Reservoir Computing for Macroeconomic Forecasting with Mixed Frequency Data

Giovanni Ballarin[1], Petros Dellaportas[2,3], Lyudmila Griliryeva[4,5],
Marcel Hirt[8], Sophie van Huellen[6,7], Juan-Pablo Ortega[8]

June 14, 2023

Abstract

Macroeconomic forecasting has recently started embracing techniques that can deal with large-scale datasets and series with unequal release periods. MIxed-DAta Sampling (MIDAS) and Dynamic Factor Models (DFM) are the two main state-of-the-art approaches that allow modeling series with non-homogeneous frequencies. We introduce a new framework called the Multi-Frequency Echo State Network (MFESN) based on a relatively novel machine learning paradigm called reservoir computing. Echo State Networks (ESN) are recurrent neural networks formulated as nonlinear state-space systems with random state coefficients where only the observation map is subject to estimation. MFESNs are considerably more efficient than DFMs and allow for incorporating many series, as opposed to MIDAS models, which are prone to the curse of dimensionality. All methods are compared in extensive multistep forecasting exercises targeting US GDP growth. We find that our MFESN models achieve superior or comparable performance over MIDAS and DFMs at a much lower computational cost.

Key Words: Reservoir Computing, Echo State Networks, forecasting, US output growth, GDP, mixed-frequency data, time series, Multi-Frequency Echo State Network, MIDAS, DFM

JEL: C53, C45, E17

Acknowledgments: The authors acknowledge financial support from UK Research and Innovation (grant number ES/V006347/1). GB thanks the hospitality and the generosity of the University of Warwick without which this paper would not have happened. GB is grateful to the Center for Doctoral Studies in Economics (CDSE) at the University of Mannheim for providing travel funding through the 2020 CDSE Teaching Award. The authors acknowledge support by the state of Baden-Württemberg through bwHPC. The authors acknowledge the use of the UCL Myriad High Performance Computing Facility (Myriad@UCL), and associated support services, in completing this work. MH acknowledges that the majority of his work was done when he was affiliated with UCL and hence thanks the support of UCL in this project. The authors acknowledge the support of the Swiss National Science Foundation (grant number 200021_175801/1). The authors also thank Thanos Moraitis and Alfonso Silva Ruiz for excellent research assistance and help in data collection and curation.

[1] Department of Economics, University of Mannheim, L7, 3-5, Mannheim, 68131, Germany. Giovanni.Ballarin@gess.uni-mannheim.de

[2] Department of Statistical Science, UCL, Gower Str., London WC1E 6BT, UK. P.Dellaportas@ucl.ac.uk

[3] Department of Statistics, Athens University of Economics and Business, 10434 Athens, Greece. Petros@aueb.gr

[4] Faculty of Mathematics and Statistics, University of St. Gallen, Badanstrasse 6, CH-9000 St. Gallen, Switzerland. Lyudmila.Grigoryeva@unisg.ch

[5] Honorary Associate Professor, Department of Statistics, University of Warwick, Coventry CV4 7AL, UK. Lyudmila.Grigoryeva@warwick.ac.uk

[6] Global Development Institute (GDI), University of Manchester, Manchester M13 9PL, UK. Sophie.vanHuellen@manchester.ac.uk

[7] Department of Economics, SOAS University of London, WC1H 0XG, London, UK. Sv8@soas.ac.uk

[8] Division of Mathematical Sciences, Nanyang Technological University, 21 Nanyang Link, Singapore 637371. MarcelAndre.Hirt@ntu.edu.sg (some of the work done while at UCL), Juan-Pablo.Ortega@ntu.edu.sg

Contents

1 Introduction

The availability of timely and accurate forecasts of key macroeconomic variables is of crucial importance to economic policymakers, businesses, and the banking sector alike. Fundamental macroeconomic figures, such as GDP growth, become available at low frequency with a considerable time lag and are subject to various rounds of revisions after their release. This is particularly problematic in a fast-changing and uncertain economic environment, as experienced during the Great Recession of 2007-2008 (Hindrayanto et al. (2016)) and the recent pandemic (Buell et al. (2021), Huber et al. (2021)). However, a large number of the potentially predictive financial market (and other macroeconomic) indicators are available at a daily or even higher frequency (Andreou et al. (2013)). The desire to utilize such high-frequency data for macroeconomic forecasting has led to the exploration of techniques that can deal with large-scale datasets and series with unequal release periods (see Borio (2011, 2013), Morley (2015); we also refer the reader to Fuleky (2020) for more details regarding high-dimensional data and to Armesto et al. (2010) and Bańbura et al. (2013) for a review on mixed-frequency data).

We contribute to the existing literature by proposing a new macroeconomic forecasting framework that utilizes high-dimensional and mixed-frequency input data, the Multi-Frequency Echo State Network (MFESN). The MFESN originates from a machine learning paradigm called Reservoir Computing (RC). RC is a family of learning models that take advantage of the information processing capabilities of complex dynamical systems (see Maass et al. (2002), Legenstein and Maass (2007), Crutchfield et al. (2010), and Lukoševičius and Jaeger (2009), Tanaka et al. (2019) for reviews). Generally speaking, RC is a versatile class of recurrent neural network (RNN) models (see Salehinejad et al. (2017) for a detailed survey). Although conventional RNNs are well-suited for handling sequence data and dynamic problems, estimating their weights during the training phase is inherently difficult (Pascanu et al. (2013), Doya (1992)). Reservoir networks stand out due to the fact that their inner weights can be *randomly generated* and *fixed*, and only the output (readout) layer weights are subject to estimation (supervised training). Echo State Network (ESN) is one of the most popular instances of RC models with provable universality, generalization properties (see Grigoryeva and Ortega (2018a,b, 2019), Gonon et al. (2020a, 2023a), Gonon and Ortega (2021), and references therein for more details), and excellent performance in forecasting, classification, and learning of dynamical systems (see Hart et al. (2021), Grigoryeva et al. (2021)). While conventional RNNs have been adopted for macroeconomic forecasting in a few instances (see, for example, Paranhos (2021)), to the best of our knowledge, we are the first to explore easily-trainable reservoir models in this context.

Our main contribution is three-fold. First, inspired by the remarkable empirical success of ESNs in prediction tasks, we propose the so-called Multi-Frequency Echo State Network (MFESN) framework, which allows multistep forecasting of the target variable at lower or the same frequencies as those of the input series. Second, we introduce two different approaches to predicting within the MFESN framework, namely *Single-Reservoir MFESN* (S-MFESN) and *Multi-Reservoir MFESN* (M-MFESN). S-MFESN is determined by modifying the ESN architecture to accommodate input and target variables of mixed frequencies. In M-MFESN, several Echo State Networks are adopted to handle input time series, each ESN corresponding to a group of input variables quoted at one given frequency. Finally, our third contribution consists of an extensive empirical comparative analysis of the forecasting capability of the proposed approaches in a concrete task of predicting the quarterly U.S. output growth. We inspect the forecasting capabilities of the MFESN framework compared to two well-established benchmarks widely used in the macroeconomic literature and among practitioners and show its empirical superiority in several thoroughly conducted forecasting exercises. Moreover, as a bi-product, we propose a new data aggregation scheme that allows bridging these two standard forecasting approaches, which is not available in the literature.

In our empirical study, we evaluate the multistep forecasting performance of the MFESN framework targeting quarterly U.S. output growth (Gross Domestic Product (GDP) growth) and utilizing a small- and medium-sized set of monthly and daily financial and macroeconomic variables. We compare the

MFESN approach against two state-of-the-art methods, MIDAS and DFM, known for their ability to incorporate data of heterogeneous frequencies and utilize high-dimensional data inputs. The MIxed DAta Sampling (MIDAS) model developed in Ghysels et al. (2004, 2007) has been adopted widely for macroeconomic forecasting with mixed-frequency data (see for instance Clements and Galvão (2008, 2009), Ghysels and Wright (2009), Francis et al. (2011), Monteforte and Moretti (2012), Galvão and Marcellino (2010), Galvão (2013), Andreou et al. (2013), Ghysels (2016), Jardet and Meunier (2022)). However, MIDAS is prone to curse-of-dimensionality problems and performs poorly when the set of predictors is of even moderate size (Clements and Galvão (2009), Kostrov (2021)) due to optimization-related issues. Recently, some attempts have been made in the literature to overcome these issues by employing variable selection techniques under some additional assumptions. For instance, Babii et al. (2022) proposes the MIDAS projection approach, which is more amenable to high-dimensional data environments under the assumption of sparsity. Even with these improvements, practical high-dimensional implementations of MIDAS remain challenging. This is in part caused by the ragged edges of the "raw" macroeconomic data, incomplete observations, and uneven sampling frequencies. The relative inflexibility of MIDAS regression lag specifications makes integrating daily and weekly data at true calendar frequencies (that is, without interpolation or aggregation) very complex. State-space models effectively mitigate these issues.

A strong state-of-the-art state-space competitor for our MFESN framework is the Dynamic Factor Model (DFM), which has been first introduced in Geweke (1977) and Sargent et al. (1977). DFMs have become the standard workhorse for macroeconomic nowcasting and prediction (for more details, we refer the reader to Stock and Watson (1996, 2002, 2016), Giannone et al. (2008), Bańbura and Rünstler (2011), Chauvet et al. (2015), Hindrayanto et al. (2016)). Conventional DFMs for data of multiple sampling frequencies are linear state-space models with a latent low-frequency process of interest and high-dimensional input time series. Although their linear structure lends itself to inference with likelihood-based methods and Kalman filtering, using DFMs in the high-dimensional setting is limited by the associated computational effort. For Gaussian state-space models, some of these issues are proposed to be handled with a more compact matrix representation as in Delle Monache and Petrella (2019). Still, in the particular settings of nowcasting and forecasting of GDP growth, the computational complexity is one of the main reasons why DFMs are rarely used with daily input series, see Bańbura et al. (2013) for a detailed review and Aruoba et al. (2009) for a mixed-frequency DFM wherein the latent factor process is updated daily, with the highest input frequency being weekly. We address these numerical difficulties using novel Python libraries for auto-differentiation and using GPUs for parallel computing, which allow the estimation of DFMs even in instances of high-frequency input observations. Further, to adapt the DFM to mixed frequency tasks, we propose a new DFM aggregation scheme with Almon polynomial structure that bridges MIDAS and the DFM for our forecasting comparison. To our knowledge, we are the first to present this aggregation scheme which reduces the number of parameters subject to estimation. In contrast, previous DFM such as in Mariano and Murasawa (2003), Bańbura and Rünstler (2011), Camacho and Pérez-Quirós (2010), Frale et al. (2011) commonly assume a fixed aggregation scheme a-priori depending on whether the macroeconomic variable is a flow or stock variable.

To carry out a fair comparison of our MFESN framework with the state-of-the-art MIDAS and DFM models, we designed two model evaluation settings that differ regarding whether the financial crisis of 2007-2008 is included in the estimation period or not. In the first forecasting setting, all the competing models are estimated using the data from January 1st, 1990, until December 31st, 2007. Their performance in the forecasting into and after the financial crisis period is assessed. In the second evaluation setting, fitting is done with data largely encompassing the crisis period, again from January 1st, 1990 but now up to December 31st, 2011. In both cases, the forecasting (testing) period spans time up to the COVID-19 pandemic events, namely the fourth quarter of 2019. Along with the two state-of-the-art DFM and MIDAS models, we use the unconditional mean of the sample as a baseline benchmark against the reservoir models. We find that our ESN-inspired models attain comparable or much better performance than DFMs at a much lower computational cost, even for a relatively long forecasting horizon of four quarters. Additionally, ESNs do not suffer from curse-of-dimensionality problems, which are known to be pervasive for MIDAS models and hence consistently outperform them in a number of forecasting exercises.

The remainder of the paper is structured as follows. Section 2 introduces the notation and terminology used throughout the paper. Section 3 presents reservoir models and discusses their advantages, as well as estimation, hyperparameter tuning, penalization and nonlinear multistep forecasting. In Section 4, we introduce the Multi-Frequency Echo State Network (MFESN) framework, propose the single-reservoir and multi-reservoir MFESN models, and spell out their defining features. Section 5 contains the empirical study of the comparative GDP forecasting performance of MFESNs with respect to the set of benchmark models. We assess one-step and multistep forecasting results in several setups, with a small and a medium-sized set of regressors. We fit models with data before and after the 2007-08 financial crisis, and with different estimation windows. Section 6 concludes and discusses future research avenues and applications. Finally, the Appendix contains detailed information on the implementation of all models, robustness checks and additional figures.

Code. Our code, the data, and all results presented in the paper are made available in the GitHub repository https://github.com/rceconmodelling/reservoir-computing-for-macroeconomic-modelling.

2 Notation and Preliminaries

This section introduces the notation used throughout the paper. It also presents what we call *temporal notation*, which allows us to write multi-frequency models consistently and unambiguously, even when an arbitrary number of sampling frequencies is considered. We also provide definitions of low- and high-frequency forecasting schemes.

2.1 Notation

We use the symbol $\mathbb{N}$ (respectively, $\mathbb{N}^+$) to denote the set of natural numbers with the zero element included (respectively, excluded). $\mathbb{Z}$ denotes the set of all integers. We use $\mathbb{R}$ (respectively, $\mathbb{R}_+$) to denote the set of all (respectively, positive excluding zero element) reals. We abbreviate the set $[n] = \{1, \ldots, n\}$, with $n \in \mathbb{N}^+$.

Vector notation. A column vector is denoted by a bold lowercase symbol like $\boldsymbol{r}$ and $\boldsymbol{r}^\top$ indicates its transpose. Given a vector $\boldsymbol{v} \in \mathbb{R}^n$, we denote its entries by v_i, with $i \in \{1, \ldots, n\}$; we also write $\boldsymbol{v} = (v_i)_{i \in \{1,\ldots,n\}}$. The symbols $\boldsymbol{i}_n, \boldsymbol{0}_n \in \mathbb{R}^n$ stand for the vectors of length n consisting of ones and of zeros, respectively. Additionally, given $n \in \mathbb{N}^+$, $\boldsymbol{e}_n^{(i)} \in \mathbb{R}^n$, $i \in \{1, \ldots, n\}$ denotes the canonical unit vector of length n determined by $\boldsymbol{e}_n^{(i)} = (\delta_{ij})_{j \in \{1,\ldots,n\}}$. For any $\boldsymbol{v} \in \mathbb{R}^n$, $\|\boldsymbol{v}\|$ denotes its Euclidean norm.

Matrix notation. We denote by $\mathbb{M}_{n,m}$ the space of real $n \times m$ matrices with $m, n \in \mathbb{N}^+$. When $n = m$, we use the symbols $\mathbb{M}_n$ and $\mathbb{D}_n$ to refer to the space of square and diagonal matrices of order n, respectively. Given a matrix $A \in \mathbb{M}_{n,m}$, we denote its components by A_{ij} and we write $A = (A_{ij})$, with $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots m\}$. The symbol $\mathbb{I}_n \in \mathbb{D}_n$ denotes the identity matrix, and the symbol $\mathbb{O}_n$ stands for the zero matrix of dimension n. For any $A \in \mathbb{M}_{n,m}$, $\|A\|_2$ denotes its matrix norm induced by the Euclidean norms in $\mathbb{R}^m$ and $\mathbb{R}^n$, and $\|A\|_2 = \sigma_{\max}(A)$, with $\sigma_{\max}(A)$ the largest singular value of A.

Input and target stochastic processes. We fix a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ on which all random variables are defined. The input and target signals are modeled by discrete-time stochastic processes $\boldsymbol{z} = (\boldsymbol{z}_t)_{t\in\mathbb{Z}}$ and $\boldsymbol{y} = (\boldsymbol{y}_t)_{t\in\mathbb{Z}}$ taking values in $\mathbb{R}^K$ and $\mathbb{R}^J$, respectively. Moreover, we write $\boldsymbol{z}(\omega) = (\boldsymbol{z}_t(\omega))_{t\in\mathbb{Z}}$ and $\boldsymbol{y}(\omega) = (\boldsymbol{y}_t(\omega))_{t\in\mathbb{Z}}$ for each outcome $\omega \in \Omega$ to denote the realizations or sample paths of $\boldsymbol{z}$ and $\boldsymbol{y}$, respectively. Since $\boldsymbol{z}$ can be seen as a random sequence in $\mathbb{R}^K$, we write interchangeably $\boldsymbol{z} : \mathbb{Z} \times \Omega \longrightarrow \mathbb{R}^K$ and $\boldsymbol{z} : \Omega \longrightarrow (\mathbb{R}^K)^{\mathbb{Z}}$. The same applies to the analogous assignments involving $\boldsymbol{y}$.

Temporal notation. Let $(u_t)_{t\in I}$, $u_t \in \mathbb{R}$ be a (scalar) time series with I some index set (in this paper it will always be discrete). Time series $(u_t)_{t\in I}$ will be denoted just as (u_t) when the index set I is specified by the context. We write $u_{s_1:s_2} = (u_t)_{t\in\{s_1,\dots,s_2\}}$ for integers $s_1 < s_2$ and time series (u_t). To define the concept of the sampling frequency, we must introduce an additional series, call it $(v_s)_{s\in J}$. The time index J is not the same as I. We assume that u_t is sampled at the coarsest rate; equivalently, it has the *lowest* sampling frequency, which we call in what follows the *reference frequency.* In practice, this means that in the same window of time, u_t will be observed at most as frequently as v_s. The case when the sampling frequency of v_s is strictly higher than that of u_t is of primary interest.

We assume that all sampling happens in instants that are evenly spaced in time. Series other than the reference one and with higher sampling frequencies are given an additional time index, the *tempo index*, written $t, *|\kappa$, where κ is the *frequency multiplier.* Our tempo notation assumes that low- and high-frequency series are sampled with temporal *alignment*: this means that the reference time index t and the tempo index $*|\kappa$ have the following properties.

Definition 2.1 *A reference time index $t \in \mathbb{N}$ and a tempo index $*|\kappa$ for a given high-frequency $\kappa \in \mathbb{N}^+$ are such that the following relations hold*

(i) $t, 0|\kappa \equiv t$

(ii) $t, \kappa|\kappa \equiv t + 1$

(iii) $t, s|\kappa \equiv t + \lfloor s/\kappa \rfloor, (s \bmod \kappa)|\kappa \quad$ for $\forall s \in \mathbb{N}$

(iv) $t, -s|\kappa \equiv (t-1) - \lfloor s/\kappa \rfloor, \kappa - (s \bmod \kappa)|\kappa \quad$ for $\forall s \in \mathbb{N}$,

where mod *is the modulo operation and for any $x \in \mathbb{R}$ the floor operator $\lfloor x \rfloor$ outputs the greatest $z \in \mathbb{N}$ such that $z \leq x$.*

Since we can exchange "frequency" and "frequency multiplier" in the tempo notation, we will make no distinction between the two terms in what follows. We now give an example to clarify the use of the tempo notation.

Example 2.2 (Macroeconomic Mixed Frequency Data) Let (y_t) be the time series of quarterly sampled GDP growth and let (v_r) be the time series of industrial production (IP) available at monthly frequency. Notice that we use two different time indexes, t and r. We assume that both series are observed at the *end* of the relevant time period: GDP is released at the end of each quarter, and IP is released at the end of each month. Additionally, we assume that GDP is observed at the end of the last month of the quarter which coincides with the release of monthly IP. We may then write $t \equiv \lfloor r/3 \rfloor$ for all t and r, since there are exactly 3 months in each quarter. In our tempo notation we proceed in a reverse fashion and we instead anchor time to the lowest frequency index. The frequency multiplier is $\kappa = 3$, therefore $r \equiv t, s|3$ with $s \in \{1, 2, 3\}$.

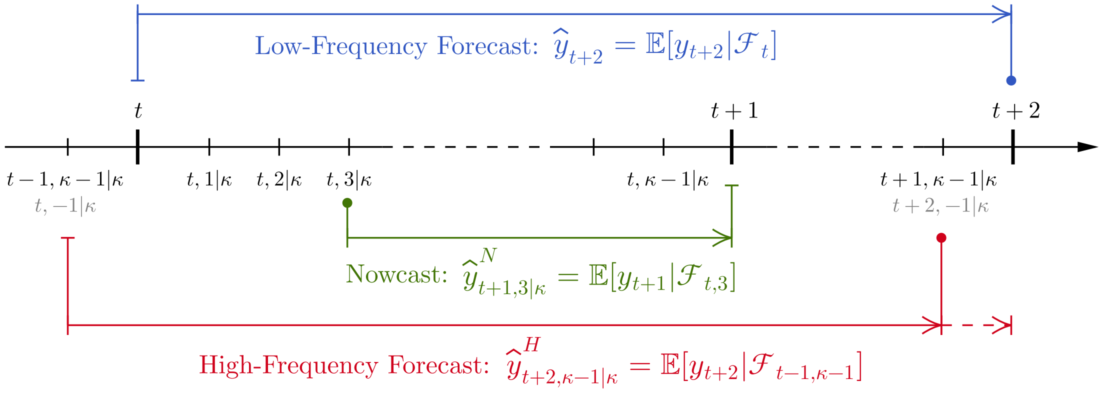

Low-Frequency Forecast: $\widehat{y}_{t+2} = \mathbb{E}[y_{t+2}|\mathcal{F}_t]$
t
$t+1$
$t+2$
$t-1, \kappa-1|\kappa$
$t, -1|\kappa$
$t, 1|\kappa$
$t, 2|\kappa$
$t, 3|\kappa$
$t, \kappa-1|\kappa$
$t+1, \kappa-1|\kappa$
$t+2, -1|\kappa$
Nowcast: $\widehat{y}^{N}_{t+1,3|\kappa} = \mathbb{E}[y_{t+1}|\mathcal{F}_{t,3}]$
High-Frequency Forecast: $\widehat{y}^{H}_{t+2,\kappa-1|\kappa} = \mathbb{E}[y_{t+2}|\mathcal{F}_{t-1,\kappa-1}]$

Figure 1: Diagram of the low-/high-frequency forecasting and nowcasting schemes in tempo notation. Arrows point to time indices of the forecast target, solid dots indicate the high-frequency time placeholder for the constructed high-frequency forecasts.

2.2 Forecasting Schemes

To clarify the design of the forecasting experiments conducted in this paper, we present two different types of prediction illustrated in Figure 1.

Let t denote time in the reference frequency of the target series (y_t) and suppose a regressor (z_r) of frequency κ is included in the forecasting model. The notation can be readily extended to include multiple regressors. Let $h \geq 0$ be a *low-frequency* prediction horizon counted from the last available observation of (y_t). Let $l \geq 0$ be a *high-frequency* horizon with respect to frequency κ.

Low-frequency forecasting. We call an h-steps ahead forecast *low-frequency* when predictions for the target variable are constructed only at the end of the low-frequency periods. The information set which is used at the time of h-steps ahead low-frequency forecasting at t is the σ-algebra defined as

$$\mathcal{F}_t = \sigma\left(\left\{y_t, y_{t-1}, y_{t-2}, \ldots, z_{t,0|\kappa}, z_{t,-1|\kappa}, z_{t,-2|\kappa}, \ldots\right\}\right) \tag{2.1}$$

and, when using the mean square error as a loss, the optimal forecast is given by

$$\widehat{y}_{t+h} = \mathbb{E}\left[y_{t+h}|\mathcal{F}_t\right]. \tag{2.2}$$

High-frequency forecasting. In this forecasting scheme, one may also use high-frequency regressors to produce additional high-frequency forecasts of the low-frequency target variable. For example, in the case of a target released at the end of each year and having monthly quoted covariates, the low-frequency forecasting scheme would correspond to constructing forecasts always at the end of the last month of the year (December). At the same time, with all the information collected up to the end of December, there are other possibilities to construct forecasts. In particular, the forecaster could consider placing herself at the end of any other month of the year instead and construct predictions for the monthly proxy of the yearly variable for the next hth year.

