicators are cross-sectionally independent given $f_t$, so the common component generates a rank-one cross-sectional covariance:

$$\mathrm{Cov}(x^*_{i,t}, x^*_{j,t}) \;=\; \lambda_i \lambda_j \,\mathrm{Var}(f_t) \qquad (i \neq j), \qquad \mathrm{Var}(\boldsymbol{\lambda} f_t) \;=\; \mathrm{Var}(f_t)\, \boldsymbol{\lambda}\boldsymbol{\lambda}', \tag{9}$$

with $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)'$.

### 3.2.2 Mixed-frequency aggregation

Latent monthly GDP growth $g^*_t$ is generated from the same common factor,

$$g^*_t \;=\; \delta f_t + e_{y,t}, \tag{10}$$

with $\delta = 1.5$ and $e_{y,t}$ an independent AR(2) idiosyncratic component that deliberately reuses the indicator idiosyncratic autoregressive parameters of Section 3.2.1: $e_{y,t} = \psi_1 e_{y,t-1} + \psi_2 e_{y,t-2} + \varepsilon_{y,t}$ with $\varepsilon_{y,t} \sim \mathcal{N}(0,\, 0.3^2)$. The factor loading $\delta$, the innovation standard deviation 0.3, and the $\psi$ are fixed constants of the generator; together with the unit factor-innovation scale $\sigma_f = 1$ they imply a factor share of the latent monthly GDP variance of $\delta^2\gamma_f(0) \,/\, (\delta^2\gamma_f(0) + \gamma_{e_y}(0)) \approx 0.98$. The observed quarterly target $y^Q_\tau$ is the Mariano and Murasawa (2003) log-linear aggregate of $g^*_{t_\tau}, \ldots, g^*_{t_\tau - 4}$ within quarter $\tau$, where $t_\tau = 3\tau$ is the last month of the quarter:

$$y^Q_\tau \;=\; \tfrac{1}{3} g^*_{t_\tau} + \tfrac{2}{3} g^*_{t_\tau - 1} + g^*_{t_\tau - 2} + \tfrac{2}{3} g^*_{t_\tau - 3} + \tfrac{1}{3} g^*_{t_\tau - 4}. \tag{11}$$

The triangular weights $(1/3, 2/3, 1, 2/3, 1/3)$ arise from the log-linear approximation $x^Q_t \approx (x_t + x_{t-1} + x_{t-2})/3$ for the quarterly log level and a first-difference of the resulting quarterly

series.

Every AR(2) process in the design is initialized by drawing its two initial lags from the exact stationary distribution derived in the supplementary annex, Section 1.1. The latent monthly GDP series is simulated with two pre-sample months so that the first quarter's aggregation window in (11) is complete.

### 3.2.3 Variance-preserving noise injection

The DGP is exposed to additive Gaussian measurement noise that is *rescaled to preserve the marginal variance of each indicator*. Let $\sigma^2_{x_i^*} = \mathrm{Var}(x^*_{i,t})$. The observed indicator is

$$\boxed{x_{i,t} \;=\; \frac{x^*_{i,t} + \tau_i\, \sigma_{x_i^*}\, \eta_{i,t}}{\sqrt{1+\tau_i^2}}, \qquad \eta_{i,t} \sim \mathcal{N}(0,1),\ \eta_{i,t} \perp x^*_{i,t}.} \tag{12}$$

By construction $\mathrm{Var}(x_{i,t}) = \sigma^2_{x_i^*}$ and the noise-to-signal ratio is

$$\mathrm{NSR}_i \;=\; \frac{\mathrm{Var(noise)}}{\mathrm{Var(signal)}} \;=\; \tau_i^2. \tag{13}$$

The derivation of (12)–(13) is given in the supplementary annex, Section 1.2. This rescaling is the central identifying device of the design: at every level of the parameter $\tau_{\max}$, the marginal variances of all indicators are identical to those of the noise-free panel, so a model cannot distinguish noisy from clean indicators by inspecting unconditional variances. What changes across the sweep is the fraction of each indicator's variance that carries the common signal, not the total scale.

Evaluating across the noise grid traces the level and the heterogeneity of measurement error simultaneously and identifies the regime in which each mechanism dominates, while holding fixed the dimensions (sample size, factor structure, aggregation, panel size) that would otherwise confound the comparison.

A direct consequence is that measurement noise attenuates the effective factor loading, which is the economic mechanism through which noise hurts every model:

$$\lambda_i^{\mathrm{eff}} \;=\; \frac{\lambda_i}{\sqrt{1+\tau_i^2}}, \qquad \mathrm{Corr}(x_{i,t}, f_t) \;=\; \frac{\mathrm{Corr}(x^*_{i,t}, f_t)}{\sqrt{1+\tau_i^2}}. \tag{14}$$

As $\tau_i \to \infty$, $\lambda_i^{\mathrm{eff}} \to 0$ and the indicator becomes pure noise.

### 3.2.4 Measurement-error designs

The specification in (12) is exposed to the heterogeneous-noise design which has 3 levels (0.1, 0.5, 2.0). Each indicator $i$ is assigned a distinct $\tau_i$ linearly spaced from zero to a ceiling $\tau_{\max}$, with $\tau_{\max}$ swept across the grid. Indicator 0 is perfectly clean ($\tau_0 = 0$), whereas indicator $n-1$ is the noisiest ($\tau_{n-1} = \tau_{\max}$). The design therefore varies both the dispersion and the level of

signal quality across the panel. The $\tau_i$ assignment is preserved across replications, so the ranking of indicators by signal quality is a fixed, known object rather than a per-replication draw.

#### 3.2.5 GDP moments and persistence

The 100 Monte Carlo replications generate the following quarterly GDP properties. Distributional moments pool the 200 quarters in each replication, while ACF(1) is computed within each replication and then averaged.

**Table 2:** Quarterly GDP moments in DGP-A.

| Property | Generated DGP-A |
|---|---|
| Mean quarterly growth | 0.05124 |
| Standard deviation | 6.03195 |
| ACF(1) | 0.723 |
| 5th percentile | -9.92568 |
| Median | 0.04761 |
| 95th percentile | 9.99121 |

These GDP moments are common to all three measurement-error designs because the noise parameter $\tau_i$ affects the indicators but not the latent GDP process in (10).

### 3.3 DGP-B: an estimated neural economy

DGP-B replaces the parametric process of Section 3.2 with one whose conditional mean is estimated from data. It is a *data-generating process*, not a forecasting model. The network below never produces a forecast that is scored anywhere in this paper; it manufactures the synthetic economies on which the 6 forecasting models of Section 4 are evaluated.

The same predictor-set contrast is retained. The machine-learning models use all 100 simulated monthly indicators, while the DFM uses its fixed 15-series broad-macro panel. Here noise is inherited from the empirical panel rather than indexed by a known $\tau_i$. DGP-B thus asks whether learners can exploit the wider information set — their data-availability advantage — well enough to beat the curated classical benchmark once the DFM no longer contains the conditional mean by construction.

#### 3.3.1 What DGP-B adds

Relative to DGP-A, DGP-B no longer imposes a factor structure. DGP-A generates comovement from one latent AR(2) factor with a rank-one cross-sectional covariance (9); DGP-B instead inherits the cross-sectional and dynamic dependence of the real panel, including departures from low rank and linearity. The DFM is consequently an approximation rather than the truth, as is every other competitor.

The dynamics are estimated rather than chosen. Persistence in DGP-A is the AR(2) pair $(0.4, 0.1)$ applied identically to every series, whereas in DGP-B each column's persistence is learned (Section 3.3.4). The innovations are also resampled rather than Gaussian: DGP-B draws one-step residual vectors from the estimated model's own residual pool, preserving skewness, excess kurtosis and contemporaneous cross-sectional dependence. These changes remove the benchmark's specification advantage, so rankings that survive the move from DGP-A to DGP-B cannot be attributed to the DFM being the true model.

Because $m$ is estimated, DGP-B has no ground-truth parameter analogous to $\tau_{\max}$ that can be swept while leaving the remaining design fixed. DGP-A therefore supplies the controlled stress test, while DGP-B supplies an estimated counterfactual in which the benchmark is misspecified.