In this scheme, one often artificially *reduces* the information set. Although not all the available information is exploited, this procedure has its benefits: first, it renders high-frequency forecast instances; second, it allows taking into account misspecification due to a seasonal response of (y_t) to (z_r). This is especially important whenever multiple time series with different sampling frequencies are combined in one model and seasonality effects are either difficult to detect or impossible to avoid. In the context of macroeconomic forecasting, we refer the reader to Clements and Galvão (2008, 2009), Chen and Ghysels (2010) and Jardet and Meunier (2022) where these questions are carefully discussed.

Let the forecaster place herself at time t: she wishes to construct a high-frequency forecast for some $t, l|\kappa$ with $l \in \mathbb{N}$. The maximal information set available at t is $\mathcal{F}_t$ as in (2.1). However, if she uses $\mathcal{F}_t$ then the forecast for $t, l|\kappa$ coincides with the low-frequency forecast and is given by (2.2) for any l. Notice that the forecasts can be constructed using the reduced information sets instead. Let $h = \lceil l/\kappa \rceil$, $\ell = l \bmod \kappa$, and $m = h - \lfloor l/\kappa \rfloor$, and define

$$\begin{aligned}\mathcal{F}_{t-m,\ell} &= \sigma\left(\left\{y_{t-m}, y_{t-1-m}, \ldots, z_{t-m,\ell|\kappa},\ z_{t-m,(\ell-1)|\kappa},\ z_{t-m,(\ell-2)|\kappa}, \ldots\right\}\right) \\ &= \sigma\left(\left\{y_{t-m}, y_{t-1-m}, \ldots, z_{t+1-m,-(\kappa-\ell)|\kappa}, z_{t+1-m,-(\kappa-\ell)+1|\kappa}, z_{t+1-m,-(\kappa-\ell+2)|\kappa}, \ldots\right\}\right).\end{aligned}$$

The high-frequency forecast information sets nest the low-frequency forecasting setup since $\mathcal{F}_{t-m,\ell} \equiv \mathcal{F}_t$ if $l = \kappa h$ for $h \in \mathbb{N}$ and the forecast for the high-frequency proxy constructed for the moments $t, l|\kappa$ for the low-frequency variable is provided by the conditional expectation

$$\widehat{y}^{H}_{t+h,\ell|\kappa} = \mathbb{E}\left[y_{t+h} | \mathcal{F}_{t-m,\ell}\right].$$

It is easy to see that if the forecaster is interested in nowcasting, it can be readily obtained by taking $m = 0$ and writing for all $0 < \ell \leq \kappa - 1$:

$$\widehat{y}^{N}_{t+1,\ell|\kappa} = \mathbb{E}\left[y_{t+1} | \mathcal{F}_{t,\ell}\right].$$

We refer the reader to Appendix A for more details on nowcasting and multicasting settings.

3 Reservoir Models

In this section, we introduce *reservoir computing* models (Jaeger and Haas, 2004) for forecasting of stochastic time series of a single frequency. We focus on a family of RC systems called *Echo State Networks* (ESNs), which have been successfully applied to forecasting of deterministic dynamical systems (Pathak et al., 2017, 2018, Wikner et al., 2021, Arcomano et al., 2022). In the following, we discuss the linear estimation of ESN model parameters, the hyperparameters tuning, the loss penalty selection, and how to carry out nonlinear forecasting.

3.1 Reservoir Models

Reservoir computing (RC) models are nonlinear state-space systems that, in the forecasting setting, are defined by the following equations:

$$\boldsymbol{x}_t = F(\boldsymbol{x}_{t-1}, \boldsymbol{z}_t), \tag{3.1}$$

$$\boldsymbol{y}_{t+1} = h_{\boldsymbol{\theta}}(\boldsymbol{x}_t) + \boldsymbol{\epsilon}_t, \tag{3.2}$$

for all $t \in \mathbb{Z}$, where the *state map* $F : \mathbb{R}^N \times \mathbb{R}^K \to \mathbb{R}^N$, $N, K \in \mathbb{N}^+$ is called also the *reservoir map*, and the *observation map* $h_{\boldsymbol{\theta}} : \mathbb{R}^N \to \mathbb{R}^J$, $J \in \mathbb{N}^+$ is referred to as the *readout* layer, parametrized by $\boldsymbol{\theta} \in \Theta$. Sequences $(\boldsymbol{z}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{z}_t \in \mathbb{R}^K$, and $(\boldsymbol{y}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{y}_t \in \mathbb{R}^J$, stand for the *input* and the *output (target)* of the system, respectively, and $(\boldsymbol{x}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{x}_t \in \mathbb{R}^N$, are the associated *reservoir states*. In (3.2), $(\boldsymbol{\epsilon}_t)_{t\in\mathbb{Z}}$ are J-dimensional independent zero-mean innovations with variance $\sigma^2_\epsilon \mathbb{I}_J$ that are also independent of $\boldsymbol{x}_t$ across all t. Importantly, many families of RC systems have been proven to have universal approximation properties for L^p-integrable stochastic processes (Gonon and Ortega, 2020), and estimation and generalization error bounds have been established in Gonon et al. (2020a, 2023a).

In the case of an ESN model, the state and observation equations (3.1)-(3.2) are given by

$$\boldsymbol{x}_t = \alpha \boldsymbol{x}_{t-1} + (1-\alpha)\sigma(A\boldsymbol{x}_{t-1} + C\boldsymbol{z}_t + \boldsymbol{\zeta}) \tag{3.3}$$

$$\boldsymbol{y}_{t+1} = \boldsymbol{a} + W^\top \boldsymbol{x}_t + \boldsymbol{\epsilon}_t, \tag{3.4}$$

where $A \in \mathbb{M}_N$ is the *reservoir matrix*, $C \in \mathbb{M}_{N,K}$ is the *input matrix*, $\boldsymbol{\zeta} \in \mathbb{R}^N$ is the *input shift*, $\alpha \in [0,1)$ is the *leak rate* and $W \in \mathbb{M}_{N,J}$ are the *readout coefficients*. The map $\sigma : \mathbb{R} \to \mathbb{R}$ is an activation function applied elementwise, which in what follows we take to be the hyperbolic tangent. We refer to A, C, $\boldsymbol{\zeta}$ as *state parameters* that are randomly generated. Notice that if $A = 0$ and $\alpha = 0$ the state equation reduces to a nonlinear regression model with random coefficients (or a feedforward neural network with random weights) which is usually referred to as an *Extreme Learning Machine* (Cao et al., 2018, Gonon et al., 2023a).

Properties of ESN models. We focus on ESNs with the so-called *echo state property (ESP)*, that is, when for any $\boldsymbol{z} \in (\mathbb{R}^K)^{\mathbb{Z}}$ there exists a unique $\boldsymbol{y} \in (\mathbb{R}^J)^{\mathbb{Z}}$ such that (3.3)-(3.4) hold (see Grigoryeva and Ortega (2018a,b, 2019) and references therein). One can require that the ESP holds only on the level of the state equation, that is for any input sequence $\boldsymbol{z} \in (\mathbb{R}^K)^{\mathbb{Z}}$ there exists a unique state sequence $\boldsymbol{x} \in (\mathbb{R}^N)^{\mathbb{Z}}$ such that (3.3) holds. The result in Corollary 3.2 in Grigoryeva and Ortega (2018b), which is also valid for the case of ESNs with the leak rate, shows that the sufficient condition of the ESP associated with (3.3) to hold is $\|A\|_2 \, L_\sigma < 1$ where L_σ is the Lipschitz constant of the activation function σ (in our setting, $L_{tanh} = 1$). This sufficient ESP condition has been extensively studied in the ESN literature; see Jaeger (2010), Jaeger and Haas (2004), Buehner and Young (2006), Bai Zhang et al. (2012), Yildiz et al. (2012), Wainrib and Galtier (2016), Manjunath and Jaeger (2013) for more details. The result in Corollary 3.2 in Grigoryeva and Ortega (2018b) also shows that this condition implies the so-called *fading memory property* (Boyd and Chua, 1985), which from the practical point of view means that the impact of initial $\boldsymbol{x}_0$ is negligible for sufficiently long samples.

In the stochastic setting, part (i) of Proposition 4.2 in Grigoryeva and Ortega (2021) proves that the condition $\|A\|_2 < 1$ guarantees variance stationarity of the states associated with variance stationary inputs. Moreover, Manjunath and Ortega (2023) show that this condition implies the so-called stochastic state contractivity ensuring a stochastic analog of the ESP. Notably, violations of $\|A\|_2 < 1$ do not have detrimental implications for the performance of ESNs in various learning tasks, as reported in multiple empirical studies.

Computational advantages of ESNs. We emphasize that the core computational advantage of ESNs is that state parameters A, C, and $\boldsymbol{\zeta}$ are randomly sampled and need not be estimated. Additionally, since observation equation (3.4) is linear in $\boldsymbol{x}_t$, coefficients W can be estimated via (penalized) least squares regression, as we explain in the following subsection. The choice of properties of state parameters determines memory properties and forecasting performance of linear (Ballarin et al., 2023) and nonlinear ESNs (Gonon et al., 2020b) as we discuss in Section 3.2.1.

3.2 Estimation

We now discuss in detail the estimation of coefficients W in (3.4). Let a sample $(\boldsymbol{z}_t, \boldsymbol{y}_t)_{t=1}^T$ of input and target pairs be available. Given an initial state $\boldsymbol{x}_0$, the reservoir states can be computed iteratively according to state equation (3.3) as:

$$\boldsymbol{x}_1 = \alpha \boldsymbol{x}_0 + (1-\alpha)\sigma(A\boldsymbol{x}_0 + C\boldsymbol{z}_1 + \boldsymbol{\zeta}), \quad \ldots, \quad \boldsymbol{x}_T = \alpha \boldsymbol{x}_{T-1} + (1-\alpha)\sigma(A\boldsymbol{x}_{T-1} + C\boldsymbol{z}_T + \boldsymbol{\zeta}).$$

Collect the states and the targets into the state and the observation matrices, respectively, as

$$X = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{T-1})^\top \in \mathbb{M}_{T-1,N}, \qquad Y = (\boldsymbol{y}_2, \boldsymbol{y}_3, \ldots, \boldsymbol{y}_T)^\top \in \mathbb{M}_{T-1,J}.$$

Consider the ridge regression estimator for W given by

$$\widehat{W}_\lambda := \underset{W \in \mathbb{R}^N}{\arg\min} \sum_{t=1}^{T-1} \left\| \boldsymbol{y}_{t+1} - W^\top \boldsymbol{x}_t \right\|_2^2 + \lambda \|W\|_2^2 = \left(X^\top X + \lambda((T-1)\, \mathbb{I}_N) \right)^{-1} X^\top Y, \tag{3.5}$$

where $\lambda \in \mathbb{R}_+$ is the ridge penalty strength. When $\lambda \to 0$, the estimator $\widehat{W}_\lambda$ converges to the minimum-norm least squares solution (Ishwaran and Rao, 2014). In applications, ridge regression is the most commonly used estimation method applied to ESNs, as it provides a straightforward regularization scheme both when $N < T$ and $N \geq T$. This is especially important since in practice the ESN state dimension is often chosen to be 10^3–10^4 (see for example Pathak et al. (2017)). Additionally, a virtue of the ridge regression problem is the fact that the associated objective function is convex and, hence, it can be efficiently solved using stochastic gradient descent even when $\min\{N, T\}$ is large and one decides against the closed-form solution (3.5). Finally, as mentioned in the properties of reservoir systems in Subsection 3.1, we notice that in the presence of the fading memory property, the estimation does not depend significantly on the choice of $\boldsymbol{x}_0$ as sample size T increases.

We refer to (3.5) as the *fixed-parameter* estimator. In our empirical analyses, we also implement *expanding* and *rolling window* estimation strategies which update $\widehat{W}_\lambda$ as new observations become available (we refer the reader to Appendix B.1 for details). In the rest of the paper, for brevity, we use $\widehat{W}$ to denote the ridge estimator of coefficients W assuming that the appropriate choice of the penalty strength λ is made for each concrete situation.

3.2.1 Hyperparameter Tuning

As discussed in Subsection 3.1, the performance of ESNs depends on the choice of randomly drawn state parameters A, C, $\boldsymbol{\zeta}$. Much work has been put into determining optimal specifications (see for example Rodan and Tino (2011), Goudarzi et al. (2016), Farkas et al. (2016), Grigoryeva et al. (2015, 2016), Gonon et al. (2020b)). We construct these parameters by first sampling $\widetilde{A}$, $\widetilde{C}$ and $\widetilde{\boldsymbol{\zeta}}$ from appropriately chosen laws. Then, we normalize each element of the tuple such that

$$\overline{A} = \widetilde{A}/\rho(\widetilde{A}), \quad \overline{C} = \widetilde{C}/\|\widetilde{C}\|, \quad \overline{\boldsymbol{\zeta}} = \widetilde{\boldsymbol{\zeta}}/\|\widetilde{\boldsymbol{\zeta}}\|, \tag{3.6}$$

where $\rho(\widetilde{A})$ denotes the spectral radius of $\widetilde{A}$. As discussed in the properties of reservoir systems in Subsection 3.1, the sufficient condition of the ESP is $\|A\|_2 < 1$. By this normalizing choice, we allow for some more flexibility in terms of marginal violations of the non-sharp ESP constraint. Finally, defining $A = \rho\overline{A}$, $C = \gamma\overline{C}$, and $\boldsymbol{\zeta} = \omega\overline{\boldsymbol{\zeta}}$, we can rewrite state equation (3.3) as

$$\boldsymbol{x}_t = \alpha \boldsymbol{x}_{t-1} + (1-\alpha)\sigma(\rho\overline{A}\boldsymbol{x}_{t-1} + \gamma\overline{C}\boldsymbol{z}_t + \omega\overline{\boldsymbol{\zeta}}). \tag{3.7}$$

We refer to tuple $\boldsymbol{\varphi} := (\alpha, \rho, \gamma, \omega)$ as the *hyperparameters* of the ESN. Specifically, $\alpha \in [0, 1)$ is the leak rate and $\rho \in \mathbb{R}_+$ is called the *spectral radius* of the reservoir matrix, $\gamma \in \mathbb{R}_+$ is the *input scaling*, and $\omega \in \mathbb{R}_+$ is the *shift scaling*. The choice of the hyperparameters determines the properties of the state map. For simplicity, in Section 5, we choose the hyperparameters based on the empirical ESN literature. In Appendix B.2, we also propose a general though more computationally intensive procedure to select hyperparameters in a data-driven way that could be interesting to practitioners.

3.2.2 Penalty Selection

To apply ridge estimator (3.5), it is necessary to first select a penalty λ. Cross-validation (CV) is a common selection procedure for regularization strength in penalized methods such as ridge, LASSO, and Elastic Net. CV techniques have also been applied in the time series context (Kock et al., 2020, Ballarin, 2023) with their validity established in Bergmeir et al. (2018).

In our empirical study, to account for temporal dependence, we use a sequential CV strategy with ten validation folds. More precisely, we reserve the last 50 observations for validation and all other previous data points for training. The first fold consists of the first five observations out of the validation set, and the model is fitted using all training data. The following validation fold comprises the next five subsequent validation observations while the training set is expanded by five data points (from the previous fold). This procedure is repeated ten times and the CV loss is the average of the one-step-ahead forecast MSE on each fold. In expanding or rolling window setups, we rerun the CV penalty selection to ensure that estimated ESN coefficients do not induce oversmoothing. We refer the reader to Appendix B.3 for additional details.

3.3 Relation to Nonparametric Regression

Together with hyperparameters and penalty strength selection, the choice of the state dimension N is a key ingredient of an ESN model. A large state space generally implies better approximation bounds (Gonon et al., 2023a,b). Although it is customary in the empirical literature to take N as large as possible (Lukoševičius, 2012), some recent literature discusses both the statistical risk bounds and the approximation-risk trade-off bounds for various RC families (see Gonon et al. (2020a) and Gonon et al. (2023b) for details). Under simplified assumptions that $\alpha = 0$ and $\rho = 0$ in (3.7), ESNs have a natural connection to random-weights neural networks (Cao et al., 2018) and random projection regression (Maillard and Munos, 2012), and are thus comparable to nonparametric sieve methods. If the data were independently sampled, known results on sieve estimation would require that at most $N/T = o(1)$ up to logarithmic factors for consistency (Belloni et al., 2015). Chen and Christensen (2015) have extended this result to β-mixing data with B-spline and wavelet sieves. Sieve rates appear to suggest that choosing $N = O(T)$ in echo state networks could lead to nontrivial forecasting bias owing to poor approximation properties. Unfortunately, this comparison relies on neglecting the dynamic component of the ESN model, and as such it is only qualitative. It is, therefore, an important topic for future research.

A different but related problem is the potential degradation of forecasting performance when a model is at the interpolation threshold in the overparametrized regime, $N \geq T$. Ridge regression is also commonly applied to address generalization concerns in statistical learning (see Hastie et al. (2009)). Recent work has studied more in-depth the link between regularization and generalization: Hastie et al. (2022) show that “ridgeless”, that is interpolation, solutions can be optimal in some scenarios. However, in our empirical evaluations in Section 5, cross-validation consistently selects non-zero ridge penalties, confirming that ridge penalization plays an important role in ESN forecasting performance.

3.4 ESN Forecasting

We are primarily interested in using ESN models to construct conditional forecasts of target variables. Given that the conditional mean is the best mean square error estimator for h-step-ahead target $\boldsymbol{y}_{t+h}$, $h \geq 1$, our main focus is approximating

$$\widehat{\boldsymbol{y}}_{t+h|t} := \mathbb{E}\left[\boldsymbol{y}_{t+h} | \boldsymbol{x}_{0:t}, \boldsymbol{z}_{0:t}\right] .$$

The case $h = 1$ is trivial, since the ESN model is estimated by regressing $\boldsymbol{y}_{t+1}$ on state $\boldsymbol{x}_t$, and thus we can set $\widetilde{\boldsymbol{y}}_{t+1|t} = \widehat{W}^\top \boldsymbol{x}_t$. However, when $h > 1$ the nonlinear state dynamics precludes a direct computation of the conditional mean. This is in contrast to linear models like VARMAs or DFMs, where the assumption of linearity implies that conditional expectations reduce to simple matrix-vector operations. In particular, linear models are such that the variance (and any other higher-order moments) of the noise term do not impact the conditional mean forecast.

Let $p_\theta(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}, \boldsymbol{z}_t)$ and $g_\theta(\boldsymbol{y}_{t+1} | \boldsymbol{x}_t)$ be the state transition and observation densities, respectively. Then, for $h > 1$,

$$\widehat{\boldsymbol{y}}_{t+h|t} = \int \boldsymbol{y}_{t+h} \, g_\theta(\boldsymbol{y}_{t+h} | \boldsymbol{x}_{t+h-1}) \prod_{j=1}^{h-1} p_\theta(\boldsymbol{x}_{t+j} | \boldsymbol{x}_{t+j-1}, \boldsymbol{z}_{t+j}) \nu(\boldsymbol{z}_{t+j} | \boldsymbol{x}_{t+j-1}) \mathrm{d}\boldsymbol{z}_{t+j} \mathrm{d}\boldsymbol{x}_{t+j} \mathrm{d}\boldsymbol{y}_{t+h}, \quad (3.8)$$

where $\nu(\boldsymbol{z}_{t+j} | \boldsymbol{x}_{t+j-1})$ is the conditional density of inputs. Here, we introduce the additional assumption that $\boldsymbol{x}_{t+j-1}$ is sufficient to condition on past states and inputs, that is

$$\nu(\boldsymbol{z}_{t+j}|\boldsymbol{x}_{t+j-1}) \equiv \nu(\boldsymbol{z}_{t+j}|\boldsymbol{x}_{0:t+j-1}, \boldsymbol{z}_{0:t+j-1}). \tag{3.9}$$

Some elements in the expectation integral are not directly available. Specifically, while an ESN explicitly models both $p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{z}_t)$ and $g_\theta(\boldsymbol{y}_{t+1}|\boldsymbol{x}_t)$, the density $\nu(\boldsymbol{z}_{t+j}|\boldsymbol{x}_{t+j-1})$ is unavailable.

In the remaining part of this subsection, we present a novel ESN-based approach to forecasting the target variable. Our idea is to enrich the ESN model with an auxiliary observation equation for the input covariates. As we demonstrate in Section 5, our proposed method shows superior performance with respect to the standard state-of-the-art benchmarks.

3.4.1 Multi-step Forecasting of Targets via Iterative Forecasting of Inputs

In general, we are interested in constructing forecasts of target variables that are not the same as the model inputs. To do so, we resolve the issue of the intractability of (3.8) while simultaneously capitalizing on the available results using ESNs in the forecasting of dynamical systems. More explicitly, we add to the ESN specification (3.3)-(3.4) an equation that allows sidestepping modeling the density ν directly, thus making the computation of $\widehat{\boldsymbol{y}}_{t+h|t}$ feasible even when $h > 1$.

Consider the ESN where the reservoir states $(\boldsymbol{x}_t)_{t\in\mathbb{Z}}$ follow (3.3), while the target sequence is the same as the input sequence $(\boldsymbol{z}_t)_{t\in\mathbb{Z}}$,

$$\boldsymbol{x}_t = \alpha \boldsymbol{x}_{t-1} + (1-\alpha)\sigma(A\boldsymbol{x}_{t-1} + C\boldsymbol{z}_t + \boldsymbol{\zeta}) \tag{3.10}$$

$$\boldsymbol{z}_{t+1} = \mathcal{W}^\top \boldsymbol{x}_t + \boldsymbol{u}_{t+1}. \tag{3.11}$$

Here, we use symbol $\mathcal{W}$ for the output coefficients to separate this case from the general ESN equations (3.3)-(3.4). In (3.11), $(\boldsymbol{u}_t)_{t\in\mathbb{Z}}$ are K-dimensional independent zero-mean innovations with variance $\sigma_u^2 \mathbb{I}_K$ that are also independent of $\boldsymbol{x}_t$ across all t.

In this case, the reservoir map $F(\boldsymbol{x}_{t-1}, \boldsymbol{z}_t)$ in (3.1) is determined by (3.10), and it is possible to re-feed the forecasted variables back into the state equation as inputs. This yields the following state recursion:

$$\boldsymbol{x}_t = F(\boldsymbol{x}_{t-1}, \mathcal{W}^\top \boldsymbol{x}_{t-1} + \boldsymbol{u}_t) =: G_\theta(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t),$$

where the subscript θ denotes the dependence on the model coefficients. In the reservoir computing literature, regimes, where the ESN state equation is iteratively fed with the model outputs, are called "autonomous" (Gonon et al., 2020b). They are widely and successfully utilized for the prediction of deterministic dynamical systems. Indeed, in those instances, provided that the ridge estimate $\widehat{\mathcal{W}}$ is available from data according to Subsection 3.2, the $h > 1$ steps autonomous state iteration is given by

$$F_\theta^*(\boldsymbol{x}_t) := \alpha \boldsymbol{x}_t + (1-\alpha)\sigma((A + C\widehat{\mathcal{W}}^\top)\boldsymbol{x}_t + \boldsymbol{\zeta})$$

and

$$\boldsymbol{x}_{t+h} = \underbrace{F_\theta^* \circ F_\theta^* \circ \cdots \circ F_\theta^*}_{h \text{ times}}(\boldsymbol{x}_t).$$

Hence one can directly obtain the h-steps ahead predictions of the input time series as $\boldsymbol{z}_{t+h} = \widehat{\mathcal{W}}^\top \boldsymbol{x}_{t+h-1}$.

In the case of stochastic target variables, assuming (3.9), we notice that for the conditional forecast of the states, it holds that

$$\widehat{\boldsymbol{x}}_{t+1|t} = \mathbb{E}\left[\boldsymbol{x}_{t+1}|\boldsymbol{x}_{0:t}, \boldsymbol{z}_{0:t}\right] = \int \boldsymbol{x}_t\, p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{z}_t)\nu(\boldsymbol{z}_t|\boldsymbol{x}_{t-1})\mathrm{d}\boldsymbol{z}_t = \int G_\theta(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\phi(\boldsymbol{u}_t)\mathrm{d}\boldsymbol{u}_t, \tag{3.12}$$

where density ϕ of $\boldsymbol{u}_t$ is, again, unavailable. Note that, even under the assumption $\boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \Sigma_{\boldsymbol{u}})$, which is standard in the filtering literature, the presence of nonlinear map G_θ makes the computation of the forecasts of $\boldsymbol{z}_{t+h}$ a non-straightforward exercise. Nevertheless, this forecast construction can be readily used when one is interested exclusively in predicting the time series $\boldsymbol{z}_t$.

Whenever the final goal of the exercise is forecasting some other explained variable $\boldsymbol{y}_{t+h}$ h-steps ahead, additional issues arise. In this case, one needs to compute the conditional expectation in (3.8) which is intractable even under Gaussian assumptions on the innovations. One option is to apply particle filtering techniques such as bootstrap sampling or sequential importance sampling (SIS) to evaluate the expectation (Doucet et al., 2001). We emphasize that the state dimension is usually chosen to be large, and hence implementing filtering techniques requires some care.

Our approach is to avoid dealing with the nonlinear densities involved in (3.8) with the help of (3.12) and, instead, to reduce the computation of the conditional expectation $\widehat{\boldsymbol{y}}_{t+h|t}$ to a composition of functions. By the linearity of observation equation (3.4) and the assumption of independence in the zero-mean noise $\boldsymbol{\epsilon}_{t+h}$, we write

$$\widehat{\boldsymbol{y}}_{t+h|t} = W^\top \widehat{\boldsymbol{x}}_{t+h-1|t} = \int W^\top \boldsymbol{x}_{t+h-1} \prod_{j=1}^{h-1} p_\theta(\boldsymbol{x}_{t+j}|\boldsymbol{x}_{t+j-1}, \boldsymbol{z}_{t+j}) \nu(\boldsymbol{z}_{t+j}|\boldsymbol{x}_{t+j-1}) \mathrm{d}\boldsymbol{x}_{t+j} \mathrm{d}\boldsymbol{z}_{t+j}$$

and use the approximation

$$\widehat{\boldsymbol{y}}_{t+h|t} \approx \widetilde{\boldsymbol{y}}_{t+h} = W^\top \underbrace{F^*_\theta \circ F^*_\theta \circ \cdots \circ F^*_\theta}_{h-1 \text{ times}} (\boldsymbol{x}_t), \tag{3.13}$$

which originates from

$$\widehat{\boldsymbol{x}}_{t|t-1} = \int G_\theta(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\phi(\boldsymbol{u}_t)\mathrm{d}\boldsymbol{u}_t \approx G_\theta(\boldsymbol{x}_{t-1}, \mathbb{E}[\boldsymbol{u}_t]) = F(\boldsymbol{x}_{t-1}, \mathcal{W}^\top \boldsymbol{x}_{t-1}) \equiv F^*_\theta(\boldsymbol{x}_{t-1}), \tag{3.14}$$

where $\boldsymbol{u}_t$ is assumed to be zero-mean. The validity of (3.14) itself requires implicit assumptions on the nature of the distribution of $\boldsymbol{u}_t$, but here we want to keep the analysis of $\widehat{\boldsymbol{y}}_{t+h|t}$ to a minimum, and just use the insights from the dynamical systems ESN literature. We are hence not delving deeper into alternative approaches to estimate forecasts or, more generally, to compute conditional expectations of ESN models with stochastic inputs.

4 Multi-Frequency Echo State Models

In this subsection, we construct a broad class of ESN models that can accommodate input and target time series sampled at distinct sampling frequencies. We call this family of reservoir models the *Multi-Frequency Echo State Networks* (MFESNs). The state-space structure of MFESNs is naturally amenable to the setting of time series with mixed frequencies. Additionally, the prediction strategy discussed in Section 3.4 is straightforward to extend to MFESNs.

We present two groups of MFESN architectures. The first family is based on a single echo state network architecture and we call these models *Single-Reservoir Multi-Frequency Echo State Networks* (S-MFESNs). The second group, referred to as *Multi-Reservoir Multi-Frequency Echo State Networks* (M-MFESNs), allows for as many state equations as the number of distinct sampling frequencies present in the input data.

4.1 Single-Reservoir MFESN

Recall that, in the temporal notation of Definition 2.1, we reserve t to be the reference time index, which is also used for the target variable, and all other frequencies will be measured with respect to the reference frequency.

Consider L collections of different time series. We assume that the lth collection, $l \in [L]$, consists of n_l time series that are sampled at a common frequency κ_l and contain observations $(\boldsymbol{z}^{(l)}_{t,s|\kappa_l})_{t,s}$ with $\boldsymbol{z}^{(l)}_{t,s|\kappa_l} \in \mathbb{R}^{n_l}$ for all $t \in \mathbb{Z}$ and $s \in \{0, \ldots, \kappa_l - 1\}$. Let $\kappa_{\max} = \max_l \kappa_l$ be the highest sampling frequency among the L time series groups and let $q_l := \kappa_{\max}/\kappa_l$ indicate how low each κ_l sampling frequency is with respect to $\kappa_{\max}$. We can now stack together and repeat the observations in a way that is consistent with the high-frequency index by defining

$$\boldsymbol{z}_{t,s|\kappa_{\max}} := \left(\boldsymbol{z}^{(1)\top}_{t,\lfloor s/q_1 \rfloor|\kappa_1}, \boldsymbol{z}^{(2)\top}_{t,\lfloor s/q_2 \rfloor|\kappa_2}, \ldots, \boldsymbol{z}^{(L)\top}_{t,\lfloor s/q_L \rfloor|\kappa_L}\right)^\top \in \mathbb{R}^{\sum_{l=1}^{L} n_l}, \quad s \in \{0, \ldots, \kappa_{\max} - 1\},$$

where for all $l \in [L]$, $\boldsymbol{z}^{(l)}_{0,0|\kappa_l} = \mathbf{0}_{n_l}$. Thus, it is possible to write a single high-frequency ESN as

$$\boldsymbol{x}_{t,s|\kappa_{\max}} = \alpha \boldsymbol{x}_{t,s-1|\kappa_{\max}} + (1-\alpha)\sigma(A\boldsymbol{x}_{t,s-1|\kappa_{\max}} + C\boldsymbol{z}_{t,s|\kappa_{\max}} + \boldsymbol{\zeta}), \tag{4.1}$$

$$\boldsymbol{z}_{t,s+1|\kappa_{\max}} = \mathcal{W}^\top \boldsymbol{x}_{t,s|\kappa_{\max}} + \boldsymbol{u}_{t,s+1|\kappa_{\max}}, \tag{4.2}$$

where $\mathcal{W} \in \mathbb{M}_{N,\sum_{l=1}^{L} n_l}$ and $s > 0$. We term this class of MFESN models the *Single-Reservoir Multi-Frequency ESNs* (S-MFESNs).

Notice that equations (4.1)-(4.2) of the S-MFESN model prescribe the dynamics at the highest frequency, $\kappa_{\max}$. In order to forecast a lower frequency target, we map high-frequency states $\boldsymbol{x}_{t,s|\kappa_{\max}}$ to low-frequency targets $\boldsymbol{y}_{t+1} \in \mathbb{R}^J$ by introducing a *state alignment* scheme. An *aligned* S-MFESN uses the most recent state with respect to the reference time index t to construct the forecast. More precisely, the state equation of an S-MFESN is iterated $\kappa_{\max}$ times until the state $\boldsymbol{x}_{t-1,\kappa_{\max}|\kappa_{\max}} = \boldsymbol{x}_{t,0|\kappa_{\max}}$ is obtained and then target $\boldsymbol{y}_{t+1}$ is forecast with observation equation

$$\boldsymbol{y}_{t+1} = W^\top \boldsymbol{x}_{t,0|\kappa_{\max}} + \boldsymbol{\epsilon}_{t+1}, \quad W \in \mathbb{M}_{N,J}. \tag{4.3}$$

Estimation of aligned S-MFESN. Both coefficient matrices W and $\mathcal{W}$ can be estimated as explained in Subsection 3.2 under appropriate choices of corresponding penalty strengths. In particular, in order to obtain $\widehat{\mathcal{W}}$, the state and the observation matrices in (3.5) are given by

$$X_{\kappa_{\max}} = (\boldsymbol{x}_{1,0|\kappa_{\max}}, \ldots, \boldsymbol{x}_{1,\kappa_{\max}-1|\kappa_{\max}}, \ldots, \boldsymbol{x}_{T-1,0|\kappa_{\max}}, \ldots, \boldsymbol{x}_{T-1,\kappa_{\max}-1|\kappa_{\max}})^\top \in \mathbb{M}_{(T-1)\kappa_{\max}-1,N},$$

$$Y_{\kappa_{\max}} = (\boldsymbol{z}_{1,1|\kappa_{\max}}, \ldots, \boldsymbol{z}_{1,\kappa_{\max}|\kappa_{\max}}, \ldots, \boldsymbol{z}_{T-1,1|\kappa_{\max}}, \ldots, \boldsymbol{z}_{T-1,\kappa_{\max}|\kappa_{\max}})^\top \in \mathbb{M}_{(T-1)\kappa_{\max}-1,\sum_{l=1}^{L} n_l},$$

while

$$X = \left(\boldsymbol{x}_{1,0|\kappa_{\max}}, \boldsymbol{x}_{2,0|\kappa_{\max}}, \ldots, \boldsymbol{x}_{T-1,0|\kappa_{\max}}\right)^\top \in \mathbb{M}_{T-1,N},$$

$$Y = (\boldsymbol{y}_2, \ldots, \boldsymbol{y}_T)^\top \in \mathbb{M}_{T-1,J},$$

are used for the estimation of $\widehat{W}$. We note that the state equation (4.1) of S-MFESN can be initialized by $\boldsymbol{x}_{0,0|\kappa_{\max}}$, which under the fading memory property is inconsequential for long enough samples (see the discussion in Subsection 3).

Forecasting with aligned S-MFESN. Let $\widehat{W}$ and $\widehat{\mathcal{W}}$ be the sample estimates of the readout matrices as explained above. The fitted high-frequency autonomous state transition map associated with (4.1) is given by

$$F_{\kappa_{\max}}(\boldsymbol{x}_{t,s-1|\kappa_{\max}}) := \alpha \boldsymbol{x}_{t,s-1|\kappa_{\max}} + (1-\alpha)\sigma\left((A + C\widehat{\mathcal{W}}^\top)\boldsymbol{x}_{t,s-1|\kappa_{\max}} + \boldsymbol{\zeta}\right), \tag{4.4}$$

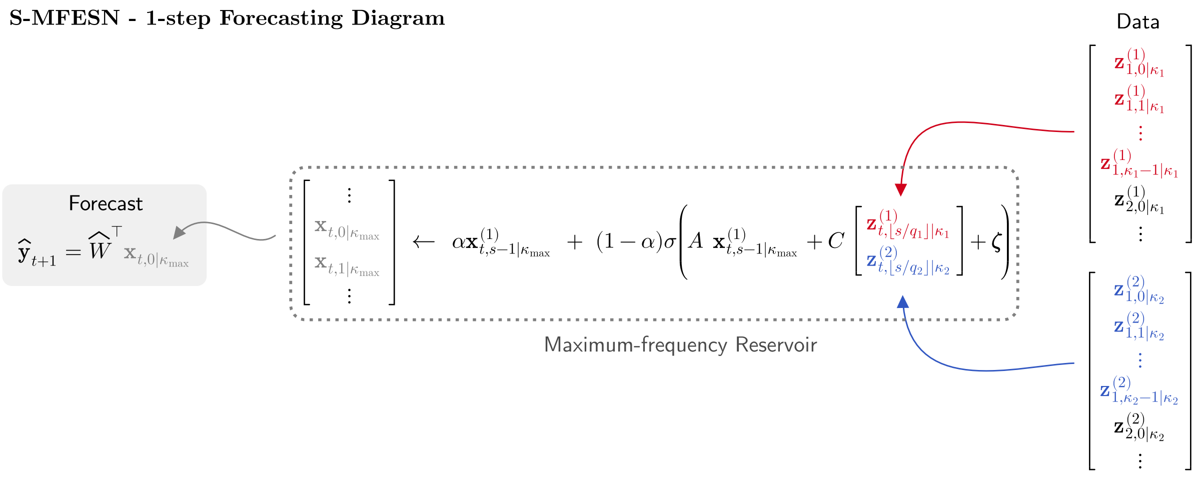

S-MFESN - 1-step Forecasting Diagram
Data
$\mathbf{z}^{(1)}_{1,0|\kappa_1}$
$\mathbf{z}^{(1)}_{1,1|\kappa_1}$
$\mathbf{z}^{(1)}_{1,\kappa_1-1|\kappa_1}$
$\mathbf{z}^{(1)}_{2,0|\kappa_1}$
Forecast
$\widehat{\mathbf{y}}_{t+1} = \widehat{W}^\top \mathbf{x}_{t,0|\kappa_{\max}}$
$\mathbf{x}_{t,0|\kappa_{\max}}$
$\mathbf{x}_{t,1|\kappa_{\max}}$
$\leftarrow \alpha\mathbf{x}^{(1)}_{t,s-1|\kappa_{\max}} + (1-\alpha)\sigma\left(A\,\mathbf{x}^{(1)}_{t,s-1|\kappa_{\max}} + C\begin{bmatrix}\mathbf{z}^{(1)}_{t,\lfloor s/q_1\rfloor|\kappa_1}\\ \mathbf{z}^{(2)}_{t,\lfloor s/q_2\rfloor|\kappa_2}\end{bmatrix} + \zeta\right)$
Maximum-frequency Reservoir
$\mathbf{z}^{(2)}_{1,0|\kappa_2}$
$\mathbf{z}^{(2)}_{1,1|\kappa_2}$
$\mathbf{z}^{(2)}_{1,\kappa_2-1|\kappa_2}$
$\mathbf{z}^{(2)}_{2,0|\kappa_2}$

Figure 2: Scheme of a Single-Reservoir MFESN (S-MFESN) model combining input data sampled at two frequencies with state alignment and estimation for one-step ahead forecasting of the target series.

which, composed with itself exactly $\kappa_{\max}$ times, yields the target-frequency-aligned autonomous state transition map

$$F(\boldsymbol{x}_{t,0|\kappa_{\max}}) := \underbrace{F_{\kappa_{\max}} \circ F_{\kappa_{\max}} \circ \cdots \circ F_{\kappa_{\max}}}_{\kappa_{\max}\text{ times}}(\boldsymbol{x}_{t,0|\kappa_{\max}}). \tag{4.5}$$

Finally, from (3.13) the h-steps ahead low-frequency forecasts, $h \in \mathbb{N}$, can be computed as

$$\widetilde{y}_{T+h|T} = \widehat{W}^\top\big(\underbrace{F \circ F \circ \cdots \circ F}_{h-1\text{ times}}(\boldsymbol{x}_{T,0|\kappa_{\max}})\big). \tag{4.6}$$

Figure 2 gives a graphical diagram of the 1-step forecasting procedure for an S-MFESN. Additionally, Figure 12 in Appendix I provides a similar diagram for the case of multistep forecasts.

The following example illustrates this proposed forecasting strategy for the case of quarterly GDP forecasting using monthly and daily series inputs.

Example 4.1 Suppose that we wish to use an aligned S-MFESN model to forecast a quarterly one-dimensional target (y_t) using $n_{(\texttt{m})}$ monthly and $n_{(\texttt{d})}$ daily series, $(\boldsymbol{z}^{(\texttt{m})}_{t,s|\kappa_1})$ and $(\boldsymbol{z}^{(\texttt{d})}_{t,s|\kappa_2})$, respectively. We adopt the assumption that daily data is released 24 days over each calendar month and hence $\kappa_1 = 3$, $\kappa_2 = 72$ and $\kappa_{\max} = 72$, while $q_1 = 24$ and $q_2 = 1$. Let $t, *|72$ be the temporal index with a quarterly reference frequency. The input vector for the S-MFESN state equation consistent with the daily frequency is given by

$$\boldsymbol{z}^{(\texttt{m},\texttt{d})}_{t,s|72} := (\boldsymbol{z}^{(\texttt{m})\top}_{t,\lfloor s/24\rfloor|3}, \boldsymbol{z}^{(\texttt{d})\top}_{t,s|72})^\top \in \mathbb{R}^{n_{(\texttt{m})}+n_{(\texttt{d})}} \text{ with } \boldsymbol{z}^{(\texttt{d})}_{0,0|3} = \mathbf{0}_{n_{(\texttt{d})}} \text{ and } \boldsymbol{z}^{(\texttt{m})}_{0,0|24} = \mathbf{0}_{n_{(\texttt{m})}}.$$

The complete S-MFESN model with the state space dimension N can be written as:

$$\boldsymbol{x}^{(\texttt{m},\texttt{d})}_{t,s|72} = \alpha\boldsymbol{x}^{(\texttt{m},\texttt{d})}_{t,s-1|72} + (1-\alpha)\sigma(A\boldsymbol{x}^{(\texttt{m},\texttt{d})}_{t,s-1|72} + C\boldsymbol{z}^{(\texttt{m},\texttt{d})}_{t,s|72} + \boldsymbol{\zeta}), \tag{4.7}$$

$$\boldsymbol{z}^{(\texttt{m},\texttt{d})}_{t,s+1|72} = \mathcal{W}^\top\boldsymbol{x}^{(\texttt{m},\texttt{d})}_{t,s|72} + \boldsymbol{u}_{t,s+1|72}, \tag{4.8}$$

$$y_{t+1} = W^\top\boldsymbol{x}^{(\texttt{m},\texttt{d})}_{t,0|\kappa_{\max}} + \boldsymbol{\epsilon}_{t+1}, \tag{4.9}$$

where the state equations (4.7)-(4.8) are run in their own maximum frequency temporal index $s > 0$, and only the states $\boldsymbol{x}_{t-1,\kappa_{\max}|\kappa_{\max}} = \boldsymbol{x}_{t,0|\kappa_{\max}}$ are used in the observation equation (4.9). Provided the input-target pairs sample of length T, the coefficient matrices $\mathcal{W} \in \mathbb{M}_{N,n_{(\texttt{m})}+n_{(\texttt{d})}}$ in (4.8) and $W \in \mathbb{R}^N$

in (4.9) can be estimated via ridge regression as explained above.

From (4.4) the high-frequency autonomous state transition map is given by

$$F_{72}^{(\mathtt{m},\mathtt{d})}(\boldsymbol{x}_{t,s-1|72}^{(\mathtt{m},\mathtt{d})}) := \alpha \boldsymbol{x}_{t,s-1|72}^{(\mathtt{m},\mathtt{d})} + (1-\alpha)\sigma\left((A + C\widehat{\mathcal{W}}^\top)\boldsymbol{x}_{t,s-1|72}^{(\mathtt{m},\mathtt{d})} + \boldsymbol{\zeta}\right),$$

which, composed with itself exactly 72 times, by (4.5) yields the target-frequency-aligned autonomous state transition map

$$F^{(\mathtt{m},\mathtt{d})}(\boldsymbol{x}_{t,0|72}^{(\mathtt{m},\mathtt{d})}) := \underbrace{F_{72}^{(\mathtt{m},\mathtt{d})} \circ F_{72}^{(\mathtt{m},\mathtt{d})} \cdots \circ F_{72}^{(\mathtt{m},\mathtt{d})}}_{\text{72 times}}(\boldsymbol{x}_{t,0|72}^{(\mathtt{m},\mathtt{d})}).$$

By applying $F^{(\mathtt{m},\mathtt{d})}$ to state $\boldsymbol{x}_{t,0|72}^{(\mathtt{m},\mathtt{d})}$ we iterate the S-MFESN forward in time to provide an estimate for $\boldsymbol{x}_{t+1,0|72}^{(\mathtt{m},\mathtt{d})}$, which can then be linearly projected using $\widehat{W}$ to yield a forecast for y_{t+2}. For the target variable, as well as forecasts, we do not use our temporal notation for the sake of compactness and clarity of exposition. Finally, the quarterly forecasts for $h \in \mathbb{N}$ can be computed using (4.6) as

$$\widetilde{y}_{T+h|T} = \widehat{W}^\top \big(\underbrace{F^{(\mathtt{m},\mathtt{d})} \circ F^{(\mathtt{m},\mathtt{d})} \circ \cdots \circ F^{(\mathtt{m},\mathtt{d})}}_{h-1 \text{ times}}(\boldsymbol{x}_{T,0|72}^{(\mathtt{m},\mathtt{d})})\big).$$

4.2 Multi-Reservoir MFESN

Constructing an MFESN with a single reservoir is not necessarily the most effective modeling strategy. Having more than one reservoir allows more flexible modeling of state dynamics for different subsets of input variables sampled at common frequencies. For example, suppose quarterly and monthly data are used as regressors. Our presentation is general enough to accommodate other types of partitioning of series into the corresponding reservoir models. We leave it to future research to test other approaches based, for instance, on markets or data types as done in van Huellen et al. (2020).

Assume again L groups of series with input observations $(\boldsymbol{z}_{t,s|\kappa_l}^{(l)})_{t,s}$ with $\boldsymbol{z}_{t,s|\kappa_l}^{(l)} \in \mathbb{R}^{n_l}$, $l \in [L]$, for all $t \in \mathbb{Z}$ and $s \in \{0, \dots, \kappa_l - 1\}$ sampled at common frequencies $\{\kappa_1, \dots, \kappa_L\}$, respectively. For each of the L groups of input series we define the corresponding ESN model as

$$\boldsymbol{x}_{t,s|\kappa_l}^{(l)} = \alpha_l \boldsymbol{x}_{t,s-1|\kappa_l}^{(l)} + (1-\alpha_l)\sigma(A_l \boldsymbol{x}_{t,s-1|\kappa_l}^{(l)} + C_l \boldsymbol{z}_{t,s|\kappa_l}^{(l)} + \boldsymbol{\zeta}_l), \tag{4.10}$$

$$\boldsymbol{z}_{t,s+1|\kappa_l}^{(l)} = \mathcal{W}_l^\top \boldsymbol{x}_{t,s|\kappa_l}^{(l)} + \boldsymbol{u}_{t,s+1|\kappa_l}^{(l)}, \quad l \in [L], \tag{4.11}$$

with $s > 0$, $\mathcal{W}_l \in \mathbb{M}_{N_l,n_l}$ with N_l the dimension of the state space. Notice that the time index s is different for each l according to our temporal notation introduced in Definition 2.1 and each state equation runs at its own frequency κ_l. The dimensions $\{N_1, N_2, \dots, N_L\}$ of the state spaces can be chosen for the L reservoir models individually. Additionally, multiple reservoirs have the associated hyperparameter tuples $\{\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_L\}$ to be tuned. This requires some care whenever one wants to optimize all hyperparameters jointly. Since there are L reservoir state equations, we call this class of MFESN models *Multi-Reservoir Multi-Frequency ESN* (M-MFESN).