### 3.3.2 Source panel and estimation sample

The generator is estimated on a monthly panel drawn from FRED in the FRED-MD tradition (McCracken and Ng, 2016): 100 monthly indicators plus quarterly real GDP growth, all expressed in stationary transformed space, over 1985-01-01–2019-12-01, with the network's training block ending at 2017-12-01. The sample stops before 2020 deliberately: in scaled space the pandemic quarters reach several dozen standard deviations. The sample also empties the residual pool of pandemic draws. The 2001 and 2008–09 recessions remain, so the estimated economy still contains downturns.

### 3.3.3 Conditional mean

Let $\boldsymbol{z}_t \in \mathbb{R}^K$, $K = 100 + 1$, collect the transformed target and indicators in scaled space, let $\boldsymbol{o}_t \in \{0,1\}^K$ be the observation mask (the target is observed only at quarter-end months), and let $\boldsymbol{z}_t^{\text{last}}$ carry the most recent observed value of each column. The generator is an LSTM network $f_\theta$ with hidden width 8 reading a rolling window of 12 months,

$$\mathbb{E}[\boldsymbol{z}_t \mid \mathcal{F}_{t-1}] \;=\; \boldsymbol{a} \odot \boldsymbol{z}_{t-1}^{\text{last}} \;+\; g_\theta(\boldsymbol{z}_{t-\ell},\, \boldsymbol{o}_{t-\ell},\, \boldsymbol{\Delta}_{t-\ell})_{\ell=1}^{12}\,, \tag{15}$$

where $\boldsymbol{\Delta}_t$ records the elapsed time since each column was last observed and $\odot$ is the elementwise product. Two features of (15) are derived in the annex, Section 2. First, the term $\boldsymbol{a} \odot \boldsymbol{z}_{t-1}^{\text{last}}$, with $a_k$ constrained to $|a_k| < 1$ and initialised at each column's empirical AR(1) coefficient: is what keeps the iterated recursion from degenerating into a random walk in every column. Second, the learnable-decay imputation of Che et al. (2018) applied to the columns that are ever missing, which is how a quarterly target and monthly indicators share one network without ad hoc interpolation of the target.

The criterion is masked mean squared error, so unobserved cells never enter the loss and the minimiser remains the conditional mean. The fitted network carries 4580 parameters and is trained for 25 epochs, a stopping point calibrated on expanding folds within its estimation sample.

### 3.3.4 Recursion and shocks

Each replication is generated by iterating (15) forward from a seed window of real observations,

$$\boldsymbol{z}_t \;=\; \mathbb{E}[\boldsymbol{z}_t \mid \mathcal{F}_{t-1}] \;+\; s\,\boldsymbol{\varepsilon}_t, \qquad \boldsymbol{\varepsilon}_t \;\sim\; \text{bootstrap draw from } \widehat{\mathcal{E}}, \tag{16}$$

where $\widehat{\mathcal{E}}$ is the set of one-step residual vectors of the fitted model on the estimation sample and $s = 1.00$ scales them. Resampling whole vectors rather than drawing independent scalars preserves the contemporaneous cross-sectional dependence of the residuals (Efron, 1979). The target row is masked back to a quarterly calendar as the recursion proceeds, so the simulated panel has exactly the mixed-frequency missingness pattern of the real one.

Every replication is seeded from the same real-data tail, so the first 240 months are discarded before 600 months (200 quarters) are kept. Without that burn-in the replications would be conditional paths from one initial condition rather than draws from the process's stationary distribution.

### 3.3.5 GDP moments and persistence

The 100 Monte Carlo replications from DGP-B give the following quarterly GDP properties. The distributional moments pool the 200 quarters in each replication, while the generated ACF(1) is computed within each replication and then averaged.

**Table 3:** Quarterly GDP moments in DGP-B and its source data.

| Property | Generated DGP-B | Source real GDP |
|---|---|---|
| Mean quarterly growth | 0.00581 | 0.00662 |
| Standard deviation | 0.00447 | 0.00564 |
| ACF(1) | 0.392 | 0.363 |
| 5th percentile | -0.00141 | -0.00351 |
| Median | 0.00590 | 0.00682 |
| 95th percentile | 0.01281 | 0.01535 |

The column *Source real GDP* describes the same GDP that is the target of the real data experiment.

## 3.4 Experiment C: The Real Economy

The two Monte Carlo experiments differ in the benchmark's specification status and agree on everything else. Neither is the economy. This section runs the same 6 models, the same M3 nowcast protocol and the same 100-indicator panel against US data, so that the simulation findings can be read against the environment they are meant to describe.

### 3.4.1 What this experiment can and cannot do

Experiment C is not a third data-generating process. DGP-A and DGP-B *generate* economies, and repeating the draw is what produces their sampling distributions; here there is one realisation of an unknown process and all sampling variation lives in the 116 forecast origins.

Accuracy levels are not comparable across experiments. In the Monte Carlo, accuracy is averaged over replications at a single origin with a constant estimation sample, and dispersion is simulation noise. Here it is averaged over origins along one historical path with an estimation sample that grows from 48 to 164 quarters, and dispersion is sampling variation in time. Relative RMSE against the benchmark, and the ordering it implies, transfer between experiments; the levels do not.

With only 116 forecast origins, Experiment C has limited power to distinguish small differences in accuracy. Let $\delta$ denote a competitor's fractional reduction in RMSE relative to the benchmark, and let $\rho$ denote the correlation between their forecast errors at the same origin. Under a simple approximation with mean-zero Gaussian forecast errors and independent origins, the Diebold–Mariano behaves approximately as $\sqrt{N}\,\delta/\sqrt{1-\rho^2}$. With 116 origins, an RMSE reduction of about 5.7% when $\rho = 0.95$, or 7.9% when $\rho = 0.90$, would bring that ratio to the two-sided 5% rejection threshold of 1.96. The roughly four-percent gaps between models in the real data experiment are smaller still. At positive horizons, dependence between adjacent loss differences generally raises the standard error, making the calculation optimistic. Experiment C places those results in the context of the observed US economy, but provides limited evidence about small differences between models.

### 3.4.2 Design

The target is quarterly growth in US real GDP. The predictor panel is the 100 monthly FRED indicators on which DGP-B was estimated, so the machine-learning models face an identical predictor set in the estimated and the real economy, and the benchmark is fitted on the same fixed 15-series real-activity block. Full provenance, transformations and the publication calendar are in Appendix 3; three choices matter enough to state here.

**Sample.** 1978Q1–2019Q4, that is 168 quarters. The start is the earliest month at which all 100 indicators exist rather than a modelling preference. The sample ends before the pandemic: under squared-error loss a single observation of the magnitude of 2020Q2 would dominate every ratio reported here, so that each would effectively be a statement about one quarter.

**Origins.** Models are re-estimated at every origin on an expanding window, from 1990Q1 to 2018Q4. This is 116 complete re-estimations of every model, not a single fit evaluated repeatedly.

**Information set.** The ragged edge is *measured*, not assumed. FRED's observation endpoint carries no release metadata, so we read historical vintages from ALFRED and recover, for each indicator, how many of the current quarter's months a forecaster would actually have held at the

origin. The result is 6 series available through the third month, 90 through the second, and 4 only through the first. No member of the benchmark's panel belongs to the first group, so its most recent month is unobserved at every origin.

# 4 Forecasting Models

This section gives, for each of the 6 models, its central equation, information set, estimation principle, and experimental specification. The full derivations are in the supplementary annex, where each model section also explains what its hyperparameters control, why the reported values were chosen, and what changing them would do. The comparison covers an autoregression, a dynamic factor model, a random forest, a multilayer perceptron, a temporal convolutional network, and a long short-term memory network.

Two conventions apply throughout. All models are estimated under the fixed scheme and forecast the same quarterly target at the same origins (Section 5); and all multi-horizon machine-learning models use the *direct* protocol, fitting one model per horizon rather than iterating a one-step rule forward.

## 4.1 Classical benchmarks

**AR with BIC lag selection.** A univariate autoregression on quarterly GDP,

$$y_\tau^Q \;=\; c + \sum_{j=1}^{p} \rho_j \, y_{\tau-j}^Q + \varepsilon_\tau, \qquad \varepsilon_\tau \sim \mathcal{N}(0, \sigma^2), \tag{17}$$

with $p$ selected by the Bayesian information criterion (Schwarz, 1978) over $p \in \{1, \dots, p_{\max}\}$ on each estimation window and multi-step forecasts iterated from (17). It is the floor of the comparison: a model that cannot beat it is extracting nothing from the indicator panel.