Similar to S-MFESN, all L state equations are iterated each κ_l times respectively until the states $\boldsymbol{x}_{t-1,\kappa_l|\kappa_l}^{(l)} = \boldsymbol{x}_{t,0|\kappa_l}^{(l)}$ are obtained. The *aligned* M-MFESN observation equation is given by

$$\boldsymbol{y}_{t+1} = W^\top \boldsymbol{x}_{t,L} + \boldsymbol{\epsilon}_{t+1}, \text{ with } \boldsymbol{x}_{t,L} = \begin{pmatrix} \boldsymbol{x}_{t,0|\kappa_1}^{(1)} \\ \vdots \\ \boldsymbol{x}_{t,0|\kappa_L}^{(L)} \end{pmatrix} \in \mathbb{R}^{\sum_{l=1}^L N_l}, \; W \in \mathbb{M}_{\sum_{l=1}^L N_l, J}. \tag{4.12}$$

Estimation of aligned M-MFESN. The coefficient matrices W_l, $l \in [L]$, and $\mathcal{W}$ can be estimated similarly to the case of S-MFESN. The state and observation matrices for the estimation of $\widehat{\mathcal{W}}_l$, $l \in [L]$, in (3.5) are constructed as

$$\begin{aligned} X^{(l)} &= (\boldsymbol{x}^{(l)}_{1,0|\kappa_l}, \ldots, \boldsymbol{x}^{(l)}_{1,\kappa_l-1|\kappa_l}, \ldots, \boldsymbol{x}^{(l)}_{T-1,0|\kappa_l}, \ldots, \boldsymbol{x}^{(l)}_{T-1,\kappa_l-1|\kappa_l})^\top \in \mathbb{M}_{(T-1)\kappa_l-1,N_l}, \\ Y^{(l)} &= (\boldsymbol{z}^{(l)}_{1,1|\kappa_l}, \ldots, \boldsymbol{z}^{(l)}_{1,\kappa_l|\kappa_l}, \ldots, \boldsymbol{z}^{(l)}_{T-1,1|\kappa_l}, \ldots, \boldsymbol{z}^{(l)}_{T-1,\kappa_l|\kappa_l})^\top \in \mathbb{M}_{(T-1)\kappa_l-1,n_l}, \end{aligned}$$

while with the notation as in (4.12)

$$\begin{aligned} X &= (\boldsymbol{x}_{1,L}, \boldsymbol{x}_{2,L}, \ldots, \boldsymbol{x}_{T-1,L})^\top \in \mathbb{M}_{T-1,\sum_{l=1}^L N_l}, \\ Y &= (\boldsymbol{y}_2, \ldots, \boldsymbol{y}_T)^\top \in \mathbb{M}_{T-1,J}, \end{aligned}$$

are used for the estimation of $\widehat{W}$. Again, the state equations (4.10) of M-MFESN can be started with $\boldsymbol{x}^{(l)}_{0,0|\kappa_l} = \boldsymbol{0}_{N_l}$ (see Subsection 3 for more details).

Forecasting with aligned M-MFESN. Let $\widehat{W}$ and $\widehat{\mathcal{W}}_l$, $l \in [L]$, be the sample estimates of the readout matrices. For any $l \in [L]$ the κ_l-frequency autonomous state transition map is given by

$$F^{(l)}_{\kappa_l}(\boldsymbol{x}^{(l)}_{t,s-1|\kappa_l}) := \alpha_l \boldsymbol{x}^{(l)}_{t,s-1|\kappa_l} + (1-\alpha_l)\sigma\left((A_l + C_l\widehat{\mathcal{W}}_l^\top)\boldsymbol{x}^{(l)}_{t,s-1|\kappa_l} + \boldsymbol{\zeta}_l\right). \tag{4.13}$$

The target-frequency-aligned autonomous state transition map associated with each frequency l is hence defined as

$$F^{(l)}(\boldsymbol{x}_{t,0|\kappa_l}) := \underbrace{F^{(l)}_{\kappa_l} \circ F^{(l)}_{\kappa_l} \circ \cdots \circ F^{(l)}_{\kappa_l}}_{\kappa_l \text{ times}}(\boldsymbol{x}^{(l)}_{t,0|\kappa_l}). \tag{4.14}$$

Finally, from (3.13) the h-steps ahead forecasts can be computed as

$$\widetilde{y}_{T+h|T} = \widehat{W}^\top \begin{pmatrix} \underbrace{F^{(1)} \circ F^{(1)} \circ \cdots \circ F^{(1)}}_{h-1 \text{ times}}(\boldsymbol{x}^{(1)}_{T,0|\kappa_1}) \\ \vdots \\ \underbrace{F^{(L)} \circ F^{(L)} \circ \cdots \circ F^{(L)}}_{h-1 \text{ times}}(\boldsymbol{x}^{(L)}_{T,0|\kappa_L}) \end{pmatrix}. \tag{4.15}$$

In Figure 3 we provide a diagram for the case of 1-step ahead forecasting with an aligned M-MFESN involving regressors of only two frequencies. Figure 13 in Appendix I provides a similar diagram for the case of multistep forecasting.

Example 4.2 Similar to Example 4.1, we aim to forecast a quarterly target with monthly and daily series, but this time we use an M-MFESN model. We have to define two independent state equations, one for monthly and one for daily series; in the observation equations, two states must be aligned temporally and stacked to form the full set of regressors. The data consists again of quarterly (y_t), $n_{(\mathtt{m})}$ monthly series $(\boldsymbol{z}^{(\mathtt{m})}_{t,s|3})$ and $n_{(\mathtt{d})}$ daily series $(\boldsymbol{z}^{(\mathtt{d})}_{t,s|72})$.

The aligned M-MFESN model with two reservoirs of dimensions $N_{(\mathtt{m})}$ and $N_{(\mathtt{d})}$, respectively, is given by

$$\boldsymbol{x}^{(\mathtt{m})}_{t,s|3} = \alpha_1 \boldsymbol{x}^{(\mathtt{m})}_{t,s-1|3} + (1-\alpha_1)\sigma(A_1 \boldsymbol{x}^{(\mathtt{m})}_{t,s-1|3} + C_1 \boldsymbol{z}^{(\mathtt{m})}_{t,s|3} + \boldsymbol{\zeta}_1), \tag{4.16}$$

$$\boldsymbol{z}^{(\mathtt{m})}_{t,s+1|3} = \mathcal{W}^\top_{(\mathtt{m})} \boldsymbol{x}^{(\mathtt{m})}_{t,s|3} + \boldsymbol{u}^{(\mathtt{m})}_{t,s+1|3}, \tag{4.17}$$

$$\boldsymbol{x}^{(\mathtt{d})}_{t,s|72} = \alpha_2 \boldsymbol{x}^{(\mathtt{d})}_{t,s-1|72} + (1-\alpha_2)\sigma(A_2 \boldsymbol{x}^{(\mathtt{d})}_{t,s-1|72} + C_2 \boldsymbol{z}^{(\mathtt{d})}_{t,s|72} + \boldsymbol{\zeta}_2), \tag{4.18}$$

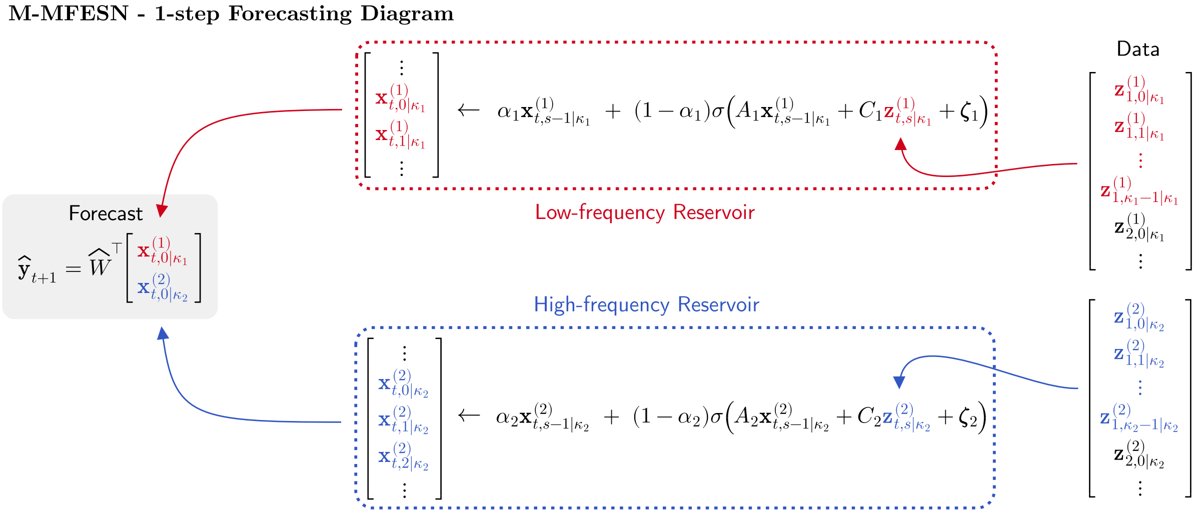

M-MFESN - 1-step Forecasting Diagram
Data
Forecast
$\widehat{\mathbf{y}}_{t+1} = \widehat{W}^\top \begin{bmatrix} \mathbf{x}^{(1)}_{t,0|\kappa_1} \\ \mathbf{x}^{(2)}_{t,0|\kappa_2} \end{bmatrix}$
$\leftarrow \alpha_1 \mathbf{x}^{(1)}_{t,s-1|\kappa_1} + (1-\alpha_1)\sigma\big(A_1\mathbf{x}^{(1)}_{t,s-1|\kappa_1} + C_1\mathbf{z}^{(1)}_{t,s|\kappa_1} + \zeta_1\big)$
Low-frequency Reservoir
High-frequency Reservoir
$\leftarrow \alpha_2 \mathbf{x}^{(2)}_{t,s-1|\kappa_2} + (1-\alpha_2)\sigma\big(A_2\mathbf{x}^{(2)}_{t,s-1|\kappa_2} + C_2\mathbf{z}^{(2)}_{t,s|\kappa_2} + \zeta_2\big)$
$\mathbf{x}^{(1)}_{t,0|\kappa_1}$, $\mathbf{x}^{(1)}_{t,1|\kappa_1}$
$\mathbf{x}^{(2)}_{t,0|\kappa_2}$, $\mathbf{x}^{(2)}_{t,1|\kappa_2}$, $\mathbf{x}^{(2)}_{t,2|\kappa_2}$
$\mathbf{z}^{(1)}_{1,0|\kappa_1}$, $\mathbf{z}^{(1)}_{1,1|\kappa_1}$, $\mathbf{z}^{(1)}_{1,\kappa_1-1|\kappa_1}$, $\mathbf{z}^{(1)}_{2,0|\kappa_1}$
$\mathbf{z}^{(2)}_{1,0|\kappa_2}$, $\mathbf{z}^{(2)}_{1,1|\kappa_2}$, $\mathbf{z}^{(2)}_{1,\kappa_2-1|\kappa_2}$, $\mathbf{z}^{(2)}_{2,0|\kappa_2}$

Figure 3: Scheme of a Multi-Reservoir MFESN (M-MFESN) model combining input data sampled at two frequencies with state alignment and estimation for one-step ahead forecasting of the target series.

$$\boldsymbol{z}^{(\mathtt{d})}_{t,s+1|72} = \mathcal{W}^\top_{(\mathtt{d})}\boldsymbol{x}^{(\mathtt{d})}_{t,s|72} + \boldsymbol{u}^{(\mathtt{d})}_{t,s+1|72}, \tag{4.19}$$

$$y_{t+1} = W^\top \begin{pmatrix} \boldsymbol{x}^{(\mathtt{m})}_{t,0|3} \\ \boldsymbol{x}^{(\mathtt{d})}_{t,0|72} \end{pmatrix} + \boldsymbol{\epsilon}_{t+1}, \tag{4.20}$$

where $s > 0$, $\mathcal{W}_{(\mathtt{m})} \in \mathbb{M}_{N_{(\mathtt{m})},n_{(\mathtt{m})}}$, $\mathcal{W}_{(\mathtt{d})} \in \mathbb{M}_{N_{(\mathtt{d})},n_{(\mathtt{d})}}$ and $W \in \mathbb{R}^{N_{(\mathtt{m})}+N_{(\mathtt{d})}}$. Here, the monthly reservoir ($\boldsymbol{x}^{(\mathtt{m})}_{t,s|3}$) has the temporal index of frequency 3, while the daily reservoir ($\boldsymbol{x}^{(\mathtt{d})}_{t,s|72}$) of 72; the high-frequency index s is different for the two models. Notice that in an M-MFESN model it is necessary to introduce 2 additional observation equations for the states, that is (4.17) and (4.19)). Notice that the state equations are iterated each κ_l times to collect the states to be aligned in the observation equation (4.20). Again, the sample-based estimates of coefficient matrices $\widehat{\mathcal{W}}_{(\mathtt{m})}$, $\widehat{\mathcal{W}}_{(\mathtt{d})}$ and $\widehat{W}$ in (4.17), (4.18), and in (4.20), respectively, can be obtained via the ridge regression as discussed above.

Exactly as in Example 4.1, using (4.13) we can introduce high-frequency autonomous state maps $F^{(\mathtt{m})}_3$ and $F^{(\mathtt{d})}_{72}$ as

$$F^{(\mathtt{m})}_3(\boldsymbol{x}^{(\mathtt{m})}_{t,s-1|3}) := \alpha_1\boldsymbol{x}^{(\mathtt{m})}_{t,s-1|3} + (1-\alpha_1)\sigma\left((A_1 + C_1\widehat{\mathcal{W}}^\top_{(\mathtt{m})})\boldsymbol{x}^{(\mathtt{m})}_{t,s-1|3} + \boldsymbol{\zeta}_1\right),$$
$$F^{(\mathtt{d})}_{72}(\boldsymbol{x}^{(\mathtt{d})}_{t,s-1|72}) := \alpha_2\boldsymbol{x}^{(\mathtt{d})}_{t,s-1|72} + (1-\alpha_2)\sigma\left((A_2 + C_2\widehat{\mathcal{W}}^\top_{(\mathtt{d})})\boldsymbol{x}^{(\mathtt{d})}_{t,s-1|72} + \boldsymbol{\zeta}_2\right),$$

as well as their target-frequency aligned counterparts $F^{(\mathtt{m})}$ and $F^{(\mathtt{d})}$, by (4.14), as

$$F^{(\mathtt{m})}(\boldsymbol{x}^{(\mathtt{m})}_{t,0|3}) := \underbrace{F^{(\mathtt{m})}_3 \circ F^{(\mathtt{m})}_3 \circ F^{(\mathtt{m})}_3}_{3 \text{ times}}(\boldsymbol{x}^{(\mathtt{m})}_{t,0|3}),$$
$$F^{(\mathtt{d})}(\boldsymbol{x}^{(\mathtt{d})}_{t,0|72}) := \underbrace{F^{(\mathtt{d})}_{72} \circ F^{(\mathtt{d})}_{72} \circ \cdots \circ F^{(\mathtt{d})}_{72}}_{72 \text{ times}}(\boldsymbol{x}^{(\mathtt{d})}_{t,0|72}).$$

The h-step ahead forecasts can be computed using the approximation in (4.15) as

$$\widetilde{y}_{T+h|T} = \widehat{W}^\top \begin{pmatrix} \underbrace{F^{(\mathtt{m})} \circ F^{(\mathtt{m})} \circ \cdots \circ F^{(\mathtt{m})}}_{h-1 \text{ times}} (\boldsymbol{x}^{(\mathtt{m})}_{T,0|3}) \\ \underbrace{F^{(\mathtt{d})} \circ F^{(\mathtt{d})} \circ \cdots \circ F^{(\mathtt{d})}}_{h-1 \text{ times}} (\boldsymbol{x}^{(\mathtt{d})}_{T,0|72}) \end{pmatrix}.$$

In this case, it is important to note that while both $F^{(\mathtt{m})}$ and $F^{(\mathtt{d})}$ are composed $h-1$ times at step h, the underlying number of autonomous reservoir iterations is different for the monthly and daily reservoirs, namely 3 and 72, and depends on their own frequencies. This also suggests that one should take into account the different time dynamics when, for example, tuning M-MFESN hyperparameters $\boldsymbol{\varphi}^{(\mathtt{m})}$ and $\boldsymbol{\varphi}^{(\mathtt{d})}$, as proposed in Subsection B.2.

5 Empirical Study

In this section, we compare the forecasting performance of our proposed MFESN to state-of-the-art benchmarks. We use a combination of macroeconomic and financial data sampled at low and high-frequency intervals, respectively. Our empirical exercises encompass several setups, with a small and a medium-sized set of regressors, fitting models with data before and after the 2007-08 crisis, and with fixed, rolling, and expanding estimation windows.

5.1 Data

Two sets of predictors of different sizes are compiled: Small-MD with 9 predictors and Medium-MD with 33 predictors in monthly and daily frequency. The reference frequency is quarterly: this is the frequency at which the target variable, US GDP growth, is available. Seasonally adjusted quarterly and monthly data is obtained from the Federal Reserve Bank of St. Louis Monthly (FRED-MD) and Quarterly (FRED-QD) Databases for Macroeconomic Research (see McCracken and Ng (2016, 2020) for detail). Daily data is obtained from Refinitiv Datastream, a subscription-based data service. All data is the last revised vintage data. The macroeconomic target and predictors, their transformations, and availability are listed in Table 5.1.

The selection of predictors follows the seminal work by Stock and Watson (1996, 2006) in which the FRED-MD and FRED-QD data are proposed. Variations of their dataset have been used profusely in the literature (for example, see Boivin and Ng (2005), Marcellino et al. (2006), Hatzius et al. (2010)). Indicators from ten macroeconomic and financial categories are considered: (1) output and income, (2) labor market, (3) housing, (4) orders and inventories, (5) price indices, (6) money and credit, (7) interest rates, (8) exchange rates, (9) equity, and (10) derivatives. The latter five categories represent financial market conditions and are sourced at daily frequency. The exception is interest rates, which move relatively slowly and enter as monthly aggregates, available in the FRED-MD data. We refer to this dataset as Medium-MD. A subset of predictors is selected for the Small-MD dataset by choosing variables that have been identified as leading indicators in the empirical literature (Ingenito and Trehan, 1996, Clements and Galvão, 2008, Andreou et al., 2013, Marsilli, 2014, Ferrara et al., 2014, Carriero et al., 2019, Jardet and Meunier, 2022). Data availability is an additional criterion, and predictors unavailable before 1990 are not considered. This excludes the VIX volatility index, which has been identified as a leading indicator in some studies, for example in Andreou et al. (2013), Jardet and Meunier (2022).

We follow instructions by McCracken and Ng (2016, 2020) on pre-processing macroeconomic predictors before they are used as input for forecasting. These are mainly differenced for detrending. We further transform financial predictors to capture market disequilibrium and volatility. Disequilibrium indicators, such as interest rate spreads, have been found to be more relevant for macroeconomic

Table 5.1: Variables, Frequencies and Transformations for Small and Medium

S	M	Start Date	T	Code	Name	Description
Quarterly						
X	X	31/03/1959	5	GDPC1	Y	Real Gross Domestic Product
Monthly						
X	X	30/01/1959	5	INDPRO	XM1	Industrial Production Index
X	X	30/01/1959	5	PAYEMS	XM4	Payroll All Employees: Total nonfarm
X	X	30/01/1959	4	HOUST	XM5	Housing Starts: Total New Privately Owned
X	X	30/01/1959	5	RETAILx	XM7	Retail and Food Services Sales
X	X	31/01/1973	5	TWEXMMTH	XM11	Nominal effective exchange rate US
X	X	30/01/1959	2	FEDFUNDS	XM12	Effective Federal Funds Rate
X	X	30/01/1959	1	BAAFFM	XM14	Moody's Baa Corporate Bond Minus FEDFUNDS
X	X	30/01/1959	1	COMPAPFFx	XM15	3-Month Commercial Paper Minus FEDFUNDS
	X	30/01/1959	2	CUMFNS	XM2	Capacity Utilization: Manufacturing
	X	30/01/1959	2	UNRATE	XM3	Civilian Unemployment Rate
	X	30/01/1959	5	DPCERA3M086SBEA	XM6	Real personal consumption expenditures
	X	30/01/1959	5	AMDMNOx	XM8	New Orders for Durable Goods
	X	31/01/1978	2	UMCSENTx	XM9	Consumer Sentiment Index
	X	30/01/1959	6	WPSFD49207	XM10	PPI: Finished Goods
	X	30/01/1959	1	AAAFFM	XM13	Moody's Aaa Corporate Bond Minus FEDFUNDS
	X	30/01/1959	1	TB3SMFFM	XM16	3-Month Treasury C Minus FEDFUNDS
	X	30/01/1959	1	T10YFFM	XM17	10-Year Treasury C Minus FEDFUNDS
	X	30/01/1959	2	GS1	XM18	1-Year Treasury Rate
	X	30/01/1959	2	GS10	XM19	10-Year Treasury Rate
	X	30/01/1959	1	GS10-TB3MS	XM20	10-Year Treasury Rate - 3-Month Treasury Bill
Daily						
X	X	30/01/1959	8	DJINDUS	XD3	DJ Industrial price index
	X	31/12/1963	8	S&PCOMP	XD1	S&P500 price index
	X	01/05/1982	1	ISPCS00-S&PCOMP†	XD2	S&P500 basis spread
	X	11/09/1989	8	SP5EIND	XD4	S&P Industrial price index
	X	31/12/1969	8	GSCITOT	XD5	Spot commodity price index
	X	10/01/1983	8	CRUDOIL	XD6	Spot price oil
	X	02/01/1979	8	GOLDHAR	XD7	Spot price gold
	X	30/03/1982	8	WHEATSF	XD8	Spot price wheat
	X	01/11/1983	8	COCOAIC,COCINUS‡	XD9	Spot price cocoa
	X	30/03/1983	1	NCLC.03-NCLC.01	XD10	Futures price oil term structure
	X	30/10/1978	1	NGCC.03-NGCC.01	XD11	Futures price gold term structure
	X	02/01/1975	1	CWFC.03-CWFC.01	XD12	Futures price wheat term structure
	X	02/01/1973	1	NCCC.03-NCCC.01	XD13	Futures price cocoa term structure

Notes: S and M stand for small and medium datasets, respectively. An 'X' indicates selection into the dataset. 'Start Date' is the date for which the series is first available (before data transformations). Following McCracken and Ng (2016, 2020), the transformation codes in column 'T' indicate with D for difference and log for natural logarithm 1: none, 2: D, 3: DD, 4: Log, 5: Dlog, 6: DDlog, 7: percentage change, 8: GARCH volatility. 'Codes' are the codes in the FRED-QD and FRED-MD datasets for quarterly and monthly data and Datastream mnemonic for the remaining frequencies. Missing values due to public holidays are interpolated by averaging over the previous five observations. ‡Available until 20/09/2021. ‡Average before 29/12/2017, COCINUS mean adjusted thereafter.

prediction than routine changes captured by differencing (see Borio and Lowe (2002), Gramlich et al. (2010), Qin et al. (2022)). In addition to disequilibrium indicators, realized stock market volatility has been found to improve macroeconomic predictions (Chauvet et al., 2015). In the absence of intraday trading data from the 1990s onward, which prevents us from utilizing conventional daily realized volatility indicators, we extract volatility indicators from daily price series by fitting a GARCH(1,1) by Bollerslev (1986).[1] In addition to volatility of stock and commodity prices, term structure indi-

[1] We include a control `scale = 1` to ensure convergence of the optimization algorithm and only include a constant mean term in the return process for simplicity.

cators are used. The term structure is forward-looking, capturing information about future demand and supply, and has been found to be a leading predictor of GDP growth (see for example Hong and Yogo (2012), Kang and Kwon (2020)).

The data spans the period January 1st, 1990 to December 31st, 2019.[2] We are interested in evaluating model performance under two stylized settings. First, a researcher fits all models up until the Great Recession, including data from Q1 1990 to Q4 2007. Second, fitting is done with data largely encompassing the crisis period, again from Q1 1990 but now up to Q4 2011. In both cases, the testing sample ranges from the next GDP growth observation after fitting up to Q4 2019. All exercises exclude the global COVID-19 economic depression, as we consider it as an extreme, unpredictable event that induces significant structural changes in the underlying macroeconomic dynamics.[3]

To avoid having to handle the many edge cases that daily data in its "raw" calendar releases involves, we use an interpolation approach. We set *ex ante* the number of working days in *any* month to be exactly 24: given that in forecasting the most recent information sets are more relevant, when interpolating daily data over months with less than calendar 24 observations, we linearly interpolate the "missing" data starting from a months' beginning (using the previous months' last observation). The choice of 24 as a daily frequency is transparent by noting that this is the closest number to actual commonly observed data releases, whilst also being a multiple of both 4 (approximate number of weeks per month) and 6 (upper bound on the number of working days per week).

5.2 Models

In this section, we present the set of models that we use throughout our empirical exercises. For a general overview, Table 5.2 summarizes all models, including hyperparameters. In our analysis, we compare the competing models based on several performance measures introduced in Appendix C.

5.2.1 Benchmarks

Unconditional mean. We use the unconditional mean of the sample used for fitting as a baseline benchmark. For GDP growth forecasting, there is evidence that the unconditional mean produces forecasts that are competitive with linear models such as VARs in terms of mean square forecasting errors (MSFE), even at relatively short horizons (Arora et al., 2013). It is therefore an important reference for the performance of all other models and we report relative MSFE with respect to the unconditional mean in the tables below.

AR(1) model. A simple autoregressive process of order one on the target variable is included as a benchmark model.[4] This is also a common benchmark in the literature, as AR(1) models are often able to capture key dynamics and produce meaningful forecasts for macroeconomic variables (Stock and Watson, 2002, Bai and Ng, 2008). We emphasize that since AR(1) model is fit to the series of quarterly GDP targets and does not use any additional information, its forecasts are identical for both the Small-MD and Medium-MD samples.

MIxed DAta Sampling (MIDAS). The first mixed-frequency model benchmark is given by a MIDAS model (Ghysels et al., 2004, 2007). Our dynamic MIDAS specification includes autoregressive

[2] In the Small-MD dataset experiments we make a small variation and instead include data starting from 1st January 1975, but *only* for the initial CV selection of ridge penalties for MFESN models. Our aim is to make sure that at least for the fixed window estimation strategy – where λ is cross-validated once and only one $\widehat{W}$ is estimated – the ridge estimator is robust. In practice, when we compare to expanding and rolling window estimators, where λ is re-selected at each window, we find that extending the initial CV data window has little impact on out-of-sample performance.

[3] In the macroeconomic literature this falls under the category of "natural disaster" events, and should not be naïvely modeled together with previous observations. In this section, we therefore avoid dealing with post-COVID-19 macroeconomic data altogether.

[4] Suggested by an anonymous referee.

Model Name	Description	Specification
Mean	Unconditional mean of target series over estimation sample.	None
AR(1)	Autoregressive model of target series estimated using OLS.	None
MIDAS	Almon-weighted MIDAS regression, linear (unconstrained) autoregressive component.	Autoregressive lags: 3 Monthly freq. lags: 9 Daily freq. lags: 30
DFM [A]	Stock aggregation, VAR(1) factor process.	Factors: 5 for Small-MD 10 for Medium-MD
DFM [B]	Almon aggregation, VAR(1) factor process	Factors: 5 for Small-MD 10 for Medium-MD
singleESN [A]	S-MFESN model: Sparse-normal $\widetilde{A}$, sparse-uniform $\widetilde{C}$, $\widetilde{\boldsymbol{\zeta}} = \mathbf{0}$. Isotropic ridge regression fit.	Reservoir dim: 30 Sparsity: 33.3% $\rho = 0.5$, $\gamma = 1$, $\alpha = 0.1$
singleESN [B]	S-MFESN model: Sparse-normal $\widetilde{A}$, sparse-uniform $\widetilde{C}$, $\widetilde{\boldsymbol{\zeta}} = \mathbf{0}$. Isotropic ridge regression fit.	Reservoir dim: 120 Sparsity: 8.3% $\rho = 0.5$, $\gamma = 1$, $\alpha = 0.1$
multiESN [A]	M-MFESN model: Monthly and daily frequency reservoirs. Sparse-normal $\widetilde{A}_1$, $\widetilde{A}_2$, sparse-uniform $\widetilde{C}_1$, $\widetilde{C}_2$, $\widetilde{\boldsymbol{\zeta}}_1 = \mathbf{0}$, $\widetilde{\boldsymbol{\zeta}}_2 = \mathbf{0}$. Isotropic ridge regression fit.	Reservoir dims: M=100, D=20 Sparsity: M=10%, D=50% M: $\rho = 0.5$, $\gamma = 1.5$, $\alpha = 0$ D: $\rho = 0.5$, $\gamma = 0.5$, $\alpha = 0.1$
multiESN [B]	M-MFESN model: Monthly and daily frequency reservoirs. Sparse-normal $\widetilde{A}_1$, $\widetilde{A}_2$, sparse-uniform $\widetilde{C}_1$, $\widetilde{C}_2$, $\widetilde{\boldsymbol{\zeta}}_1 = \mathbf{0}$, $\widetilde{\boldsymbol{\zeta}}_2 = \mathbf{0}$. Isotropic ridge regression fit.	Reservoir dims: M=100, D=20 Sparsity: M=10%, D=50% M: $\rho = 0.08$, $\gamma = 0.25$, $\alpha = 0.3$ D: $\rho = 0.01$, $\gamma = 0.01$, $\alpha = 0.99$

Table 5.2: Table of models used in applied forecasting exercises. MFESN hyperparameters are defined with respect to normalized state parameters c.f. (3.6).

lags of the target series and uses an Almon weighting scheme. As shown in Bai et al. (2013), exponential Almon MIDAS regressions are related to dynamic factor models, which we also consider as benchmarks. The MIDAS model includes three lags of quarterly GDP target variable, and 30 daily and 9 monthly lags for all daily and monthly series, respectively. This model prescription allows for some parsimony as the Almon polynomial weighing reduces the number of daily and monthly lag coefficients.

A thorough description of our MIDAS implementation can be found in Appendix E. To make optimization more efficient, we use explicit expressions for MIDAS loss gradients as in Kostrov (2021). The MIDAS estimation can be hard to perform in practice due to the complexity of nonlinear optimization. First, exponential weighting schemes might require computing floating-point numbers that exceed numerical precision. Therefore, it is a better choice to start the gradient descent close to the origin of the parameter space. Second, even with this choice of starting points, one may encounter issues with optimization results since the Almon-scheme MIDAS loss can have a large number of distinct local minima. In Appendix H.1 we document, using a simple replication experiment, that even small changes in the initial conditions can result in different local minima picked by the numerical optimization algorithm.[5] These important robustness issues are present even when using closed-form gradients and multi-start optimization routines for the MIDAS models. The computational issues become more pronounced as the number of MIDAS parameters increases unless a careful model/variable selection step is performed. We, therefore, do not include any MIDAS model specifications in the Medium-MD setup.

Dynamic Factor Model (DFM). The dynamic factor model framework has been extensively applied in macroeconometrics, starting with Geweke (1977) and Sargent et al. (1977). A DFM specification assumes that predictable dynamics of a large set of time series can be explained by a small number of factors with an autoregressive dependence (see for example Forni et al. (2005), Doz et al. (2011), Stock and Watson (2016)). We generalize the standard two-frequency DFM modeling setup (Mariano and Murasawa, 2003, Bańbura and Modugno, 2014) to a flexible mixed-frequency DFM that encompasses any number of data frequencies. Moreover, we derive a novel weighting scheme that effectively links the MIDAS and DFM approaches. For a detailed discussion of our factor model setup, we refer the reader to Appendix F. Two distinct DFM specifications are used. The first one termed DFM [A] uses the standard linear aggregation scheme, as provided in Example F.1, while the second is a variation that implements an Almon weighting scheme as presented in Example F.2 (we name it DFM [B]). The latter is similar to a MIDAS-type aggregation scheme (Marcellino and Schumacher, 2010): the factor structure effectively mitigates the parameter proliferation.

A key choice for a DFM model is the dimension of the factor process. While a number of methods have been developed over the years to systematically derive the number of factors (see, for example, the review of Stock and Watson (2016)), commonly used macroeconomic panels feature a number of challenges, such as weak factors (Onatski, 2012). Moreover, as mentioned in Appendix F.1, factor number selection in the mixed-frequency setting has not been sufficiently addressed in the literature. To sidestep these issues, we construct both DFM models with 5 unobserved factors for Small-MD and 10 for Medium-MD, respectively, and assume that they follow a VAR(1) process.

One extant issue with integrating daily data is its very high release frequency compared to monthly and especially quarterly releases: computationally this can be extremely taxing, which might be one of the reasons why to our knowledge we are *the first to provide DFM forecasts that include daily data.* Our solution is to reduce aggregate daily data every 6 days by averaging, thus leaving 4 observations per month. This eases the computational burden to estimate coefficients and latent states considerably (12 versus 72 daily observations per quarter).

5.2.2 Multi-Frequency ESNs

The first set of ESNs we propose is given by two S-MFESN models, based on Example 4.1. One model uses a reservoir of 30 neurons (we call it singleESN [A]); the other has a larger reservoir of dimension 120 (named singleESN [B]). The sparsity degree of state parameters for both models is set to be $10/N$, where N is the reservoir size. Both MFESNs share the same hyperparameters, $\rho = 0.5$, $\gamma = 1$, $\alpha = 0.1$ (see (3.7)). These values have not been tuned but are presumed credible given other ESN implementations in the literature. To make a fair comparison with DFMs, we fit the S-MFESN models using 6-day-averaged daily data. Note here that for MFESN models the computational gains of averaging are negligible, and are most apparent when tuning the ridge penalty via cross-validation.

Our second set of proposed models consists of two M-MFESNs according to Example 4.2. Both models have two reservoirs, one for each data frequency – monthly and daily – with 100 and 20 neurons, respectively; sparsity degrees are again adjusted to be $10/N$, where N is the reservoir state dimension. The first M-MFESN has hyperparameters that are hand-selected among reasonable values: we note

[5] We set the initial coefficient values to zero in all empirical exercises.

Execution Time (Seconds) for Model Estimation

				DFM		singleESN		multiESN	
Dataset	Mean	AR(1)	MIDAS	[A]	[B]	[A]	[B]	[A]	[B]
Small-MD	0.1	0.7	1.3	40.5	85.5	2.6	4.5	15.3	14.6
Medium-MD	0.1	0.8	–	48.0	226.5	2.5	5.7	17.7	14.7

Table 5.3: Execution time in seconds for model estimation measured over a single run on a quad-core computer. MFESN models timing includes ridge penalty cross-validation. MIDAS estimation time refers to optimization from a single initial value. DFM models were estimated on a single-core server and times are adjusted by a factor of 1/4 for comparison.

that the monthly-frequency reservoir has no state leak and a larger input scaling, while the daily frequency reservoir features smaller scaling than usual (to avoid compressing high volatility events with the activation function) and the same leak rate as in S-MFESN models (we call this specification multiESN [A]). For the second M-MFESN, we change hyperparameters more radically: we aim to set up a model that has a very high input memory (Ballarin et al., 2023), and that also features long-term smoothing of states. Note that here input scaling values are small, spectral radii are an order of magnitude smaller than in previous models, and leak rates are large (we term this model multiESN [B]).

5.3 Results

We start by commenting on the computational efficiency of competing models and report execution times in seconds in Table 5.3. Firstly, DFM models appear to be the most computationally effortful models among all specifications. For the Small-MD dataset, the simplest MFESN models, that is, singleESN [A] and [B], have execution times which are at most 3.5 times higher than the MIDAS model, while still being at least 15.6 times computationally cheaper than any of the DFM models. The more resource-demanding models MFESN, multiESN [A] and [B], are nevertheless at least 2.6 times faster to run than the best DFM model (DFM [A]). When moving to the Medium-MD dataset, where the MIDAS model is not, as explained earlier, a feasible choice, the most inefficient MFESN model (singleESN [B]) still outperforms the best DFM model, DFM [A], by 8.4 times, while the same holds for multiESN [A] model versus DFM [A] model by 2.7 times. We can conclude that our proposed MFESN architectures provide an attractive and computationally efficient framework for GDP forecasting in the multifrequency framework which is feasible for computations on low-cost machine configurations available to practitioners.

Competing forecasts are compared using the Model Confidence Set (MCS) test derived in Hansen et al. (2011). One should note that due to the intrinsic nature of data availability of macroeconomic time series and panels, our sample sizes are modest. This implies that the small sample sensitivities of the MCS test need to be taken into account when evaluating our comparisons. Recent analyses of the finite sample properties of the MCS methodology have shown that it requires signal-to-noise ratios which are unattainable in most empirical settings, an issue that undermines its applicability (Aparicio and de Prado, 2018). Given this fact, we also conduct pairwise model comparison tests with the Modified Diebold-Mariano (MDM) test for predictive accuracy (Diebold and Mariano, 2002, Harvey et al., 1997).

As we also provide multiple-steps-ahead forecasts, we test for the best subset of models uniformly across all horizons using the Uniform Multi-Horizon MCS (uMCS) test proposed by Quaedvlieg (2021). Since there is relatively little systematic knowledge regarding the power properties of the uMCS test in small samples, our inclusion of this procedure is meant as a statistical counterpoint to simple relative

1-Step-ahead GDP Forecasting - Small-MD Dataset

Model	Fixed Parameters				Expanding Window				Rolling Window			
	2007		2011		2007		2011		2007		2011	
	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS
Mean	1.000	*	1.000	**	1.000	**	1.000	**	1.000	**	1.000	**
AR(1)	0.758	*	1.230	**	0.789	**	1.226	**	0.775	**	1.209	**
MIDAS	**0.533**	**	1.300		0.596	**	1.129	*	0.709	**	1.170	*
DFM [A]	0.799	*	1.337		0.980	*	1.320		0.919	*	1.226	
DFM [B]	0.885		1.221	**	0.982	*	1.022	**	0.948		1.028	**
singleESN [A]	0.721	**	1.015	**	0.597	**	0.867	**	**0.529**	**	0.863	**
singleESN [B]	0.758	*	**0.921**	**	0.602	**	**0.844**	**	0.561	**	0.930	**
multiESN [A]	0.802	*	1.250		0.635	**	0.874	**	0.621	**	**0.859**	**
multiESN [B]	0.590	**	0.969	**	**0.552**	**	0.895	**	0.530	**	0.921	**

Table 5.4: Relative MSFE and Model Confidence Set (MCS) comparison between models in 1-step-ahead forecasting exercises. Unconditional mean MSFE is used as a reference. MCS columns show inclusion among best models: $*$ indicates inclusion at 90%, $**$ indicates inclusion at 75% confidence.

forecasting error comparisons, which provide limited information about the significance of performance differences. We provide more details on our implementation of the test in Appendix D. Finally, we do not report uMCS test outcomes for the expanding window setup, as Quaedvlieg (2021) argues that in such context the test is invalid.

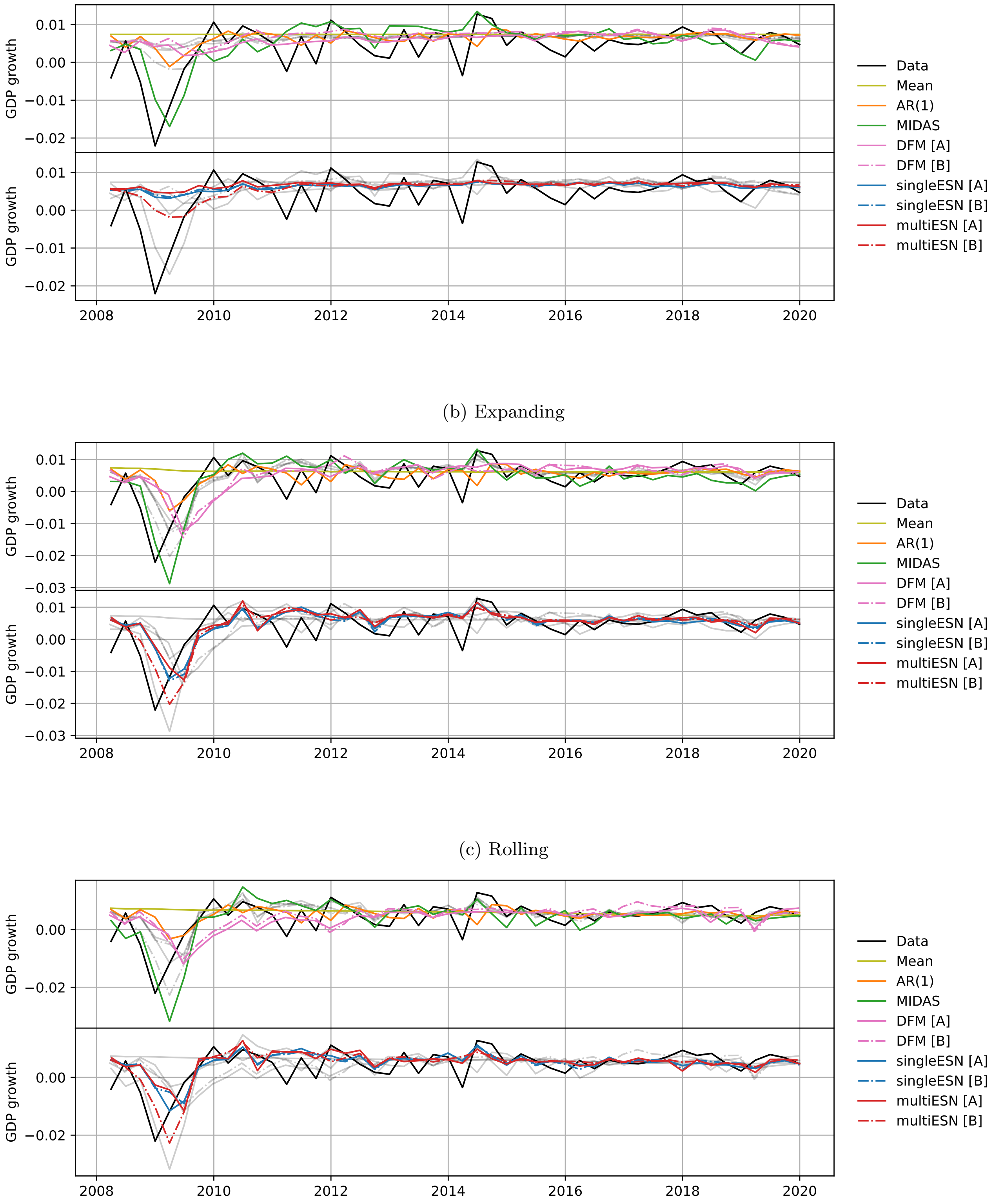
5.3.1 Small Dataset

We begin our discussion of the Small-MD forecasting results by reviewing Table 5.4. For both sample setups (2007 and 2011) and all three estimation strategies (fixed, expanding, and rolling windows) we provide relative MSFE metrics, with the unconditional mean being used as a reference. Plots of each of the model's forecasts are given in Figures 4 and 5; additional plots for cumulative SFE, cumulative RMSFEs and other metrics can be found in Appendix I.

The overall finding is that MFESN models perform very well, and, when we exclude the 2007 fixed parameters setup, they perform the best. It is easy to see from Figure 4 (a) why the 2007 fixed window estimation case is different from other cases: the 2008 Financial Crisis induced a deep drop in quarter-to-quarter GDP growth that was in stark contrast with previous business cycle fluctuations. By keeping model parameters fixed, and using only information from 1990 to 2007 – periods where systematic fluctuations are small – DFM and MFESN models are fit to produce smooth, low-volatility forecasts. MIDAS, on the other hand, yields an exponential smoothing which can be more responsive to changes in monthly and daily series. From Figure 4 (b) and (c) it is possible to see that expanding and rolling window estimation resolves this weakness of state-space models. At the same time, the AR(1) model outperforms the unconditional mean only in the 2007 sample with fixed parameters, losing to the MIDAS model in all but one scenario.

Table 5.4 shows that MFESN models always perform better than the mean in terms of MSFE, something which no other model class achieves across all setups. In both expanding and rolling window setups they also always outperform the AR(1) model. Furthermore, at least one MFESN model for each subclass (single or multi-reservoir) is always included in the model confidence set at the highest confidence level. We remind again that the MCS test of Hansen et al. (2011) might be distorted due to the modest sample sizes considered, even more so in the 2011 test sample. To complement the MCS, we provide graphical tables for pairwise Modified Diebold-Mariano (MDM) tests, with 10%

Figure 4: 1-Step-ahead GDP Forecasting – 2007 Sample – Small-MD Dataset

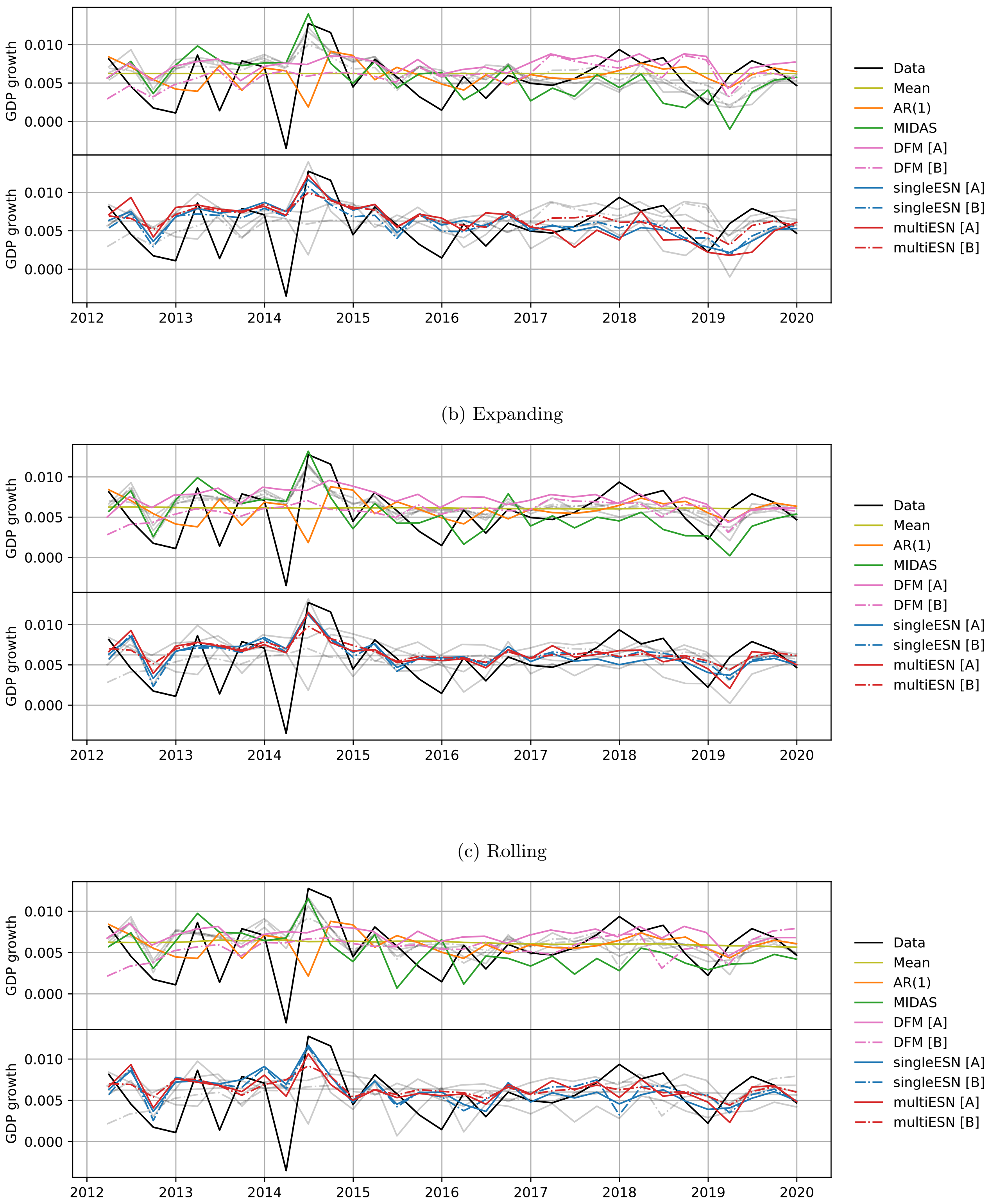


(a) Fixed

(b) Expanding

(c) Rolling

Figure 5: 1-Step-ahead GDP Forecasting – 2011 Sample – Small-MD Dataset



(a) Fixed

(b) Expanding

(c) Rolling

level rejections highlighted in Figure 14, Appendix I. The MDM tests broadly agree with the results of Table 5.4, although they do not account for multiple testing, and therefore cannot be interpreted as yielding subsets of the most accurate forecasting models in a statistical sense.