*Experimental specification.* The largest admissible order is $p = 4$. BIC selects the order, ordinary least squares estimates an intercept and the lag coefficients, and forecasts beyond one quarter are obtained by iterating the fitted equation. The four-quarter ceiling gives BIC one complete year of GDP history while limiting coefficient variance in short windows; BIC then retains fewer lags whenever the additional year-history terms do not justify their penalty. *(Annex, Section 4.)*

**Dynamic Factor Model — the benchmark.** A mixed-frequency dynamic factor model cast in linear state-space form,

$$\boldsymbol{y}_t \;=\; H\,\boldsymbol{h}_t + \boldsymbol{w}_t, \qquad \boldsymbol{h}_t \;=\; F\,\boldsymbol{h}_{t-1} + \boldsymbol{v}_t, \tag{18}$$

with $n_f = 1$ latent factor(s) and independent AR idiosyncratic components stacked in $\boldsymbol{h}_t$. The row of $H$ that corresponds to GDP implements the Mariano–Murasawa weights (11) on the latent monthly activity state, mapping monthly latent activity into the quarterly observable. Estimation is by Kalman-filter maximum likelihood; missing observations are handled by row selection in the Kalman update (Bańbura and Modugno, 2014; Durbin and Koopman, 2012). **The DFM is the benchmark** against which every other model's RMSE and MAE are reported as a ratio.

*Experimental specification.* Factor dynamics AR(2); idiosyncratic dynamics AR(2); quarterly

GDP is treated as a flow and linked to monthly activity by the five-term Mariano–Murasawa rule. Estimation maximises the Kalman-filter likelihood by L-BFGS-B, with at most 100 iterations and tolerance 0.0010. loading-sign normalisation and stationarity restrictions are imposed. The DFM is fitted on a fixed panel of 15 indicators rather than the full 100: under DGP-A these are the 15 lowest-$\tau_i$ series, hence the cleanest indicators, and under DGP-B an explicit broad-macro panel, selected by those variables whose estimation of GDP by the common part offers higher correlation with the actual GDP. Holding the benchmark's input set fixed and favourable is deliberate. The resulting information-set asymmetry is part of the estimand: machine-learning models have the information advantage of the full panel, whereas the DFM has the advantages of prior selection and parsimony. The comparison asks whether the flexible learners can exploit their broader data availability while regularising low-signal additions strongly enough to outperform the selected DFM. One factor and AR(2) component dynamics match DGP-A's rank and persistence while keeping the same parsimonious benchmark under DGP-B. *(Annex, Section 5.)*

### 4.2 Machine learning

**Random Forest.** A bagged ensemble of regression trees with random feature subsetting (Breiman, 2001). Each tree is a piecewise-constant estimator on a recursive partition $\{R_m\}_{m=1}^{M}$ of the predictor space,

$$\widehat{f}_T(\boldsymbol{x}) \;=\; \sum_{m=1}^{M} c_m \, \mathbf{1}\{\boldsymbol{x} \in R_m\}, \tag{19}$$

where $c_m$ is the mean training response among observations assigned to terminal region $R_m$. The ensemble forecast is the unweighted mean of $B$ bootstrap-resampled trees. Athey et al. (2019) formalize the econometric properties of the modern variant; the broader role of trees in applied econometrics is reviewed by Athey and Imbens (2019) and Mullainathan and Spiess (2017).

*Experimental specification.* $B = 100$ trees; maximum depth 10; the square root of the available feature count is considered at each split; minimum 3 samples per leaf and 2 per split; bootstrap resampling on. The feature row at each origin stacks AR(4) lags of the quarterly target, 3 monthly lags of every indicator, and rolling means, standard deviations and counts over windows $\{3, 6\}$ months, together with availability masks; the target is quarter $\tau(t) + h$, so one supervised problem is built per horizon. Training uses quarter-end M3 forecast origins and requires at least 20 rows. Missing predictors are handled within the training sample before the forest is fitted.

One hundred trees supplies repeated bootstrap averaging without making the Monte Carlo design depend on a very large ensemble. Depth 10 permits nonlinear interactions, while the three-observation leaf minimum prevents the deepest branches from becoming isolated forecasts. Four quarterly target lags cover a year, three monthly indicator lags cover a quarter, and the 3- and 6-month rolling windows add quarterly and half-year summaries to a model with no intrinsic ordering in time.

The forest uses the standard observation-level bootstrap: each tree samples forecast-origin rows rather than contiguous time blocks. Lagged information inside each feature vector is preserved, although serial dependence between sampled rows is not. *(Annex, Section 6.)*

### 4.3 Deep learning for unbalanced panels

3 neural architectures are included, sharing a common unbalanced-panel input layer that encodes each indicator together with a binary mask, so the missing values created by mixed frequencies are handled without ad hoc imputation. Each architecture is a composition of layers of the form

$$\boldsymbol{a}^{(\ell)} \;=\; g^{(\ell)}\!\big(W^{(\ell)}\boldsymbol{a}^{(\ell-1)} + \boldsymbol{b}^{(\ell)}\big), \qquad \boldsymbol{a}^{(0)} = \boldsymbol{x}, \tag{20}$$

with $g^{(\ell)}$ a nonlinear activation and $\theta = \{W^{(\ell)}, \boldsymbol{b}^{(\ell)}\}$ trained by minimizing horizon-specific mean squared error with the Adam optimizer (Kingma and Ba, 2015; Goodfellow et al., 2016), early stopping on a held-out validation slice, and a fixed random-seed schedule that absorbs initialization noise into the Monte Carlo variance. After fold-local target standardization, the network output is transformed directly back to GDP units and reported as the forecast. The architectures differ in their inductive bias.

**MLP.** A feedforward network: static nonlinearity through one or more dense hidden layers applied to the flattened window. It imposes no temporal structure, so time enters only through the ordering of the flattened input and the explicit target lags. It is the reference case — the other architectures are read as departures from it.

*Experimental specification.* The first hidden-layer width is $u \in \{8, 16, 24\}$ and depth is $d \in \{1, 2, 3\}$. A depth-$d$ candidate uses the funnel $(u, u/2, \ldots, u/2^{d-1})$, with ReLU activation and one separately estimated network per forecast horizon. The funnels decrease monotonically: widening after a narrow first layer would expand information that layer had already discarded. The widths 8, 16, and 24 span increasing capacity without allowing the first dense layer to dominate the small forecasting sample; depths 1–3 compare a single nonlinear transformation with progressively more hierarchical interactions. ReLU is used to avoid the repeated saturation of tanh during backpropagation; shared dropout and weight decay regularise the associated weights, while gradient clipping protects the update from an excessive norm. norm.

**TCNN.** A temporal convolutional network with causal and dilated 1D convolutions, which read local temporal patterns with a parameter count independent of the window length (S. Bai et al., 2018; Borovykh et al., 2019).

*Experimental specification.* Filters $\{4, 8, 16\}$, kernel size 3, convolutional depth $\{2, 3\}$, dense head 8, and one separately estimated network per forecast horizon. At $k_s = 3$ the admitted depths read 7 and 15 of the 12 window months (the 3 excess land on the left causal zero-padding). Depth one is excluded on purpose: it reads only 3 months and is also the cheapest candidate in the grid. The filter grid {4,8,16} moves from a strongly regularised representation to one that can recognise more temporal motifs. Kernel size 3 corresponds to one quarter of monthly observations; depths 2 and 3 then provide receptive fields of 7 and 15 months, so the grid compares partial-year with full-window reach. The eight-unit dense head combines those filters without replacing the convolutional bottleneck with a large unrestricted layer.

**LSTM.** A gated recurrent network with long-range memory and learned input, output and forget gates (Hochreiter and Schmidhuber, 1997). The additive cell-state update is what keeps gradients from contracting geometrically across the window.

*Experimental specification.* Hidden widths $\{4, 6, 8\}$ and depths $\{1, 2, 3\}$, with one separately estimated network per forecast horizon. The admitted widths are smaller than the MLP's layer sizes because the four gates and the recurrent term make each unit far more expensive. Widths 4, 6, and 8 therefore span increasing memory capacity while remaining small enough for the sample, and depths 1–3 test whether additional recurrent abstraction helps before the four-gate parameter blocks become dominant.

*Shared experimental specification.* Input window 12 months plus 4 quarterly target lags; the 100-column panel is compressed by a learned projection of dimension 2 before the temporal architecture; the masked equal-weight factor is supplied as an additional input channel. Optimisation uses learning rate 0.0010, weight decay 0.0010, dropout 0.1000, gradient-norm clipping at 1, minibatches of 16 target-quarter groups, at most 200 epochs and early-stopping patience 20 on a 0.1500 validation split. Training rows, candidate comparisons, and epoch selection use all three within-quarter months, while forecast-variance calibration is scored separately on the third, which is the actual forecast. Hyperparameter search exhaustively enumerates each architecture's size-by-depth grid under a 5-fold expanding inner-validation scheme within the fixed outer training block, with the shared optimum selected at $h = 0$ and reused at the others; a separate 3-fold loop over the terminal 0.4000 of the sample calibrates the stopping epoch. Section 5.6 gives the search grids in full.

*(Annex, Section 7.)*

# 5 Experimental Design

Both experiments run under one protocol. Only the sweep axis differs. DGP-A admits an exact measurement-noise axis because its truth is known, and DGP-B admits none (Section 3.3.1).

## 5.1 Replications, origins and windows

Each design cell is evaluated over 100 independent Monte Carlo replications. The reported experiment uses the fixed estimation scheme: within each replication, the split is placed at quarter $\lfloor n_q \times 0.90 \rfloor$, the first 0.90 of the quarterly sample is used for estimation, and one set of forecasts is produced from that split for all reported horizons. All 6 models see the same simulated panel at the same single origin, so every cross-model comparison within a replication is paired, which removes simulation noise from the rankings.

The origin is a quarter-end (M3) vintage: current-quarter monthly indicators are released and current-quarter GDP is still withheld.

## 5.2 Horizons

Forecasts are evaluated at $h \in \{0, 1, 2, 3, 4\}$ quarters, computed from the shared fit at each estimation window. Horizon $h = 0$ is the *nowcast*: the current quarter, whose GDP has not yet been published but whose indicators partly have. It is the horizon where the information sets of the competing models differ most — the DFM and the neural models can read the current quarter's indicator releases, while the univariate AR cannot. Horizons $h = 1$ to $h = 4$ progressively remove that advantage as the target moves beyond the indicators' reach.

## 5.3 Sweep axes

**DGP-A.** The stress experiment sweeps the ceiling parameter $\tau_{\max}$ of the heterogeneous-noise design through the grid $0.1, 0.5, 2.0$ (3 levels), under the variance-preserving rescaling (12). Because that rescaling enforces $\mathrm{Var}(x_{i,t}) = \mathrm{Var}(x^*_{i,t})$, the only quantity that changes across cells is the per-indicator noise-to-signal ratio $\mathrm{NSR}_i = \tau_i^2$ (Eq. 13); marginal variances of all indicators are held fixed at their noise-free values. Any cross-cell change in forecast accuracy is therefore attributable to the noise channel, not to a change in the scale of the predictors.

**DGP-B.** No sweep. The estimated economy has no ground-truth noise parameter to vary, so DGP-B contributes one design cell per model and horizon.

## 5.4 Evaluation metrics

Fix a model $m$, a horizon $h$, and a design cell $c$ (a noise level under DGP-A; the single cell under DGP-B). For replication $r = 1, \ldots, R$ and forecast origin $o = 1, \ldots, O_r$ within that replication, let $y^{(r)}_{t_o+h}$ denote the target realisation at horizon $h$ ahead of origin $t_o$, and let $\hat{y}^{(r,o)}_{m,h,c}$ denote the

corresponding point forecast. The paired forecast error is:

$$e^{(r,o)}_{m,h,c} \equiv y^{(r)}_{t_o+h} - \hat{y}^{(r,o)}_{m,h,c}, \tag{21}$$

and the per-cell sample size is $N_{h,c} = \sum_{r=1}^{R} O_r$. Under the reported fixed scheme, $O_r = 1$ for every replication. All metrics below are functionals of the $\{e^{(r,o)}_{m,h,c}\}$ collection within the cell $(m, h, c)$; because every model is scored on the same simulated panel at each $(r, o)$, the cross-model comparisons are paired.

For each cell $(m, h, c)$ we report

$$\mathrm{RMSE}_{m,h,c} = \sqrt{\tfrac{1}{N_{h,c}} \sum_{r,o} \big(e^{(r,o)}_{m,h,c}\big)^2}, \tag{22}$$

$$\mathrm{MAE}_{m,h,c} = \tfrac{1}{N_{h,c}} \sum_{r,o} |e^{(r,o)}_{m,h,c}|, \tag{23}$$

$$\mathrm{Bias}_{m,h,c} = \tfrac{1}{N_{h,c}} \sum_{r,o} e^{(r,o)}_{m,h,c}, \tag{24}$$

$$\mathrm{MAPE}_{m,h,c} = \frac{100}{|\mathcal{S}_{h,c}|} \sum_{(r,o)\in\mathcal{S}_{h,c}} \left| \frac{e^{(r,o)}_{m,h,c}}{y^{(r)}_{t_o+h}} \right|, \quad \mathcal{S}_{h,c} = \{(r,o) : y^{(r)}_{t_o+h} \neq 0\}. \tag{25}$$

All means are computed taking into account missing values, so origins for which a model fails to produce a forecast are excluded from the denominator. Intuitively, RMSE penalises large errors quadratically and is the natural loss under a Gaussian target; MAE is robust to tails and preserves the units of $y$; mean bias diagnoses systematic over- or under-prediction; and MAPE is scale-free but undefined whenever $y^{(r)}_{t_o+h} = 0$, which is why it is restricted to the mask $\mathcal{S}_{h,c}$.

The headline metric is RMSE relative to the DFM benchmark,

$$\mathrm{relRMSE}_{m,h,c} = \frac{\mathrm{RMSE}_{m,h,c}}{\mathrm{RMSE}_{\mathrm{DFM},h,c}}, \tag{26}$$

and analogously

$$\mathrm{relMAE}_{m,h,c} = \frac{\mathrm{MAE}_{m,h,c}}{\mathrm{MAE}_{\mathrm{DFM},h,c}}. \tag{27}$$

A value below one means model $m$ improves on the DFM under the corresponding loss. Reporting both is informative because RMSE and MAE need not agree when the error distribution is heavy-tailed: a model that controls extreme errors but is slightly worse on average will look better under RMSE than under MAE.

Ratios are always formed *inside* a design cell and only then averaged across cells:

$$\overline{\mathrm{relRMSE}}_{m,h} = \frac{1}{|\mathcal{C}|} \sum_{c\in\mathcal{C}} \frac{\mathrm{RMSE}_{m,h,c}}{\mathrm{RMSE}_{\mathrm{DFM},h,c}}. \tag{28}$$

The alternative — pooling errors across cells and dividing the pooled RMSEs — weights each cell by its error magnitude, so under DGP-A the noisiest cell would silently dominate the average. Reported tables therefore never pool across horizons: the per-horizon table (Section 6.1) and the noise sweep (Section 6.2) are the two reporting views, and the nowcast is always visible as its

own column.

## 5.5 The real-data protocol

Experiment C (Section 3.4) runs the same models, horizons and M3 origin convention, but its protocol differs in two respects that change how its numbers may be read. Both follow from there being one economy rather than many replications, and both are properties of the design rather than findings.

The first difference is that forecast origins replace replications. The simulated experiments evaluate 100 independent replications at a single origin, so every model faces a fresh economy of constant length and the dispersion across cells is simulation noise. Experiment C evaluates one economy at 116 successive origins on an expanding window, so the estimation sample grows from 48 to 164 quarters and the dispersion is sampling variation along a single historical path. The number of model fits is comparable — one re-estimation per origin — but they are not independent draws.

Consequently, *levels* of RMSE do not transfer between the experiments: relative RMSE against the benchmark and the implied ranking are the comparable objects, and the tables are read that way throughout. The accuracy reported for Experiment C also mixes small-sample and large-sample performance, because the earliest origins are estimated on a quarter of the data available at the last.

The second difference concerns the information set, which is measured rather than assumed. Both simulated experiments give every model all three months of the current quarter for every indicator — a full-information vintage. On real data that vintage does not exist, so Experiment C masks each indicator's trailing months according to a publication lag measured from historical vintages (supplementary annex, Section 3.5): 6 series are available through the third month, 90 through the second, and 4 only through the first. The benchmark's real-activity panel contains none of the first group, so its most recent month is unobserved at every origin.