For multiple-steps-ahead forecasts, relative RMSFE and uMCS are reported in Tables 5.5 and 5.6: we constrain our exercise to $h \in \{1, \ldots, 8\}$ steps, since we are interested in GDP growth forecasts within 2 years. Note that for $h = 1$ our results are similar, but do not reduce to the one-step-ahead results. To make correct multistep RMSFE evaluations and execute the uMCS procedure one must select h different vectors of residuals of the same length: this implies that residuals at the end of the forecasting sample must be trimmed off to compute short-term multistep RMSFEs that are comparable to the long-term ones. We focus our discussion on Figures 6 and 7, which reproduce the RMSFE numbers of the aforementioned tables graphically. Generally, we notice that MIDAS, as well as S-MFESNs, provide the worst-performing multistep forecasts, with RMSFEs considerably exceeding the unconditional mean baseline after horizon 1.

For MIDAS, we have already discussed how the existence of multiple loss minima can generate numerical instabilities. Model re-fitting at each horizon can amplify this problem, as the loss landscape itself changes as new observations are added to the fitting sample. We provide more discussions in Appendix H.1. In the case of S-MFESN models, the reason is structural: we have discussed how in our framework multistep MFESN forecasting entails iterating the state map, which can have multiple attraction (stable) points. If the hyperparameters and estimated full model $\widehat{W}$s jointly do not define a contraction, the limit of the multistep forecast does not have to be the estimated MFSEN model intercept. However, Figures 6 and 7 show that our M-MFESN models, multiESN [A] and multiESN [B], both perform on par or better than DFM models even after horizon $h = 4$. For example, in the 2007 expanding and rolling window experiments, multiESN [B] is able to outperform both DFMs and an unconditional mean forecast by meaningful margins for forecasts up to a year into the future.

5.3.2 Medium Dataset

We now present the results for the Medium-MD dataset, which includes more than 30 regressors and many high-frequency daily series. The same metrics as in the previous subsection are used for this dataset to evaluate the relative performance of different methods.

The main difference in our empirical exercises is that now we a priori exclude MIDAS from the set of forecasting methods as explained in detail in Subsection 5.2.1. Table 5.7 showcases the relative performance of DFM and MFESN models in the Medium-MD forecast setup. We find that the MFESN model multiESN [B] performs best in all setups, particularly under fixed parameters, where MCS testing reveals that it is the only model included at a 75% confidence level. Of course, for the MCS results we must again take into account the relatively small sample size, which could distort the selection of best model subsets. MDM tests of Figure 16 in Appendix I largely agree with the MCS results: in the fixed parameter setup any pairwise comparison of an alternative model against MFESN multiESN [B] is rejected in favor of the latter. A visual inspection of one-step-ahead forecasts in Figures 8 and 9 also shows that DFM models estimated over the Medium-MD datasets produce forecasts with larger variability than MFESN methods, which is likely the key driver of the difference in performance.

The multistep-ahead experiments are run as for the Small-MD dataset, with a maximum horizon of 8 quarters. Tables 5.8 and 5.9 present the relative RMSFE performance of multistep forecasts for all models, and we use Figures 10 and 11 of RMSFEs as references for our discussion. What can be seen visually – and is also reproduced in the Tables – is that multi-reservoir MFESN models and DFM model [A] have the best performance up to 4 quarters ahead; overall, taking into account also the longer term, expanding or rolling window estimation of model multiESN [B] yields the best forecasting results in the 2007 sample setup. The post-crisis 2011 sample setup makes comparison harder, as DFM and M-MFESN models largely produce results in line with the unconditional sample

Multistep-ahead GDP Forecasting - Small-MD Dataset - 2007 Sample

Setup	Model	Horizon								uMCS
		1	2	3	4	5	6	7	8	
FIX	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	
FIX	AR(1)	0.870	**0.950**	**0.982**	0.991	0.992	0.991	0.992	0.992	**
FIX	MIDAS	0.823	1.672	2.737	1.816	2.213	2.791	1.888	1.921	
FIX	DFM [A]	0.890	0.969	1.014	1.077	1.341	1.701	2.001	2.180	*
FIX	DFM [B]	0.937	1.069	1.202	1.344	1.799	2.310	2.638	2.801	
FIX	singleESN [A]	0.852	0.994	0.995	0.995	0.993	0.991	0.991	0.991	*
FIX	singleESN [B]	0.871	0.986	0.989	**0.989**	**0.985**	**0.981**	**0.981**	**0.981**	**
FIX	multiESN [A]	0.898	0.980	0.990	0.991	0.988	0.985	0.985	0.985	**
FIX	multiESN [B]	**0.767**	0.954	0.983	0.991	0.991	0.990	0.991	0.991	**
EW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	**1.000**	**1.000**	-
EW	AR(1)	0.887	0.922	0.951	0.962	0.957	**0.981**	1.001	1.008	-
EW	MIDAS	0.814	1.283	1.518	1.596	1.697	1.391	1.951	1.800	-
EW	DFM [A]	0.985	1.109	1.123	1.114	1.217	1.226	1.241	1.539	-
EW	DFM [B]	0.989	1.082	1.149	1.199	1.315	1.412	1.373	1.425	-
EW	singleESN [A]	0.771	1.260	1.485	1.564	2.070	2.728	2.550	2.834	-
EW	singleESN [B]	0.772	1.031	1.135	1.319	1.831	2.279	2.449	2.556	-
EW	multiESN [A]	0.792	0.897	0.941	0.976	1.015	1.240	1.377	1.227	-
EW	multiESN [B]	**0.740**	**0.853**	**0.894**	**0.911**	**0.873**	0.993	1.020	1.020	-
RW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	*
RW	AR(1)	0.898	0.943	0.968	0.974	0.963	**0.968**	0.970	**0.962**	**
RW	MIDAS	0.933	1.438	1.642	1.993	1.794	1.661	1.816	1.973	*
RW	DFM [A]	0.931	1.017	1.033	1.020	1.024	1.003	**0.918**	1.062	*
RW	DFM [B]	0.942	0.973	0.970	1.045	1.059	1.203	1.225	1.263	*
RW	singleESN [A]	**0.714**	1.320	1.693	1.972	2.733	3.669	3.391	3.719	*
RW	singleESN [B]	0.737	1.100	1.248	1.667	2.327	2.765	2.842	2.792	*
RW	multiESN [A]	0.773	0.972	1.053	1.111	1.187	1.293	1.505	1.131	*
RW	multiESN [B]	0.716	**0.895**	**0.916**	**0.926**	**0.890**	1.041	1.102	1.105	**

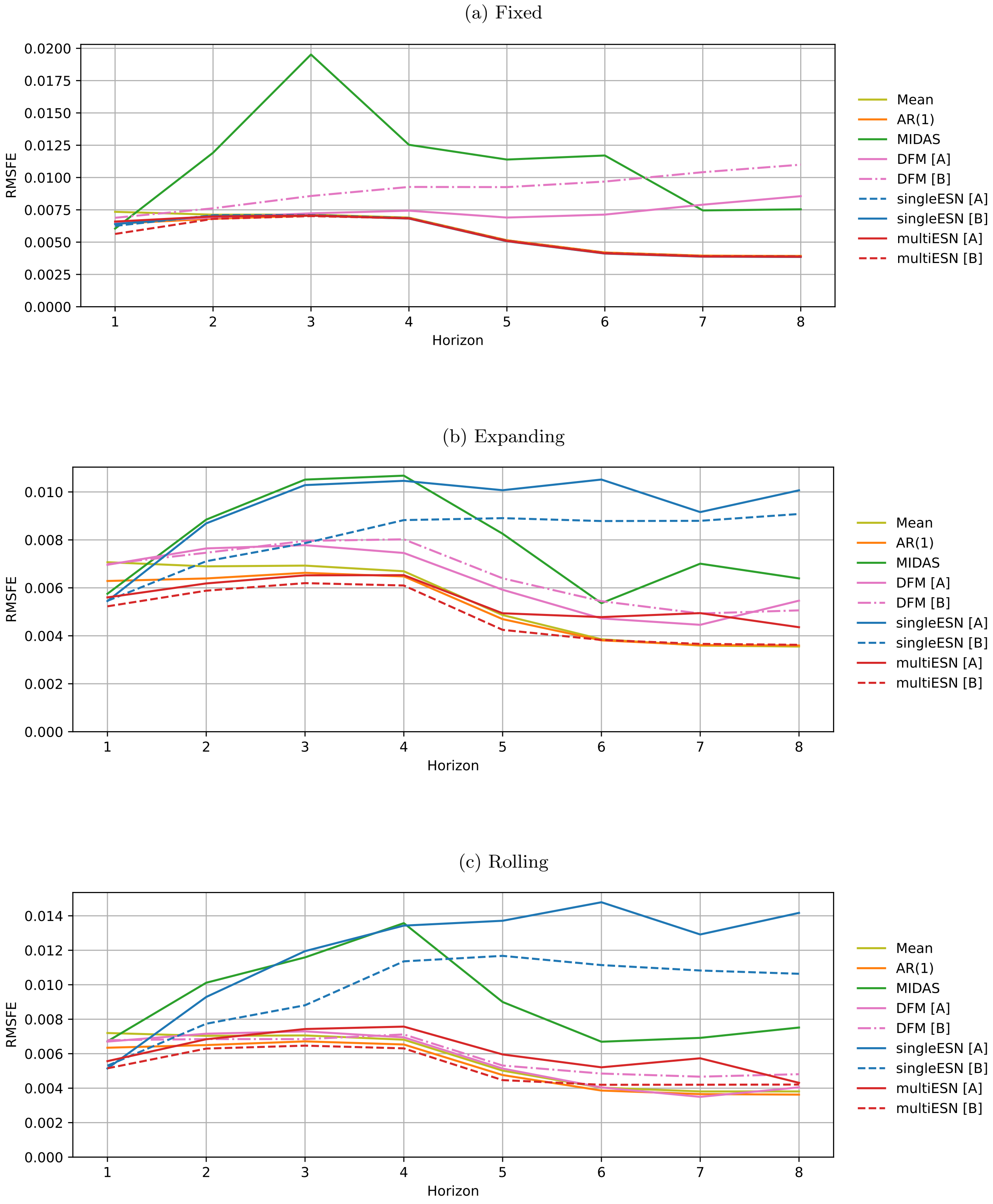
Table 5.5: Relative RMSFE and Uniform Multi-Horizon Model Confidence Set (uMCS) comparison between models in multiple-steps-ahead forecasting exercises. Unconditional mean RMSFE used as reference. FIX: Fixed parameters, EW: Expanding window, and RW: Rolling window. uMCS columns show inclusion among best models: $*$ indicates inclusion at 90% confidence, $**$ indicates inclusion at 75% confidence.

Multistep-ahead GDP Forecasting - Small-MD Dataset - 2011 Sample

Setup	Model	Horizon								uMCS
		1	2	3	4	5	6	7	8	
FIX	Mean	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	**
FIX	AR(1)	1.119	1.031	1.008	1.001	1.001	0.999	**0.999**	**0.998**	*
FIX	MIDAS	1.090	1.721	1.793	2.203	2.363	1.997	2.846	2.328	
FIX	DFM [A]	1.112	1.051	0.999	1.079	1.084	1.025	1.020	1.061	*
FIX	DFM [B]	1.058	**0.945**	**0.916**	1.003	1.012	**0.970**	1.038	1.033	**
FIX	singleESN [A]	0.978	1.705	2.561	2.704	3.314	3.151	2.999	3.316	
FIX	singleESN [B]	**0.930**	1.095	1.885	2.356	2.650	2.704	2.880	2.844	**
FIX	multiESN [A]	1.059	1.148	1.262	1.312	1.339	1.409	1.424	1.162	
FIX	multiESN [B]	0.981	1.007	0.985	**0.994**	1.008	0.999	**0.999**	**0.998**	**
EW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	-
EW	AR(1)	1.117	1.033	1.011	1.002	1.007	1.003	1.004	1.003	-
EW	MIDAS	1.005	1.382	1.339	1.354	1.609	1.444	1.803	1.263	-
EW	DFM [A]	1.144	1.132	1.057	1.093	1.076	1.067	1.038	1.016	-
EW	DFM [B]	0.985	**0.940**	**0.918**	0.995	1.010	**0.980**	1.050	**0.971**	-
EW	singleESN [A]	0.935	1.645	2.184	1.929	2.388	1.959	1.810	2.266	-
EW	singleESN [B]	**0.911**	1.092	1.101	1.529	2.195	1.843	1.847	2.060	-
EW	multiESN [A]	0.922	0.965	1.089	0.978	**0.977**	1.043	1.278	0.995	-
EW	multiESN [B]	0.944	0.992	0.978	**0.977**	0.991	0.985	**0.990**	0.996	-
RW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	
RW	AR(1)	1.080	1.000	0.984	**0.989**	0.982	0.976	**0.963**	0.968	
RW	MIDAS	1.051	1.303	1.310	1.674	1.762	1.467	1.643	1.463	
RW	DFM [A]	1.061	1.033	1.012	1.088	1.077	1.015	1.040	1.069	
RW	DFM [B]	0.947	**0.893**	**0.901**	1.009	1.040	**0.966**	1.030	**0.949**	**
RW	singleESN [A]	0.919	1.788	2.359	2.483	2.981	2.401	2.234	2.690	
RW	singleESN [B]	0.944	1.132	1.214	1.762	2.608	2.552	2.517	2.541	
RW	multiESN [A]	**0.896**	1.047	1.222	1.124	1.122	1.410	1.666	1.316	
RW	multiESN [B]	0.940	1.003	0.969	**0.989**	**0.979**	0.972	0.967	0.961	**

Table 5.6: Relative RMSFE and Uniform Multi-Horizon Model Confidence Set (uMCS) comparison between models in multiple-steps-ahead forecasting exercises. Unconditional mean RMSFE used as reference. FIX: Fixed parameters, EW: Expanding window, and RW: Rolling window. uMCS columns show inclusion among best models: $*$ indicates inclusion at 90%, $**$ indicates inclusion at 75% confidence.

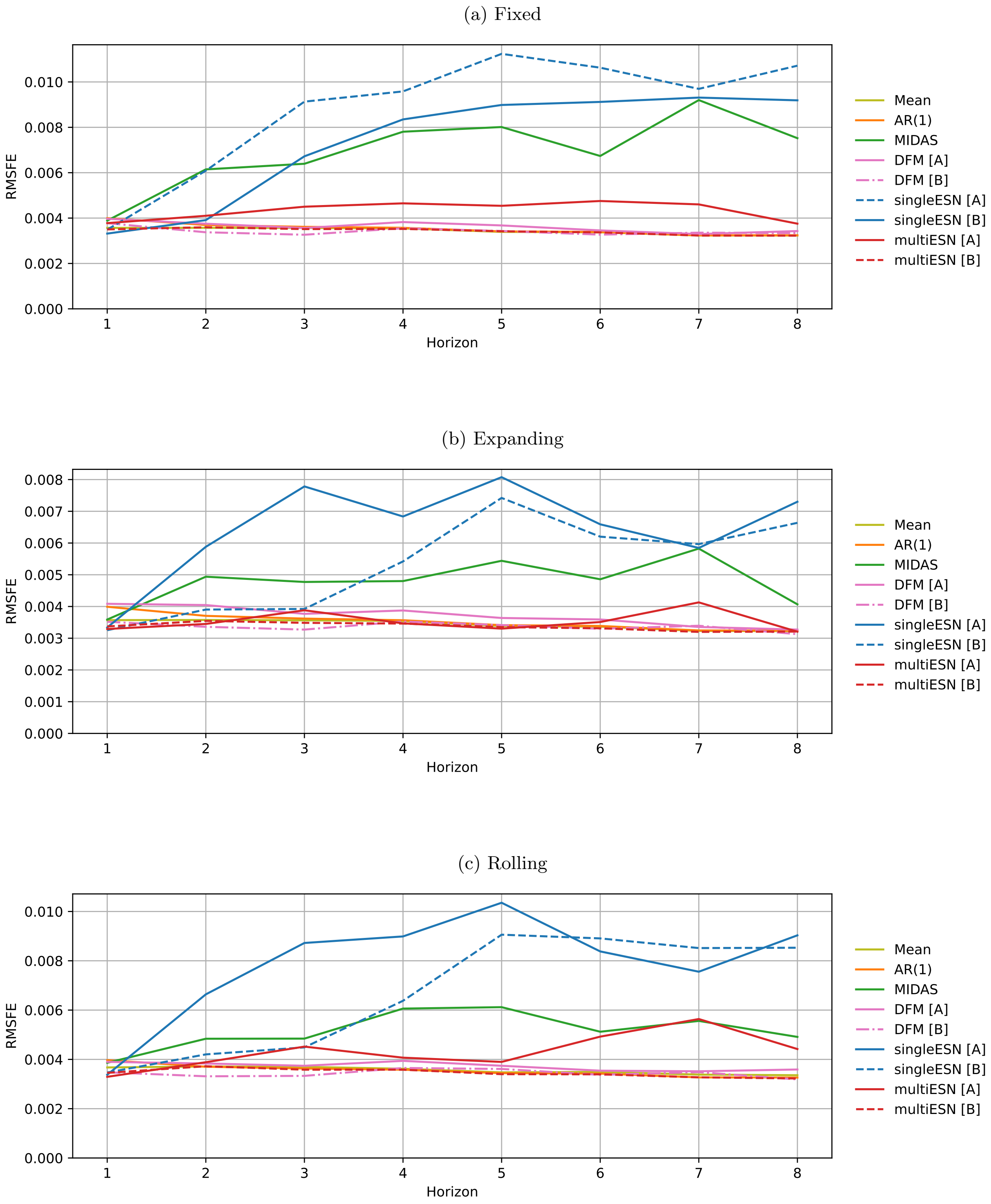
Figure 6: Multistep-ahead GDP Forecasting, RMSFE – 2007 Sample – Small-MD Dataset


(a) Fixed

(b) Expanding

(c) Rolling

Figure 7: Multistep-ahead GDP Forecasting, RMSFE – 2011 Sample – Small-MD Dataset



(a) Fixed

(b) Expanding

(c) Rolling

1-Step-ahead GDP Forecasting - Medium-MD Dataset

Model	Fixed Parameters				Expanding Window				Rolling Window			
	2007		2011		2007		2011		2007		2011	
	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS	MSFE	MCS
Mean	1.000		1.000	**	1.000	**	1.000	**	1.000	**	1.000	**
AR(1)	0.758	*	1.230	**	0.789	**	1.226	**	0.775	*	1.209	**
DFM [A]	0.841	*	1.325	*	0.682	**	1.272	**	0.747	*	1.517	**
DFM [B]	1.118	*	1.408	**	0.821	*	1.117	**	0.926		1.186	**
singleESN [A]	0.967	*	1.717	*	0.775	**	1.072	**	0.791	*	1.493	*
singleESN [B]	0.826	*	1.278	**	0.655	**	1.028	**	0.561	**	0.944	**
multiESN [A]	0.901	*	1.080	**	0.618	**	0.913	**	0.556	**	0.884	**
multiESN [B]	**0.682**	**	**0.748**	**	**0.587**	**	**0.774**	**	**0.547**	**	**0.728**	**

Table 5.7: Relative MSFE and Model Confidence Set (MCS) comparison between models in 1-step-ahead forecasting exercises. Unconditional mean MSFE is used as a reference. MCS columns show inclusion among best models: $*$ indicates inclusion at 90% confidence, $**$ indicates inclusion at 75% confidence.

mean. This evaluation is confirmed by uMCS tests, consistently with the multistep results obtained with the Small-MD dataset.

6 Conclusions

Macroeconomic forecasting – especially long-term forecasting of macroeconomic aggregates – is a topic of crucial importance for institutional policymakers, private companies, and economic researchers. Given the modern-day availability of "big data" resources, methods capable of integrating heterogeneous data sources are increasingly sought to provide more precise and robust forecasts.

This paper presents a new methodological framework inspired by the Reservoir Computing literature to deal with data sampled at multiple frequencies and with multiple-step-ahead forecasts. We have then taken Echo State Networks – a type of RC models – and formally extended them to allow the modeling of data with multiple release frequencies. Our discussion encompasses model fitting, hyperparameter tuning, and forecast computation. As a result, we provide two classes of models, single- and multiple reservoir multi-frequency ESNs, that can be effectively applied to our empirical setup: forecasting US GDP growth using monthly and daily data series. Along with the unconditional mean and AR(1) model, we considered two well-known methods, MIDAS and DFMs, as the current benchmarks available in the literature. In our applications, we find that MFESN models are computationally more efficient and easier to implement than DFMs and MIDAS, respectively, and perform better than or as well as benchmarks in terms of MSFE. These improvements are statistically significant in a number of setups, as shown by our MCS and MDM tests. Thus, we argue that our machine learning-based methodology can be a useful addition to the toolbox of contemporary macroeconomic forecasters.