Because there is only one realisation, all uncertainty comes from the origin dimension. At horizons above the nowcast, adjacent multi-step forecast errors generally share future target innovations, inducing serial dependence. Experiment C is therefore reported with Diebold–Mariano tests using a Newey–West long-run variance and the Harvey–Leybourne–Newbold small-sample correction. The details are given in Appendix 3.7. The power available from 116 origins is stated in Section 3.4.1.

## 5.6 Experimental settings

Section 4 states each model's numerical specification beside its central equation. This subsection collects the generator settings, simulation protocol, and common neural-model selection rules. The statistical role and selected value of each tuning parameter are discussed directly under the corresponding model in the annex.

**Generator settings.** DGP-A contains $T = 600$ months and $N = 100$ indicators. Its factor coefficients are $\phi = (0.6, 0.2)$, its idiosyncratic coefficients are $\psi = (0.4, 0.1)$. No additional publication lag is imposed in DGP-A. DGP-B retains $T = 600$ months after a 240-month burn-in. Its conditional mean is an LSTM with hidden width 8 and a 12-month input window. Dropout is 0.2000, residual vectors enter at scale 1, and the output contains a damped own-lag term and learned decay imputation. The estimation period runs from 1985-01-01 to 2019-12-01, with training ending in 2017-12-01. Optimisation uses batches of 32, learning rate 0.0010, weight decay 0.0001, validation share 0.1500, at most 200 epochs, and patience 20. The selected stopping point is 25 epochs. The quarterly target receives share 0.5000 of the weighted loss, subject to a per-column cap of 20.

**Simulation protocol.** The design uses 100 replications and evaluates $h \in \{0, 1, 2, 3, 4\}$. Models are estimated once on the first 0.90 of the quarterly sample and produce the reported horizon forecasts from a single quarter-end origin; with $n_q$ quarters this places the origin at quarter $\lfloor n_q \cdot 0.90 \rfloor$.

**Neural architecture grids.** All neural architectures share the optimisation settings stated in Section 4. Their search grids vary width and depth, hence model capacity, while dropout, learning rate, and weight decay remain fixed. This restriction matters because candidates inside a one-standard-error band are tied by parameter count: varying a quantity that leaves the count unchanged would otherwise let ordering, rather than evidence, determine the winner. Forecasts are direct, with one network estimated for each horizon.

**Selection and calibration.** Within the fixed outer training block, candidate and epoch selection use 5 expanding folds. Stopping-epoch and forecast-variance calibration use 3 folds. If the candidate losses are $L_1, \ldots, L_K$, the one-standard-error band has half-width $s_L/\sqrt{K}$. Expanding folds have unequal training lengths, so part of $s_L$ is learning-curve dispersion rather than sampling noise. Too few folds would make the band excessively wide and reduce the selection rule to the smallest parameter count. The separate calibration step pools its terminal block, so its precision does not depend on how that block is partitioned. These expanding folds are internal tuning splits; they do not create additional Monte Carlo forecast origins or alter the fixed outer evaluation scheme.

# 6 Results

All tables, figures, and numerical statements use the same cell-level aggregation rule defined in Section 5. Results are reported separately for the two data-generating processes because the comparison between them is itself the central finding.

## 6.1 DGP-A: performance by horizon

Table 4 reports relative RMSE by forecast horizon under the parametric measurement-error economy, averaged across the 3 noise levels within each horizon.

**Table 4:** Relative RMSE (vs. DFM) by forecast horizon (DGP-A).

| Model | $h = 0$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
|---|---|---|---|---|---|
| AR | 4.119 | 1.836 | 1.010 | 1.003 | 1.001 |
| DFM | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| RF | 1.333 | 1.036 | 0.984 | 1.000 | 1.034 |
| MLP | 1.631 | 1.229 | 1.019 | 1.043 | 1.066 |
| TCNN | 1.596 | 1.319 | 1.002 | 1.046 | 1.100 |
| LSTM | 1.927 | 1.385 | 1.016 | 1.066 | 1.071 |

*Note:* Each cell is the mean of per-design-cell relative RMSEs at that horizon: per-cell ratios to the DFM baseline are computed within each noise level (and any other design axis present) and then averaged across them.

The benchmark is not beaten: of the 5 competing models, 0 improve on the DFM's mean relative RMSE. The horizon profile is where the mechanism shows. At the nowcast ($h = 0$) the gap is at its widest — the best non-DFM model attains 1.333, and the univariate AR, which cannot read the current quarter's indicators at all, sits at 4.119 — and it collapses as the horizon extends: by $h = 4$ the AR is at 1.001 and the machine-learning models at 1.034 (RF), 1.066 (MLP), 1.100 (TCNN) and 1.071 (LSTM).

As $h$ grows, the current-quarter indicator information that the DFM exploits through its Kalman update decays out of the forecast, every model converges toward the target's unconditional mean, and the ratios approach one from above. The DFM's advantage is concentrated exactly where information about the current quarter is available and exploitable.

## 6.2 DGP-A: robustness to measurement error

The measurement-error sweep is the stress test of the parametric design. Table 5 reports performance as $\tau_{\max}$ is swept from 0.10 to 2.00 in the heterogeneous design, at the nowcast horizon.

The striking feature is how little the ranking moves. The benchmark's own absolute RMSE rises only from 0.966 at $\tau_{\max} = 0.10$ to 0.990 at $\tau_{\max} = 2.00$ — a factor of 1.024 across a twentyfold increase in the noise ceiling — and the competitors' ratios shift correspondingly little (the random forest from 1.256 to 1.417, the LSTM from 1.951 to 1.943).

**Table 5:** Forecasting performance across the noise sweep (DGP-A).

| Noise level | Model | RMSE | RMSE/DFM | MAE/DFM |
|---|---|---|---|---|
| 0.10 | AR | 3.999 | 4.139 | 3.951 |
| 0.10 | DFM | 0.966 | 1.000 | 1.000 |
| 0.10 | RF | 1.214 | 1.256 | 1.212 |
| 0.10 | MLP | 1.589 | 1.644 | 1.554 |
| 0.10 | TCNN | 1.553 | 1.608 | 1.534 |
| 0.10 | LSTM | 1.885 | 1.951 | 1.874 |
| 0.50 | AR | 3.999 | 4.177 | 3.993 |
| 0.50 | DFM | 0.957 | 1.000 | 1.000 |
| 0.50 | RF | 1.268 | 1.324 | 1.289 |
| 0.50 | MLP | 1.565 | 1.635 | 1.678 |
| 0.50 | TCNN | 1.437 | 1.501 | 1.495 |
| 0.50 | LSTM | 1.807 | 1.888 | 1.818 |
| 2.00 | AR | 3.999 | 4.040 | 3.776 |
| 2.00 | DFM | 0.990 | 1.000 | 1.000 |
| 2.00 | RF | 1.403 | 1.417 | 1.297 |
| 2.00 | MLP | 1.598 | 1.615 | 1.561 |
| 2.00 | TCNN | 1.661 | 1.678 | 1.599 |
| 2.00 | LSTM | 1.923 | 1.943 | 1.829 |

*Note:* Results are for horizon $h = 0$. Noise level is the active design's noise-to-signal parameter ($\tau_{\max}$ for heterogeneous, $\tau$ for uniform). RMSE/DFM and MAE/DFM are ratios to the DFM baseline at the same noise level; values below 1 beat the DFM.

Two design features explain the flatness. First, the heterogeneous design spaces $\tau_i$ linearly from zero to $\tau_{\max}$, so raising the ceiling degrades the noisiest indicators while leaving the cleanest ones untouched; a filter that reweights indicators by their innovation variance simply shifts weight down the ranking. Second, the DFM here is fitted on the 15 cleanest series (Section 4), which are precisely those least affected as the ceiling rises. The sweep therefore stresses the panel far more than it stresses the benchmark — itself a result worth stating, since it says the DFM's robustness in this design comes from *variable selection*, not from tolerating noise per se.

### 6.3 DGP-B: performance by horizon

Table 6 repeats the per-horizon comparison on the estimated neural economy. There is no sweep here (Section 5.3), so the horizon profile is the whole result.

**Table 6:** Relative RMSE (vs. DFM) by forecast horizon (DGP-B).

| Model | $h = 0$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
|---|---|---|---|---|---|
| AR | 1.008 | 0.950 | 0.960 | 0.970 | 0.968 |
| DFM | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| RF | 0.928 | 0.942 | 0.974 | 0.986 | 0.966 |
| MLP | 0.988 | 0.969 | 0.969 | 0.998 | 1.009 |
| TCNN | 0.987 | 0.983 | 0.970 | 1.013 | 0.985 |
| LSTM | 0.989 | 0.992 | 0.954 | 0.979 | 1.002 |

*Note:* Entries report each model's RMSE relative to the DFM at the same horizon; values below one indicate lower RMSE than the DFM.

The verdict reverses. Of the 5 competing models, 5 now improve on the DFM on average, the best ($RF$) at 0.959; at the nowcast 4 of them do, the best reaching 0.928.

The reversal should be read cautiously. The spread between the best and worst model at any horizon is only a few percent of RMSE across 100 replications, so the estimates do not support a decisive ranking among the alternatives. Table 6 instead provides evidence for a narrower claim: the DFM's advantage disappears when it is no longer the true model. The small differences among the remaining methods do not establish a general machine-learning advantage in its place.

### 6.4 The two environments side by side

Table 7 places the two experiments next to each other. The same models, the same protocol and the same 100 replications produce opposite verdicts, and the only thing that changed between them is whether the benchmark is correctly specified.

**Table 7:** Mean relative RMSE versus the DFM benchmark, both environments.

| | **DGP-A** | **DGP-B** |
|---|---|---|
| Models beating the DFM (of 5) | 0 | 5 |
| Best non-DFM model | *RF* | *RF* |
| Best mean relative RMSE | 1.077 | 0.959 |
| At the nowcast ($h = 0$) | 1.333 | 0.928 |
| Machine-learning family mean | 1.195 | 0.979 |

*Note:* Ratios are formed within design cells and then averaged (Eq. 28).

### 6.5 Experiment C: The Real Economy

#### 6.5.1 Results

The first table shows the pre-COVID estimand: a balanced set of origins for which every horizon is scored no later than 2019Q4. It reports accuracy by model and horizon relative to the benchmark.

**Table 8:** Relative RMSE (vs. DFM) by forecast horizon, balanced pre-COVID window (Real data (US)).

| Model | $h = 0$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
|---|---|---|---|---|---|
| AR | 1.277 | 1.146 | 1.035 | 1.007 | 1.002 |
| DFM | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| RF | 1.035 | 1.064 | 1.025 | 0.975 | 0.985 |
| MLP | 1.201 | 1.103 | 0.987 | 1.015 | 1.016 |
| TCNN | 1.157 | 1.106 | 1.038 | 1.005 | 0.991 |
| LSTM | 1.198 | 1.116 | 1.030 | 1.005 | 1.006 |

*Note:* Entries report each model's RMSE relative to the DFM at the same horizon; values below one indicate lower RMSE than the DFM. The table uses 116 balanced forecast origins from 1990Q1 through 2018Q4; scored targets extend through 2019Q4. Expanding estimation samples contain 48–163 observed GDP quarters. The common origin ceiling ensures that no horizon has a target later than 2019Q4.

The next table extends the same expanding-window exercise through the latest complete common quarter in the frozen data snapshot. It therefore includes the COVID-19 shock among the scored outcomes and, at subsequent origins, inside the models' estimation samples.

**Table 9:** Relative RMSE (vs. DFM) by forecast horizon, data through the latest complete quarter (Real data (US)).

| Model | $h = 0$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
|---|---|---|---|---|---|
| AR | 2.252 | 1.088 | 1.015 | 1.007 | 1.002 |
| DFM | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| RF | 1.703 | 1.068 | 1.007 | 1.008 | 0.984 |
| MLP | 1.748 | 1.058 | 0.998 | 1.010 | 1.002 |
| TCNN | 1.859 | 1.063 | 1.026 | 1.001 | 1.006 |
| LSTM | 1.944 | 1.070 | 1.006 | 1.005 | 1.005 |

*Note:* Entries report each model's RMSE relative to the DFM at the same horizon; values below one indicate lower RMSE than the DFM. The table uses 139 balanced forecast origins from 1990Q1 through 2024Q3; scored targets extend through 2025Q3. Expanding estimation samples contain 48–186 observed GDP quarters. This window includes the COVID-19 shock and subsequent origins, so later fits can train on 2020 observations.

The final accuracy table separates the training-sample-length issue from the COVID shock. It retains only pre-COVID targets and origins with at least 100 quarters of observed GDP history, so every fitted model has a 25-year or longer expanding estimation sample.

**Table 10:** Relative RMSE (vs. DFM) by forecast horizon, large pre-COVID samples (Real data (US)).

| Model | $h = 0$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
|---|---|---|---|---|---|
| AR | 1.251 | 1.206 | 1.041 | 1.009 | 1.003 |
| DFM | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| RF | 1.062 | 1.092 | 1.014 | 0.963 | 0.963 |
| MLP | 1.147 | 1.122 | 0.967 | 1.003 | 1.008 |
| TCNN | 1.125 | 1.133 | 1.057 | 0.995 | 0.969 |
| LSTM | 1.150 | 1.155 | 1.013 | 0.985 | 0.987 |

*Note:* Entries report each model's RMSE relative to the DFM at the same horizon; values below one indicate lower RMSE than the DFM. The table uses 64 balanced forecast origins from 2003Q1 through 2018Q4; scored targets extend through 2019Q4. Expanding estimation samples contain 100–163 observed GDP quarters. Every origin has at least 100 observed GDP quarters, and no scored target is later than 2019Q4.

Of the 5 competitors, 0 attain a lower RMSE than the benchmark averaged across horizons, the best being RF at 1.017. At the nowcast horizon, 0 of them improve on the benchmark, the best reaching 1.035. This shows that neural architectures perform better when the sample is larger.

### 6.5.2 Are the differences distinguishable?

Point rankings answer a different question from the one a reader has. At every horizon above the nowcast, adjacent multi-step forecast errors generally share future target innovations, inducing serial dependence in squared-loss differentials. The tests therefore use a Newey–West long-run variance with the Harvey–Leybourne–Newbold small-sample correction; detailed explanation is in Appendix 3.7.

**Table 11:** Diebold–Mariano tests against the DFM benchmark (Real data (US)).

| Horizon | Model | N | NW lags | DM | p | Favours |
|---|---|---|---|---|---|---|
| 0 | AR | 116 | 0 | 1.96 | 0.053 | dfm |
| 0 | RF | 116 | 0 | 0.38 | 0.708 | dfm |
| 0 | MLP | 116 | 0 | 2.00 | 0.048 | dfm |
| 0 | TCNN | 116 | 0 | 1.47 | 0.144 | dfm |
| 0 | LSTM | 116 | 0 | 1.78 | 0.077 | dfm |
| 1 | AR | 116 | 1 | 1.57 | 0.120 | dfm |
| 1 | RF | 116 | 1 | 1.42 | 0.159 | dfm |
| 1 | MLP | 116 | 1 | 2.16 | 0.033 | dfm |
| 1 | TCNN | 116 | 1 | 2.04 | 0.043 | dfm |
| 1 | LSTM | 116 | 1 | 1.82 | 0.071 | dfm |
| 2 | AR | 116 | 2 | 1.47 | 0.144 | dfm |
| 2 | RF | 116 | 2 | 1.27 | 0.207 | dfm |
| 2 | MLP | 116 | 2 | -0.24 | 0.808 | model |
| 2 | TCNN | 116 | 2 | 1.50 | 0.136 | dfm |
| 2 | LSTM | 116 | 2 | 0.85 | 0.400 | dfm |
| 3 | AR | 116 | 3 | 1.11 | 0.268 | dfm |
| 3 | RF | 116 | 3 | -0.84 | 0.404 | model |
| 3 | MLP | 116 | 3 | 0.48 | 0.634 | dfm |
| 3 | TCNN | 116 | 3 | 0.30 | 0.767 | dfm |
| 3 | LSTM | 116 | 3 | 0.11 | 0.916 | dfm |
| 4 | AR | 116 | 4 | 1.16 | 0.246 | dfm |
| 4 | RF | 116 | 4 | -0.46 | 0.649 | model |
| 4 | MLP | 116 | 4 | 0.51 | 0.608 | dfm |
| 4 | TCNN | 116 | 4 | -0.26 | 0.799 | model |
| 4 | LSTM | 116 | 4 | 0.14 | 0.886 | dfm |

*Note:* Squared-error loss, paired by forecast origin. A negative DM statistic favours the model over the benchmark. Standard errors are Newey–West with a truncation of $h$ lags at zero-based horizon $h$, and the Harvey–Leybourne–Newbold small-sample correction is applied, so the statistic is referred to $t(N-1)$.