Lastly, we wish to highlight the many potential areas of research that we believe would be interesting to explore in the future. We have not discussed the role of the distribution from which we sample the entries of the reservoir matrices. While it is known that these can have significant effects on the forecasting capacity of an ESN model, the literature lacks definitive theoretical results (even for dynamical systems applications) or systematic studies with stochastic inputs and targets. The hyperparameter tuning routine we have developed neither allows separating individual hyperparameters nor does it tackle the identification problem. Moreover, we assume that the ridge regression penalty strength, λ, is tuned *ex ante*: it would be interesting and desirable to understand if it is

Multistep-ahead GDP Forecasting - Medium-MD Dataset - 2007 Sample

Setup	Model	Horizon								uMCS
		1	2	3	4	5	6	7	8	
FIX	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	*
FIX	AR(1)	0.870	0.950	0.982	0.991	0.992	0.991	0.992	0.992	
FIX	DFM [A]	0.914	**0.947**	**0.955**	0.988	1.015	1.027	1.034	0.995	**
FIX	DFM [B]	1.046	1.204	1.293	1.341	1.649	1.984	2.101	2.070	*
FIX	singleESN [A]	0.985	0.995	0.995	0.995	0.994	0.992	0.992	0.992	*
FIX	singleESN [B]	0.912	0.985	0.985	**0.985**	**0.980**	**0.976**	**0.976**	**0.976**	*
FIX	multiESN [A]	0.950	0.993	0.994	0.994	0.992	0.990	0.990	0.990	*
FIX	multiESN [B]	**0.826**	0.972	0.988	0.990	0.989	0.986	0.985	0.985	*
EW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	**1.000**	**1.000**	-
EW	AR(1)	0.887	0.922	0.951	0.962	0.957	0.981	1.001	1.008	-
EW	DFM [A]	0.805	0.916	0.978	1.038	1.077	1.126	1.077	1.073	-
EW	DFM [B]	0.893	1.134	1.418	1.567	2.238	2.964	3.375	3.629	-
EW	singleESN [A]	0.879	1.125	1.305	1.442	1.860	2.166	2.361	2.443	-
EW	singleESN [B]	0.802	1.174	1.439	1.744	2.305	2.869	2.935	3.167	-
EW	multiESN [A]	0.780	0.935	1.012	1.005	1.093	1.337	1.328	1.313	-
EW	multiESN [B]	**0.760**	**0.874**	**0.911**	**0.891**	**0.863**	**0.971**	1.030	1.051	-
RW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	
RW	AR(1)	0.898	0.943	0.968	0.974	0.963	**0.968**	**0.970**	**0.962**	
RW	DFM [A]	0.837	0.913	0.924	0.954	1.012	0.997	1.018	1.005	
RW	DFM [B]	0.932	1.116	1.232	1.414	1.952	2.704	3.183	3.294	
RW	singleESN [A]	0.873	1.274	1.530	1.652	2.095	2.575	2.786	3.014	
RW	singleESN [B]	0.732	1.190	1.490	1.712	2.218	2.861	2.967	3.094	
RW	multiESN [A]	0.732	0.914	0.960	1.011	1.202	1.618	1.683	1.572	
RW	multiESN [B]	**0.731**	**0.871**	**0.875**	**0.844**	**0.771**	0.971	1.014	1.014	**

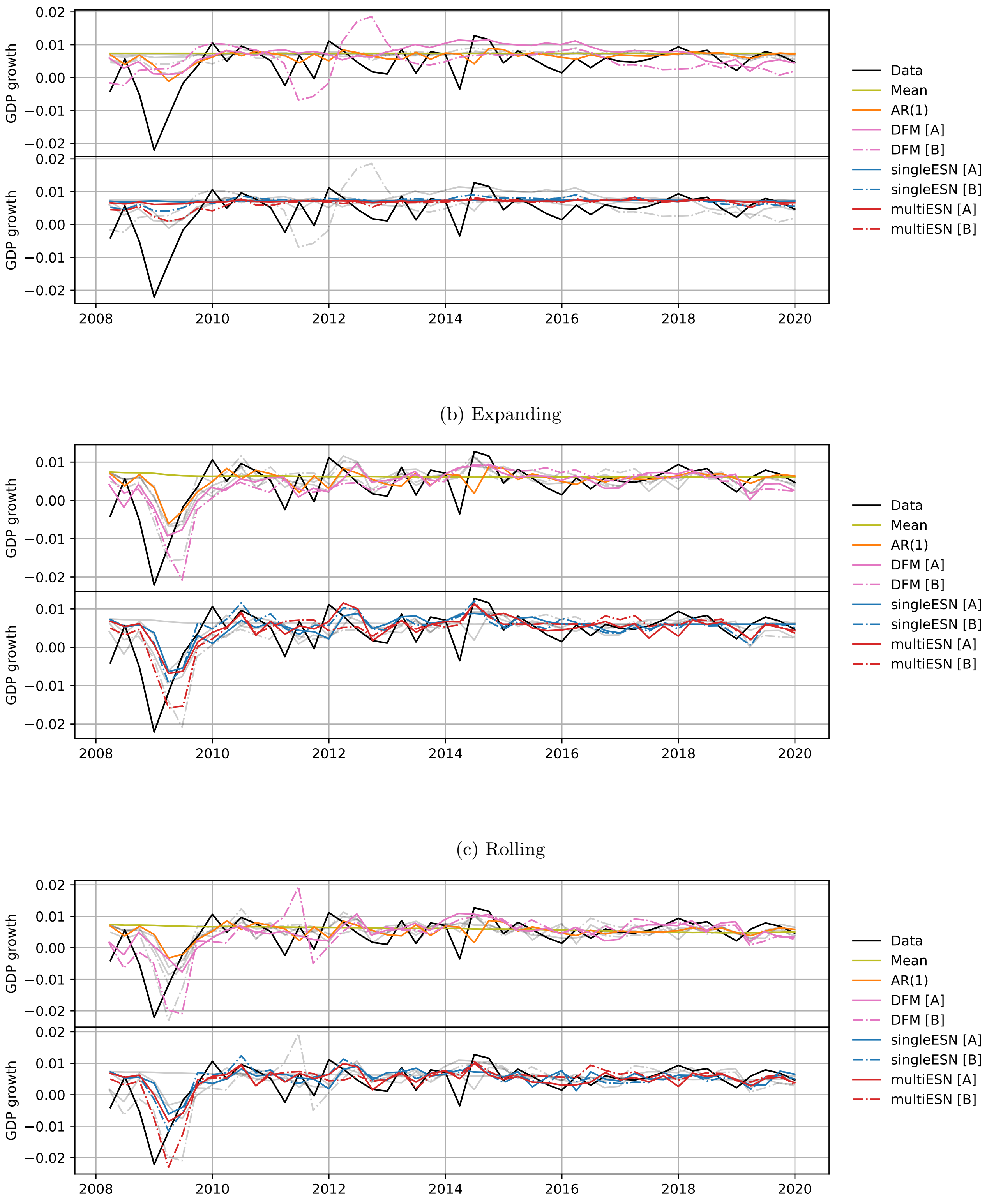
Table 5.8: Relative RMSFE and Uniform Multi-Horizon Model Confidence Set (uMCS) comparison between models in multiple-steps-ahead forecasting exercises. Unconditional mean RMSFE used as reference. FIX: Fixed parameters, EW: Expanding window, and RW: Rolling window. uMCS columns show inclusion among best models: $*$ indicates inclusion at 90% confidence, $**$ indicates inclusion at 75% confidence.

Multistep-ahead GDP Forecasting - Medium-MD Dataset - 2011 Sample

Setup	Model	Horizon								uMCS
		1	2	3	4	5	6	7	8	
FIX	Mean	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	*
FIX	AR(1)	1.119	1.031	1.008	1.001	1.001	0.999	**0.999**	**0.998**	**
FIX	DFM [A]	1.126	0.987	0.962	1.054	1.031	**0.988**	1.001	1.002	**
FIX	DFM [B]	1.149	0.987	**0.885**	1.064	1.142	1.134	1.273	1.296	
FIX	singleESN [A]	1.283	1.921	2.527	3.038	3.285	3.154	3.193	3.655	
FIX	singleESN [B]	1.059	1.523	1.918	2.417	2.812	2.683	2.703	2.970	
FIX	multiESN [A]	1.011	1.061	1.434	1.477	1.748	2.030	2.023	1.994	
FIX	multiESN [B]	**0.841**	**0.945**	0.997	**0.978**	1.004	1.015	1.013	1.014	**
EW	Mean	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	-
EW	AR(1)	1.117	1.033	1.011	1.002	1.007	1.003	1.004	1.003	-
EW	DFM [A]	1.092	0.942	**0.944**	1.049	1.026	**0.994**	**0.996**	**0.999**	-
EW	DFM [B]	0.971	1.046	1.031	1.114	1.238	1.116	1.223	1.310	-
EW	singleESN [A]	1.039	1.451	1.980	2.385	2.699	2.353	2.506	2.608	-
EW	singleESN [B]	0.992	1.828	2.465	3.072	3.547	3.357	3.368	3.610	-
EW	multiESN [A]	0.934	1.014	1.391	1.252	1.371	1.369	1.228	1.279	-
EW	multiESN [B]	**0.857**	**0.931**	1.003	**0.973**	1.002	1.009	1.025	1.029	-
RW	Mean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	**
RW	AR(1)	1.080	1.000	0.984	**0.989**	**0.982**	**0.976**	**0.963**	**0.968**	**
RW	DFM [A]	1.113	0.982	**0.927**	1.038	1.030	0.997	1.016	1.028	*
RW	DFM [B]	0.881	0.996	1.021	1.098	1.150	1.114	1.114	1.212	**
RW	singleESN [A]	1.193	2.267	3.265	3.580	4.090	3.790	4.015	4.562	
RW	singleESN [B]	0.927	1.933	2.612	3.265	3.753	3.567	3.556	3.792	
RW	multiESN [A]	0.900	1.049	1.500	1.465	1.789	1.707	1.505	1.462	
RW	multiESN [B]	**0.816**	**0.916**	0.977	1.009	**0.982**	0.988	0.974	0.981	**

Table 5.9: Relative RMSFE and Uniform Multi-Horizon Model Confidence Set (uMCS) comparison between models in multiple-steps-ahead forecasting exercises. Unconditional mean RMSFE used as reference. FIX: Fixed parameters, EW: Expanding window, and RW: Rolling window. uMCS columns show inclusion among best models: $*$ indicates inclusion at 90% confidence, $**$ indicates inclusion at 75% confidence.

Figure 8: 1-Step-ahead GDP Forecasting – 2007 Sample – Medium-MD Dataset

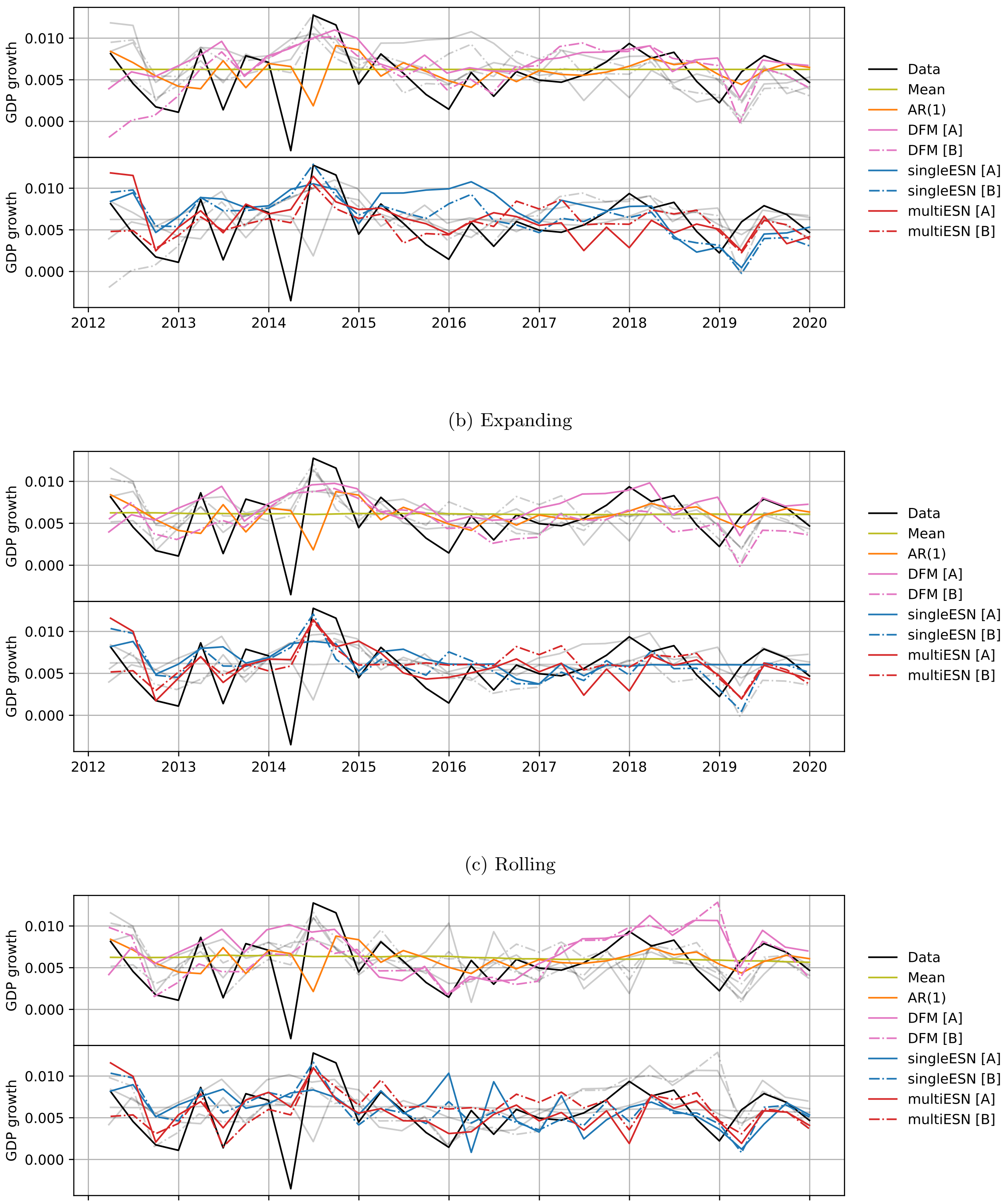


(a) Fixed

(b) Expanding

(c) Rolling

Figure 9: 1-Step-ahead GDP Forecasting – 2011 Sample – Medium-MD Dataset

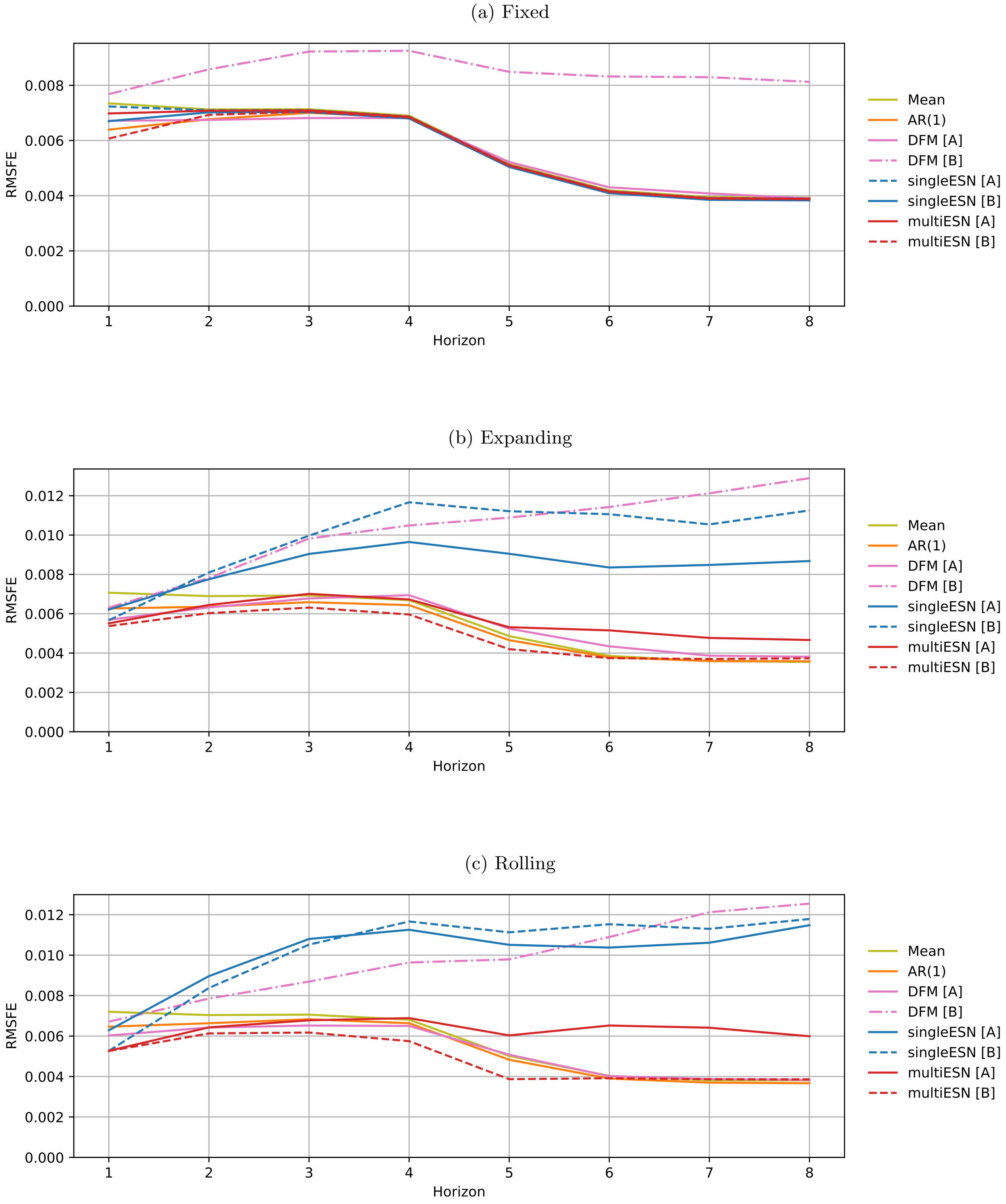


(a) Fixed

(b) Expanding

(c) Rolling

Figure 10: Multistep-ahead GDP Forecasting, RMSFE – 2007 Sample – Medium-MD Dataset



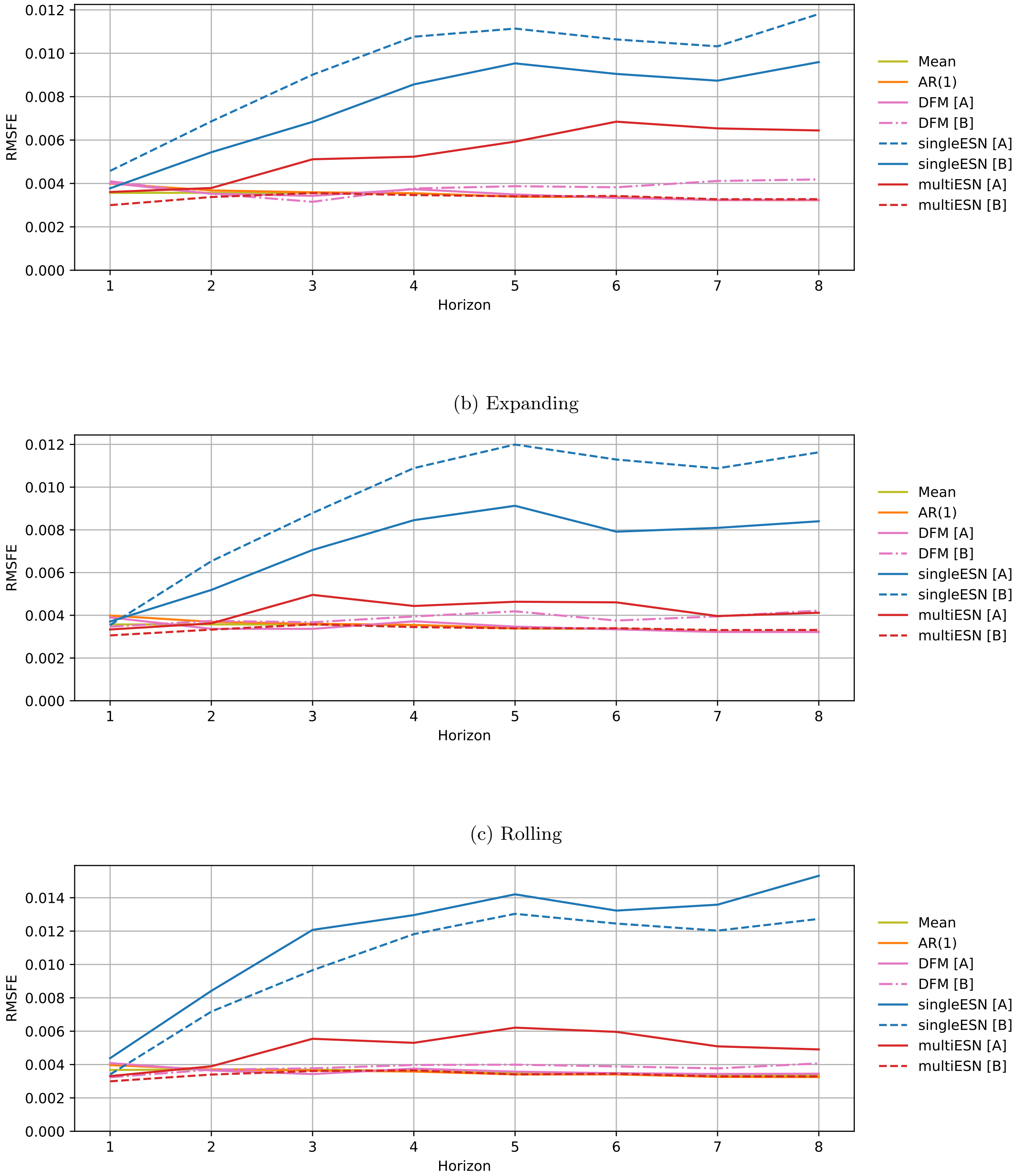
(a) Fixed

(b) Expanding

(c) Rolling

Figure 11: Multistep-ahead GDP Forecasting, RMSFE – 2011 Sample – Medium-MD Dataset

(a) Fixed



(b) Expanding

(c) Rolling

possible to jointly tune λ and $\boldsymbol{\varphi}$, or rather if one can fully separate their selection. In our preliminary experiments, we have noticed that the roles of the ridge penalty and the input scaling, for example, cannot be trivially disentangled – thus prompting the ψ-form normalization. Model selection for the dimension of MFESN models is another question that would be key to exploring and designing more efficient and effective ESN models, especially when dealing with multiple frequencies and reservoirs. Finally, practitioners may be interested in identifying the combination of frequencies in the regressor series that would lead to the most accurate GDP forecasts produced by MFESN models.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2016.

S. Almon. The distributed lag between capital appropriations and expenditures. *Econometrica*, 33(1):178–196, 1965.

E. Andreou, E. Ghysels, and A. Kourtellos. Should macroeconomic forecasters use daily financial data and how? *Journal of Business & Economic Statistics*, 31(2):240–251, 2013.

D. W. K. Andrews. Heteroskedasticity and autocorrelation consistent covariance matrix estimation. *Econometrica*, 59 (3):817–858, 1991.

C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society B*, 72:269–342, 2010.

D. Aparicio and M. L. de Prado. How hard is it to pick the right model? MCS and backtest overfitting. *Algorithmic Finance*, 7(1-2):53–61, 2018.

T. Arcomano, I. Szunyogh, A. Wikner, J. Pathak, B. R. Hunt, and E. Ott. A hybrid approach to atmospheric modeling that combines machine learning with a physics-based numerical model. *Journal of Advances in Modeling Earth Systems*, 14(3):e2021MS002712, 2022.

M. T. Armesto, K. M. Engemann, and M. T. Owyang. Forecasting with Mixed Frequencies. *Federal Reserve Bank of St. Louis Review*, 92(6):521–536, 2010.

S. Arora, M. A. Little, and P. E. McSharry. Nonlinear and nonparametric modeling approaches for probabilistic forecasting of the US gross national product. *Studies in Nonlinear Dynamics and Econometrics*, 17(4):395–420, 2013.

S. B. Aruoba, F. X. Diebold, and C. Scotti. Real-time measurement of business conditions. *Journal of Business & Economic Statistics*, 27(4):417–427, 2009.

A. Babii, E. Ghysels, and J. Striaukas. Machine learning time series regressions with an application to nowcasting. *Journal of Business & Economic Statistics*, 40(3):1094–1106, 2022.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*, 2014.

J. Bai and S. Ng. Determining the number of primitive shocks in factor models. *Journal of Business & Economic Statistics*, 25(1):52–60, 2007.

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

J. Bai, E. Ghysels, and J. H. Wright. State space models and MIDAS regressions. *Econometric Reviews*, 32(7):779–813, 2013.

Bai Zhang, D. J. Miller, and Yue Wang. Nonlinear system modeling with random matrices: echo state networks revisited. *IEEE Transactions on Neural Networks and Learning Systems*, 23(1):175–182, jan 2012.

G. Ballarin. Ridge regularized estimation of VAR models for inference. Preprint. 2023.

G. Ballarin, L. Grigoryeva, and J.-P. Ortega. Memory of recurrent networks: Do we compute it right? Preprint. 2023.

M. Bańbura and M. Modugno. Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data. *Journal of Applied Econometrics*, 29(1):133–160, 2014.

M. Bańbura and G. Rünstler. A look into the factor model black box: Publication lags and the role of hard and soft data in forecasting GDP. *International Journal of Forecasting*, 27(2):333–346, 2011.

M. Bańbura, D. Giannone, M. Modugno, and L. Reichlin. Now-Casting and the Real-Time Data Flow. In *Handbook of Economic Forecasting*, pages 195–237. Elsevier, 2013.

A. Belloni, V. Chernozhukov, D. Chetverikov, and K. Kato. Some new asymptotic theory for least squares series: Pointwise and uniform results. *Journal of Econometrics*, 186(2):345–366, 2015.

C. Bergmeir, R. J. Hyndman, and B. Koo. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120:70–83, 2018.

F. Blasques, S. J. Koopman, M. Mallee, and Z. Zhang. Weighted maximum likelihood for dynamic factor analysis and forecasting with mixed frequency data. *Journal of Econometrics*, 193(2):405–417, 2016.

J. Boivin and S. Ng. Understanding and comparing factor-based forecasts. Technical report, National Bureau of Economic Research, 2005.