Across all horizons, 0 of the model–horizon comparisons that favour a competitor over the benchmark are significant at the five percent level; at the nowcast the count is 0. The most favourable competitor overall is RF, with $p = 0.404$.

### 6.5.3 What the real data adds

The contribution of this experiment is not a third ranking to average with the other two. It is a check on the conditions under which the simulated rankings were produced.

First, the environment is no longer chosen. DGP-A fixes the benchmark's approximation error at zero by construction and DGP-B removes that advantage by construction; here neither was

arranged, and the benchmark faces whatever structure the US economy actually has.

Second, the information set is no longer idealised. Both simulated experiments give every model all three months of the current quarter for every indicator. On real data that vintage does not exist, and the measured calendar withholds the most recent month from 90 of 100 indicators and two months from 4 more. The benchmark and the missing-aware neural models are affected differently by that asymmetry, and it is the one respect in which Experiment C is a harder problem than either simulation rather than merely a different one.

Third — and this is a limitation rather than a contribution — the values remain final-vintage. The models see revised history rather than the numbers a forecaster would have had, so the exercise is pseudo real time in values and real time only in the shape of its information set.

# 7 Discussion

In DGP-A the latent factor the DFM estimates *is* the common component of Eq. (9), so the benchmark's approximation error is zero and its remaining loss is estimation error alone. None of the 5 competitors closes that gap (0 of them beat it), and the machine-learning family mean, 1.195, sits above one throughout. This result answers the deliberately asymmetric predictor-set question in the benchmark-favourable environment. The machine-learning models see all 100 indicators, including the increasingly noisy end of the panel, whereas the DFM sees the 15 lowest-$\tau_i$ indicators. The full panel is an information advantage because it contains those measurements and additional ones, but the fitted broad-panel models do not translate that greater availability into lower RMSE when the truth is linear, low-rank, and already represented by the benchmark. As $\tau_{\max}$ rises, the DFM degrades barely at all (0.966 to 0.990) for two related reasons: its curated subset excludes the noisiest series, and its Kalman update can further reduce the gain assigned to a noisy innovation. The variance-preserving design (12) makes that attribution clean because every noise level has the same marginal indicator variance and the Kalman gain responds to the attenuated effective loading $\lambda_i^{\text{eff}}$ of Eq. (14). Machine-learning models reach a comparable representation only through implicit regularization, and the additional capacity of a random forest or an LSTM has nothing extra to extract when the true signal is linear and low-rank. This result is consistent with the Goulet Coulombe et al. (2022) decomposition, which attributes the machine-learning advantage to nonlinearity and regularization rather than to capacity as such.

DGP-B holds the models, protocol, panel width, sample length and replication count fixed while changing the conditional mean to an estimated network fitted to a real macro panel and replacing Gaussian innovations with resampled residuals. Every model, including the DFM, is then misspecified, and 5 of the 5 competitors edge past the benchmark. The predictor-set rule is unchanged: machine-learning models retain the full panel and the DFM retains a selected 15-series panel, but weak relevance and idiosyncratic noise now come from the estimated economy rather than a controlled $\tau_i$ grid. The contrast with DGP-A therefore shows that full-panel learners can sometimes realise their data-availability advantage once the classical benchmark loses correct specification and the DGP becomes nonlinear; it identifies correct specification, rather than intrinsic superiority, as the main source of DGP-A's ranking.

The DGP-B reversal is nevertheless small: the competing models remain within a few percent of one another in RMSE. Those gaps are sufficient to show that the DFM's dominance is not robust to losing correct specification, but they are too small to support a stable ordering among the alternatives. The evidence is therefore informative about the contingency of the DFM result without establishing a machine-learning advantage in its place.

Experiment C anchors this controlled comparison to the observed economy rather than supplying a third data-generating process to rank. Its measured ragged edge and expanding estimation window make the exercise more realistic than either simulation, however it neither confirms nor overturns the simulated ordering; it provides a external check it.

These findings remain conditional on a narrow design. Both simulations concern one target and use full-information vintages, so the publication lags that motivate state-space nowcasting

are absent. DGP-A additionally imposes linearity, Gaussianity and a one-factor structure, whereas DGP-B inherits the limitations of its estimation sample, including truncation before 2020 (Section 3.3.2); the estimated economy therefore contains recessions but no pandemic-scale shock.

Natural extensions include structural breaks and ragged-edge publication lags. Nonlinear indicator loadings or regime switching in DGP-A would reopen, within the controlled design, the channel through which neural networks and tree ensembles typically gain (Kock and Teräsvirta, 2014; Goulet Coulombe et al., 2022; Athey et al., 2019). Heavy-tailed innovations and rare large shocks would test how those methods interact with the variance-preserving scheme. The observation-noise and shock-scale grids of Section 3.3.5 provide an unused DGP-B stress axis, at the cost of the exact interpretation attached to $\tau_i$ in DGP-A. High-frequency predictors and a broader set of model specifications are further extensions of the comparison, since machine learning models are usually suit to large amounts of data.

# 8 Conclusion

This paper compared 2 classical and 4 machine-learning nowcasting models on a controlled mixed-frequency Monte Carlo benchmark, using the Dynamic Factor Model as the reference baseline and running the identical protocol — 100 replications, horizons 0, 1, 2, 3, 4 quarters, the same origins — against two data-generating processes chosen to differ in the benchmark's specification status: the DFM is correctly specified under DGP-A and misspecified under DGP-B. Across both processes, the central predictor-set contrast is held fixed: machine-learning models use the full high-dimensional panel, which gives them the information advantage of broader data availability even though it includes many weak or noisy variables, while the DFM uses a curated 15-indicator subset and benefits from prior selection and parsimony.

The answer depends on the specification. When the process is a low-rank factor structure with Gaussian measurement noise, the DFM is correctly specified and is not beaten: 0 of the 5 competitors improve on it, the best reaching 1.077, and the benchmark's accuracy barely degrades across a twentyfold increase in the measurement-noise ceiling. When the process is instead neural and non-linear estimated from a real macroeconomic panel — so that no competitor contains the truth — 5 of the same 5 models beat it, the best at 0.959. Reported against one process alone, either result would read as a verdict on the methods; reported together, they identify it as a verdict on the environment. The full panel's information advantage is therefore not sufficient for the fitted learners to beat a correctly specified and well-curated DFM in DGP-A, but some flexible models convert that broader availability into forecast gains in DGP-B once the DFM's specification advantage is removed.

The reversal should not be over-read because the DGP-B margins are only a few percent of RMSE. The second experiment establishes that the DFM's advantage is contingent on correct specification, not that a machine-learning alternative has replaced it.

Experiment C (Section 3.4) puts that contingency to the one environment neither simulation could choose. Over 116 quarterly origins of US data, with a publication calendar measured from historical vintages rather than assumed, 0 of the 5 competitors improve on the benchmark. No competitor achieves a statistically significant gain at any horizon, and the indicator values remain revised. For an institution with an established DFM, replacing it would therefore require reliable gains against that benchmark in a genuinely real-time evaluation.

## References


Akepanidtaworn, Klakow and Korkrid Akepanidtaworn (2025). *GDP Nowcasting Performance of Traditional Econometric Models vs Machine-Learning Algorithms: Simulation and Case Studies*. IMF Working Paper 2025/252. International Monetary Fund. DOI: 10.5089/9798229033626.001.

Alvarez, Rocio, Maximo Camacho, and Gabriel Pérez-Quirós (2016). "Aggregate versus Disaggregate Information in Dynamic Factor Models". In: *International Journal of Forecasting* 32.3, pp. 680–694. DOI: 10.1016/j.ijforecast.2015.10.006.

Athey, Susan and Guido W. Imbens (2019). "Machine Learning Methods That Economists Should Know About". In: *Annual Review of Economics* 11, pp. 685–725. DOI: 10.1146/annurev-economics-080217-053433.

Athey, Susan, Julie Tibshirani, and Stefan Wager (2019). "Generalized Random Forests". In: *Annals of Statistics* 47.2, pp. 1148–1178. DOI: 10.1214/18-AOS1709.

Bai, Jushan and Serena Ng (2002). "Determining the Number of Factors in Approximate Factor Models". In: *Econometrica* 70.1, pp. 191–221. DOI: 10.1111/1468-0262.00273.

– (2008). "Forecasting Economic Time Series Using Targeted Predictors". In: *Journal of Econometrics* 146.2, pp. 304–317. DOI: 10.1016/j.jeconom.2008.08.010.

Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun (2018). *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. Preprint 1803.01271. arXiv. DOI: 10.48550/arXiv.1803.01271.

Bańbura, Marta and Michele Modugno (2014). "Maximum Likelihood Estimation of Factor Models on Datasets with Arbitrary Pattern of Missing Data". In: *Journal of Applied Econometrics* 29.1, pp. 133–160. DOI: 10.1002/jae.2306.

Bańbura, Marta and Gerhard Rünstler (2011). "A Look into the Factor Model Black Box: Publication Lags and the Role of Hard and Soft Data in Forecasting GDP". In: *International Journal of Forecasting* 27.2, pp. 333–346. DOI: 10.1016/j.ijforecast.2010.01.011.

Bańbura, Marta et al. (2013). "Now-Casting and the Real-Time Data Flow". In: *Handbook of Economic Forecasting*. Ed. by Graham Elliott and Allan Timmermann. Vol. 2A. Elsevier. Chap. 4, pp. 195–237. DOI: 10.1016/B978-0-444-53683-9.00004-9.

Boivin, Jean and Serena Ng (2006). "Are More Data Always Better for Factor Analysis?" In: *Journal of Econometrics* 132.1, pp. 169–194. DOI: 10.1016/j.jeconom.2005.01.027.

Bok, Brandyn et al. (2018). "Macroeconomic Nowcasting and Forecasting with Big Data". In: *Annual Review of Economics* 10, pp. 615–643. DOI: 10.1146/annurev-economics-080217-053214.

Borovykh, Anastasia, Sander Bohte, and Cornelis W. Oosterlee (2019). "Dilated Convolutional Neural Networks for Time Series Forecasting". In: *Journal of Computational Finance* 22.4, pp. 73–101. DOI: 10.21314/JCF.2018.358.

Breiman, Leo (2001). "Random Forests". In: *Machine Learning* 45, pp. 5–32. DOI: 10.1023/A:1010933404324.

Camacho, Maximo and Gabriel Pérez-Quirós (2010). "Introducing the Euro-Sting: Short-Term Indicator of Euro Area Growth". In: *Journal of Applied Econometrics* 25.4, pp. 663–694. DOI: 10.1002/jae.1174.

Camacho, Maximo and Gabriel Pérez-Quirós (2011). "Spain-STING: Spain Short-Term Indicator of Growth". In: *The Manchester School* 79.s1, pp. 594–616. DOI: 10.1111/j.1467-9957.2010.02212.x.

Camacho, Maximo, Gabriel Pérez-Quirós, and Pilar Poncela (2013). "Short-Term Forecasting for Empirical Economists: A Survey of the Recently Proposed Algorithms". In: *Foundations and Trends in Econometrics* 6.2, pp. 101–161. DOI: 10.1561/0800000018.

Che, Zhengping et al. (2018). "Recurrent Neural Networks for Multivariate Time Series with Missing Values". In: *Scientific Reports* 8.1, p. 6085. DOI: 10.1038/s41598-018-24271-9.

Dauphin, Jean-François et al. (2022). *Nowcasting GDP: A Scalable Approach Using DFM, Machine Learning and Novel Data, Applied to European Economies*. IMF Working Paper 2022/052. International Monetary Fund. DOI: 10.5089/9798400204425.001.

Doz, Catherine, Domenico Giannone, and Lucrezia Reichlin (2011). "A Two-Step Estimator for Large Approximate Dynamic Factor Models Based on Kalman Filtering". In: *Journal of Econometrics* 164.1, pp. 188–205. DOI: 10.1016/j.jeconom.2011.02.012.

Durbin, James and Siem Jan Koopman (2012). *Time Series Analysis by State Space Methods*. 2nd. Oxford University Press.

Efron, Bradley (1979). "Bootstrap Methods: Another Look at the Jackknife". In: *The Annals of Statistics* 7.1, pp. 1–26. DOI: 10.1214/aos/1176344552.

Forni, Mario et al. (2000). "The Generalized Dynamic-Factor Model: Identification and Estimation". In: *Review of Economics and Statistics* 82.4, pp. 540–554. DOI: 10.1162/003465300559037.

Giannone, Domenico, Lucrezia Reichlin, and David Small (2008). "Nowcasting: The Real-Time Informational Content of Macroeconomic Data". In: *Journal of Monetary Economics* 55.4, pp. 665–676. DOI: 10.1016/j.jmoneco.2008.05.010.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. MIT Press.

Goulet Coulombe, Philippe et al. (2022). "How Is Machine Learning Useful for Macroeconomic Forecasting?" In: *Journal of Applied Econometrics* 37.5, pp. 920–964. DOI: 10.1002/jae.2910.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long Short-Term Memory". In: *Neural Computation* 9.8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Hopp, Daniel (2022). "Economic Nowcasting with Long Short-Term Memory Artificial Neural Networks (LSTM)". In: *Journal of Official Statistics* 38.3, pp. 847–873. DOI: 10.2478/jos-2022-0037.

Kingma, Diederik P. and Jimmy Ba (2015). "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations (ICLR)*.

Kock, Anders Bredahl and Timo Teräsvirta (2014). "Forecasting Performances of Three Automated Modelling Techniques during the Economic Crisis 2007–2009". In: *International Journal of Forecasting* 30.3, pp. 616–631. DOI: 10.1016/j.ijforecast.2013.01.003.

Mariano, Roberto S. and Yasutomo Murasawa (2003). "A New Coincident Index of Business Cycles Based on Monthly and Quarterly Series". In: *Journal of Applied Econometrics* 18.4, pp. 427–443. DOI: 10.1002/jae.695.

McCracken, Michael W. and Serena Ng (2016). "FRED-MD: A Monthly Database for Macroeconomic Research". In: *Journal of Business & Economic Statistics* 34.4, pp. 574–589. DOI: 10.1080/07350015.2015.1086655.

Medeiros, Marcelo C. et al. (2021). "Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods". In: *Journal of Business & Economic Statistics* 39.1, pp. 98–119. DOI: 10.1080/07350015.2019.1637745.

Mullainathan, Sendhil and Jann Spiess (2017). "Machine Learning: An Applied Econometric Approach". In: *Journal of Economic Perspectives* 31.2, pp. 87–106. DOI: 10.1257/jep.31.2.87.

Richardson, Adam, Thomas van Florenstein Mulder, and Tuğrul Vehbi (2021). "Nowcasting GDP Using Machine-Learning Algorithms: A Real-Time Assessment". In: *International Journal of Forecasting* 37.2, pp. 941–948. DOI: 10.1016/j.ijforecast.2020.10.005.

Schwarz, Gideon (1978). "Estimating the Dimension of a Model". In: *Annals of Statistics* 6.2, pp. 461–464. DOI: 10.1214/aos/1176344136.

Stock, James H. and Mark W. Watson (2002). "Macroeconomic Forecasting Using Diffusion Indexes". In: *Journal of Business & Economic Statistics* 20.2, pp. 147–162. DOI: 10.1198/073500102317351921.

– (2016). "Dynamic Factor Models, Factor-Augmented Vector Autoregressions, and Structural Vector Autoregressions in Macroeconomics". In: *Handbook of Macroeconomics*. Ed. by John B. Taylor and Harald Uhlig. Vol. 2A. Elsevier. Chap. 8, pp. 415–525. DOI: 10.1016/bs.hesmac.2016.04.002.

Zou, Hui and Trevor Hastie (2005). "Regularization and Variable Selection via the Elastic Net". In: *Journal of the Royal Statistical Society B* 67.2, pp. 301–320. DOI: 10.1111/j.1467-9868.2005.00503.x.