T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

C. Borio. Rediscovering the macroeconomic roots of financial stability policy: Journey, challenges, and a way forward. *Annual Review of Financial Economics*, 3(1):87–117, 2011.

C. Borio. The Great Financial Crisis: Setting priorities for new statistics. *Journal of Banking Regulation*, 14(3-4): 306–317, 2013.

C. Borio and P. W. Lowe. Asset prices, financial and monetary stability: Exploring the Nexus. *SSRN Electronic Journal*, 2002.

S. Boyd and L. Chua. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits and Systems*, 32(11):1150–1161, 1985.

F. Bräuning and S. J. Koopman. Forecasting macroeconomic variables using collapsed dynamic factor analysis. *International Journal of Forecasting*, 30(3):572–584, 2014.

M. Buehner and P. Young. A tighter bound for the echo state property. *IEEE Transactions on Neural Networks*, 17(3): 820–824, 2006.

B. Buell, R. Cherif, C. Chen, Hyeon, J. Tang, and N. Wendt. Impact of COVID-19: Nowcasting and big data to track economic activity in Sub-Saharan Africa. *IMF Working Paper*, 124:1–61, 2021.

M. Camacho and G. Pérez-Quirós. Introducing the euro-sting: Short-term indicator of euro area growth. *J. Appl. Econ.*, 25:663–694, 2010.

W. Cao, X. Wang, Z. Ming, and J. Gao. A review on neural networks with random weights. *Neurocomputing*, 275: 278–287, 2018.

A. Carriero, A. B. Galvão, and G. Kapetanios. A comprehensive evaluation of macroeconomic forecasting methods. *International Journal of Forecasting*, 35(4):1226–1239, 2019.

M. Chauvet, Z. Senyuz, and E. Yoldas. What does financial volatility tell us about macroeconomic fluctuations? *Journal of Economic Dynamics and Control*, 52:340–360, 2015.

X. Chen and T. M. Christensen. Optimal uniform convergence rates and asymptotic normality for series estimators under weak dependence and weak conditions. *Journal of Econometrics*, 188(2):447–465, 2015.

X. Chen and E. Ghysels. News - good or bad - and its impact on volatility predictions over multiple horizons. *Review of Financial Studies*, 24(1):46–81, 2010.

M. Clements and A. Galvão. Macroeconomic forecasting with mixed-frequency data: forecasting output growth in the United States. *Journal of Business & Economic Statistics*, 26:546–554, 2008.

M. P. Clements and A. Galvão. Forecasting US output growth using leading indicators: an appraisal using MIDAS models. *Journal of Applied Econometrics*, 7(7):1187–1206, 2009.

J. P. Crutchfield, W. L. Ditto, and S. Sinha. Introduction to focus issue: Intrinsic and designed computation: Information processing in dynamical systems - beyond the digital hegemony. *Chaos*, 20(3):037101, 2010.

D. Delle Monache and I. Petrella. Efficient matrix approach for classical inference in state space models. *Economics Letters*, 181:22–27, 2019.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal statistical Society*, 39(1):1–38, 1977.

F. X. Diebold and R. S. Mariano. Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 20(1): 134–144, 2002.

J. V. Dillon, I. Langmore, D. Tran, E. Brevdo, S. Vasudevan, D. Moore, B. Patton, A. Alemi, M. Hoffman, and R. A. Saurous. Tensorflow distributions. *arXiv:1711.10604*, 2017.

R. Douc, E. Moulines, J. Olsson, and R. Van Handel. Consistency of the maximum likelihood estimator for general hidden Markov models. *The Annals of Statistics*, 39(1):474–513, 2011.

A. Doucet, N. de Freitas, and N. J. Gordon, editors. *Sequential Monte Carlo Methods in Practice.* Statistics for Engineering and Information Science. Springer, 2001.

K. Doya. Bifurcations in the learning of recurrent neural networks. In *Proceedings of IEEE International Symposium on Circuits and Systems*, volume 6, pages 2777–2780, 1992.

C. Doz, D. Giannone, and L. Reichlin. A two-step estimator for large approximate dynamic factor models based on Kalman filtering. *Journal of Econometrics*, 164(1):188–205, 2011.

I. Farkas, R. Bosak, and P. Gergel. Computational analysis of memory capacity in echo state networks. *Neural Networks*, 83:109–120, 2016.

L. Ferrara, C. Marsilli, and J.-P. Ortega. Forecasting growth during the Great Recession: is financial volatility the missing ingredient? *Economic Modelling*, 36:44–50, 2014.

M. Forni, M. Hallin, M. Lippi, and L. Reichlin. The generalized dynamic factor model: One-sided estimation and forecasting. *Journal of the American Statistical Association*, 100(471):830–840, 2005.

C. Foroni and M. Marcellino. A comparison of mixed approaches for modelling euro area macroeconomic variables. Technical report, EUI, 2011.

C. Frale, M. Marcellino, G. L. Mazzi, and T. Proietti. EUROMIND: a monthly indicator of the euro area economic conditions. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 174:439–470, 2011.

N. Francis, E. Ghysels, and M. T. Owyang. The low-frequency impact of daily monetary policy shocks. Technical report, Federal Reserve Bank of St. Louis, 2011.

P. Fuleky, editor. *Macroeconomic Forecasting in the Era of Big Data.* Springer International Publishing, 2020.

P. Gagliardini, E. Ghysels, and M. Rubin. Indirect inference estimation of mixed frequency stochastic volatility state space models using MIDAS regressions and ARCH models. *Journal of Financial Econometrics*, 15(4):509–560, 2017.

A. B. Galvão. Changes in predictive ability with mixed frequency data. *International Journal of Forecasting*, 29(3): 395–410, 2013.

A. B. Galvão and M. Marcellino. Endogenous monetary policy regimes and the great moderation. Technical report, EUI, 2010.

J. Geweke. The dynamic factor analysis of economic time series. *Latent variables in socio-economic models*, 1977.

E. Ghysels. Macroeconomics and the reality of mixed frequency data. *Journal of Econometrics*, 193(2):294–314, 2016.

E. Ghysels and J. H. Wright. Forecasting professional forecasters. *Journal of Business & Economic Statistics*, 27(4): 504–516, 2009.

E. Ghysels, P. Santa-Clara, and R. Valkanov. The MIDAS touch: Mixed data sampling regression models. Technical Report 919, UCLA: Finance, 2004.

E. Ghysels, A. Sinko, and R. Valkanov. MIDAS regressions: Further results and new directions. *Econometric Reviews*, 26(1):53–90, feb 2007.

D. Giannone, L. Reichlin, and D. Small. Nowcasting: The real-time informational content of macroeconomic data. *Journal of Monetary Economics*, 55(4):665–676, 2008.

S. J. Godsill, A. Doucet, and M. West. Monte Carl smoothing for nonlinear time series. *Journal of the American Statistical Association*, 99(465):156–168, 2004.

L. Gonon and J.-P. Ortega. Reservoir computing universality with stochastic inputs. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1):100–112, 2020.

L. Gonon and J.-P. Ortega. Fading memory echo state networks are universal. *Neural Networks*, 138:10–13, 2021.

L. Gonon, L. Grigoryeva, and J.-P. Ortega. Risk bounds for reservoir computing. *Journal of Machine Learning Research*, 21(240):1–61, 2020a.

L. Gonon, L. Grigoryeva, and J.-P. Ortega. Memory and forecasting capacities of nonlinear recurrent networks. *Physica D*, 414(132721):1–13., 2020b.

L. Gonon, L. Grigoryeva, and J.-P. Ortega. Approximation error estimates for random neural networks and reservoir systems. *The Annals of Applied Probability*, 33(1):28–69, 2023a.

L. Gonon, L. Grigoryeva, and J. P. Ortega. Infinite-dimensional reservoir computing. *Arxiv preprint*, 2023b.

A. Goudarzi, S. Marzen, P. Banda, G. Feldman, M. R. Lakin, C. Teuscher, and D. Stefanovic. Memory and information processing in recurrent neural networks. Technical report, Portland State University, 2016.

D. Gramlich, G. L. Miller, M. V. Oet, and S. J. Ong. Early warning systems for systemic banking risk: Critical review and modeling implications. *Banks and Bank Systems*, 5(2):199–211, 2010.

L. Grigoryeva and J.-P. Ortega. Universal discrete-time reservoir computers with stochastic inputs and linear readouts using non-homogeneous state-affine systems. *Journal of Machine Learning Research*, 19(24):1–40, 2018a.

L. Grigoryeva and J.-P. Ortega. Echo state networks are universal. *Neural Networks*, 108:495–508, 2018b.

L. Grigoryeva and J.-P. Ortega. Differentiable reservoir computing. *Journal of Machine Learning Research*, 20(179): 1–62, 2019.

L. Grigoryeva and J.-P. Ortega. Dimension reduction in recurrent networks by canonicalization. *Journal of Geometric Mechanics*, 13(4):647–677, 2021.

L. Grigoryeva, J. Henriques, L. Larger, and J.-P. Ortega. Optimal nonlinear information processing capacity in delay-based reservoir computers. *Scientific Reports*, 5(12858):1–11, 2015.

L. Grigoryeva, J. Henriques, L. Larger, and J.-P. Ortega. Nonlinear memory capacity of parallel time-delay reservoir computers in the processing of multidimensional signals. *Neural Computation*, 28:1411–1451, 2016.

L. Grigoryeva, A. G. Hart, and J.-P. Ortega. Learning strange attractors with reservoir systems. *arXiv preprint arXiv:2108.05024*, 2021.

M. Hallin and R. Liška. Determining the number of factors in the general dynamic factor model. *Journal of the American Statistical Association*, 102(478):603–617, 2007.

P. R. Hansen, Z. Huang, and H. H. Shek. Realized GARCH: a joint model for returns and realized measures of volatility. *Journal of Applied Econometrics*, 27(6):877–906, 2011.

A. G. Hart, J. L. Hook, and J. H. P. Dawes. Echo State Networks trained by Tikhonov least squares are L2(μ) approximators of ergodic dynamical systems. *Physica D: Nonlinear Phenomena*, 421:132882, 2021.

A. C. Harvey, S. J. Koopman, and J. Penzer. Messy time series: A unified approach. *Advances in Econometrics*, 13: 103–144, 1998.

D. Harvey, S. Leybourne, and P. Newbold. Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2):281–291, 1997.

T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, second edition, 2009.

T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *The Annals of Statistics*, 50(2):949–986, 2022.

J. Hatzius, P. Hooper, F. Mishkin, K. Schoenholtz, and M. Watson. Financial conditions indexes: A fresh look after the financial crisis. Technical report, National Bureau of Economic Research, 2010.

S. Hihi and Y. Bengio. Hierarchical recurrent neural networks for long-term dependencies. *Advances in neural information processing systems*, 8, 1995.

I. Hindrayanto, S. J. Koopman, and J. de Winter. Forecasting and nowcasting economic growth in the euro area using factor models. *International Journal of Forecasting*, 32(4):1284–1305, 2016.

H. Hong and M. Yogo. What does futures market interest tell us about the macroeconomy and asset prices? *Journal of Financial Economics*, 105(3):473–490, 2012.

J. L. Horowitz. *Semiparametric and Nonparametric Methods in Econometrics*. Springer, New York, NY, USA, 2009.

F. Huber, G. Koop, L. Onorante, M. Pfarrhofer, and J. Schreiner. Nowcasting in a pandemic using non-parametric mixed frequency VARs. *ECB Working Paper Series*, 2510:1–40, 2021.

R. Ingenito and B. Trehan. Using monthly data to predict quarterly output. *Econometric Reviews*, pages 3–11, 1996.

A. Inoue, L. Jin, and B. Rossi. Rolling window selection for out-of-sample forecasting with time-varying parameters. *Journal of Econometrics*, 196(1):55–67, 2017.

H. Ishwaran and J. Rao. Geometry and Properties of Generalized Ridge Regression in High Dimensions. In S. Ahmed, editor, *Contemporary Mathematics*, volume 622, pages 81–93. American Mathematical Society, Providence, Rhode Island, 2014.

H. Jaeger. The 'echo state' approach to analysing and training recurrent neural networks with an erratum note. Technical report, German National Research Center for Information Technology, 2010.

H. Jaeger and H. Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80, 2004.

C. Jardet and B. Meunier. Nowcasting world GDP growth with high-frequency data. *Journal of Forecasting*, 41(6): 1181–1200, 2022.

B. Jungbacker and S. J. Koopman. Likelihood-based dynamic factor analysis for measurement and forecasting. Technical report, Tinbergen Institute Discussion Paper, 2015.

J. Kang and K. Y. Kwon. Can commodity futures risk factors predict economic growth? *Journal of Futures Markets*, 40(12):1825–1860, 2020.

N. Kantas, A. Doucet, S. S. Singh, J. Maciejowski, N. Chopin, and Others. On particle methods for parameter estimation in state-space models. *Statistical science*, 30(3):328–351, 2015.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. 2014.

A. B. Kock, M. Medeiros, and G. Vasconcelos. Penalized Time Series Regression. In P. Fuleky, editor, *Macroeconomic Forecasting in the Era of Big Data: Theory and Practice*, Advanced Studies in Theoretical and Applied Econometrics, pages 193–228. Springer International Publishing, Cham, 2020.

A. Kostrov. *Essays on the use of MIDAS regressions in banking and finance*. PhD thesis, Universität St. Gallen, 2021.

R. Legenstein and W. Maass. What makes a dynamical system computationally powerful? In S. Haykin, editor, *New directions in statistical signal processing: from systems to brain*. MIT Press, Cambridge, MA, 2007.

F. LeGland and L. Mevel. Recursive estimation in hidden Markov models. In *Proceedings of the 36th IEEE Conference on Decision and Control*, volume 4, pages 3468–3473, 1997.

M. Leippold and H. Yang. Particle filtering, learning, and smoothing for mixed-frequency state-space models. *Econometrics and Statistics*, 12:25–41, 2019.

M. Lukoševičius. A Practical Guide to Applying Echo State Networks. In G. Montavon, G. B. Orr, and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade: Second Edition*, Lecture Notes in Computer Science, pages 659–686. Springer, Berlin, Heidelberg, 2012.

M. Lukoševičius and H. Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.

W. Maass, T. Natschläger, and H. Markram. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Computation*, 14:2531–2560, 2002.

O.-A. Maillard and R. Munos. Linear regression with random projections. *Journal of Machine Learning Research*, 13 (89):2735–2772, 2012.

G. Manjunath and H. Jaeger. Echo state property linked to an input: exploring a fundamental characteristic of recurrent neural networks. *Neural Computation*, 25(3):671–696, 2013.

G. Manjunath and J.-P. Ortega. Transport in reservoir computing. *Physica D: Nonlinear Phenomena*, 449:133744, 2023.

M. Marcellino and C. Schumacher. Factor MIDAS for nowcasting and forecasting with ragged-edge data: A model comparison for German GDP. *Oxford Bulletin of Economics and Statistics*, 72(4):518–550, 2010.

M. Marcellino, J. H. Stock, and M. W. Watson. A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series. *Journal of Econometrics*, 135(1-2):499–526, 2006.

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

C. Marsilli. *Mixed-Frequency Modeling and Economic Forecasting.* PhD thesis, Université de Franche-Comté, 2014.

M. W. McCracken and S. Ng. FRED-MD: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics*, 34(4):574–589, 2016.

M. W. McCracken and S. Ng. FRED-QD: A quarterly database for macroeconomic research. Technical report, Federal Reserve Bank of St. Louis, 2020.

L. Monteforte and G. Moretti. Real-time forecasts of inflation: The role of financial variables. *Journal of Forecasting*, 32(1):51–61, 2012.

J. Morley. Macro-finance linkages. *Journal of Economic Surveys*, 30(4):698–711, 2015.

A. Onatski. Asymptotics of the principal components estimator of large factor models with weakly influential factors. *Journal of Econometrics*, 168(2):244–258, 2012.

L. Paranhos. Predicting inflation with neural networks. 2021.

R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, volume 28, pages 1310–1318. PMLR, 2013.

J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott. Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data. *Chaos*, 27(12), 2017.

J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Physical Review Letters*, 120(2):24102, 2018.

D. Pettenuzzo, A. Timmermann, and R. Valkanov. A MIDAS approach to modeling first and second moment dynamics. *Journal of Econometrics*, 193(2):315–334, 2016.

T. Proietti and F. Moauro. Dynamic factor analysis with non-linear temporal aggregation constraints. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 55(2):281–300, 2006.

D. Qin, S. van Huellen, Q. C. Wang, and T. Moraitis. Algorithmic modelling of financial conditions for macro predictive purposes: Pilot application to USA data. *Econometrics*, 10(2):22, 2022.

R. Quaedvlieg. Multi-horizon forecast comparison. *Journal of Business & Economic Statistics*, 39(1):40–53, 2021.

A. Rodan and P. Tino. Minimum complexity echo state network. *IEEE Transactions on Neural Networks*, 22(1):131–44, 2011.

H. Salehinejad, J. Baarbe, S. Sankar, J. Barfett, E. Colak, and S. Valaee. Recent advances in recurrent neural networks. 2017.

T. J. Sargent, C. A. Sims, and Others. Business cycle modeling without pretending to have too much a priori economic theory. *New methods in business cycle research*, 1:145–168, 1977.

F. Schorfheide, D. Song, and A. Yaron. Identifying long-run risks: A Bayesian mixed-frequency approach. *Econometrica*, 86(2):617–654, 2018.

J. H. Stock and M. W. Watson. Evidence on structural instability in macroeconomic time series relations. *Journal of Business & Economic Statistics*, 14(1):11–30, 1996.

J. H. Stock and M. W. Watson. Macroeconomic forecasting using diffusion indexes. *Journal of Business & Economic Statistics*, 20(2):147–162, 2002.

J. H. Stock and M. W. Watson. Forecasting with Many Predictors. In G. Elliot, C. W. Granger, and A. Timmermann, editors, *Handbook of Economic Forecasting*, volume 1. Elsevier edition, 2006.

J. H. Stock and M. W. Watson. Dynamic factor models, factor-augmented vector autoregressions, and structural vector autoregressions in macroeconomics. In *Handbook of macroeconomics*, volume 2, pages 415–525. Elsevier, 2016.

G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose. Recent advances in physical reservoir computing: A review. *Neural Networks*, 115:100–123, 2019.

L. van der Maaten. Learning a parametric embedding by preserving local structure. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, pages 384–391. PMLR, 2009.

S. van Huellen, D. Qin, S. Lu, H. Wang, Q. C. Wang, and T. Moraitis. Modelling opportunity cost effects in money demand due to openness. *International Journal of Finance & Economics*, 27(1):697–744, 2020.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NeurIPS*, pages 1–11, 2017.

G. Wainrib and M. N. Galtier. A local echo state property through the largest Lyapunov exponent. *Neural Networks*, 76:39–45, apr 2016.

M. W. Watson and R. F. Engle. Alternative algorithms for the estimation of dynamic factor, mimic and varying coefficient regression models. *Journal of Econometrics*, 23(3):385–400, 1983.

A. Wikner, J. Pathak, B. R. Hunt, I. Szunyogh, M. Girvan, and E. Ott. Using data assimilation to train a hybrid forecast system that combines machine-learning and knowledge-based components. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(5):53114, 2021.

I. B. Yildiz, H. Jaeger, and S. J. Kiebel. Re-visiting the echo state property. *Neural Networks*, 35:1–9, nov 2012.

A Nowcasting and Multicasting

Nowcasting. We call *nowcasting* the setup in which one constructs a high-frequency proxy for a yet-unobserved target which will be available at the end of the *current* low-frequency period. As such, we construct a nowcast only for horizons $0 < l \leq \kappa - 1$; notice that $l = \kappa$ yields a contemporaneous regression at $t + 1$, while $l = 0$ falls into the category of low-frequency forecasting considered in Section 2.2, hence both these two cases are excluded. The σ-algebras that are used in order to construct nowcasts $\widehat{y}_{t+1,\ell|\kappa}$ are given by

$$\begin{aligned}\mathcal{F}_{t,\ell|\kappa} &= \sigma\left(\left\{y_t, y_{t-1}, \ldots, z_{t,\ell|\kappa},\ z_{t,(\ell-1)|\kappa},\ z_{t,(\ell-2)|\kappa}, \ldots\right\}\right) \\ &= \sigma\left(\left\{y_t, y_{t-1}, \ldots, z_{t+1,-(\kappa-\ell)|\kappa}, z_{t+1,-(\kappa-\ell)+1|\kappa}, z_{t+1,-(\kappa-\ell+2)|\kappa}, \ldots\right\}\right).\end{aligned}$$

The l-steps nowcast for the high-frequency proxy constructed at moments $t, \ell|\kappa$ of the current period for the low-frequency variable which becomes available at $t+1, 0|\kappa \equiv t+1$ is provided by the conditional expectation

$$\widehat{y}^{N}_{t+1,\ell|\kappa} = \mathbb{E}\left[y_{t+1}|\mathcal{F}_{t,\ell}\right].$$

Multicasting. One always aims to construct one-step and multistep forecasts by using all the available information at a given point in time. It is, therefore, natural to compare models by constructing high-frequency nowcasts for the target variable to be released at the end of the current period and its high-frequency proxy forecasts for the next periods. To avoid confusion, we refer to this situation as *multicasting*. More explicitly, provided that the forecaster finds herself at time index $t, s|\kappa$ and is interested in all the forecasts up to some maximal low-frequency horizon $H \geq 1$, for each $1 \leq l \leq H\kappa$ the multicasting scheme yields the following combination:

(a) *Nowcasting* when $0 < l \leq \kappa - 1$ and $\ell = l$: $\widehat{y}^{N}_{t+1,\ell|\kappa} = \mathbb{E}\left[y_{t+1}|\mathcal{F}_{t,\ell}\right]$.

(b) Forecasting when $l > \kappa - 1$:

- *Low-frequency forecasting* if l satisfies $l \bmod \kappa = 0$: $\widehat{y}_{t+h} = \mathbb{E}\left[y_{t+h}|\mathcal{F}_{t}\right]$.
- *High-frequency forecasting* if $l \bmod \kappa \neq 0$: $\mathcal{F}_{t,\ell}$: $\widehat{y}^{H}_{t+h,\ell|\kappa} = \mathbb{E}\left[y_{t+h}|\mathcal{F}_{t,\ell}\right]$.

B ESN Implementation

B.1 Fixed, Expanding and Rolling Window Estimation

Model parameter stability is an important and well-studied question in linear time series analysis. Indeed, identifying and explaining structural breaks play a key role in macroeconomic modeling. To account for this possibility, we compare multiple estimation setups which may reflect possible changes in model parameters.

Suppose again that a sample $Y = (\boldsymbol{y}_2, \boldsymbol{y}_3, \ldots, \boldsymbol{y}_T)^\top \in \mathbb{M}_{T-1,J}$ of targets is available, an initial state $\boldsymbol{x}_0$ is given and regressors $Z = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_{T-1})^\top \in \mathbb{M}_{T-1,K}$ are observed. Additionally, the researcher has available an out-of-sample dataset, $Y^\dagger = (\boldsymbol{y}_{T+1}, \boldsymbol{y}_{T+2}, \ldots, \boldsymbol{y}_{T+S})^\top \in \mathbb{M}_{S,J}$, $Z^\dagger = (\boldsymbol{z}_T, \boldsymbol{z}_{T+1}, \ldots, \boldsymbol{z}_{T+S-1})^\top \in \mathbb{M}_{S,K}$ for $S \geq 1$. We now define the estimation setups which can be used for subsequent forecasting for $h \in \mathbb{N}^+$ steps ahead and can be adjusted for the multi-frequency setup. We consider the following estimation strategies:

(i) **Fixed parameters**: An estimator $\widehat{W}$ is computed strictly over sample observations Y and Z with some penalty λ chosen with data available up to time T. Model parameters are kept fixed when the estimated model is applied to construct out-of-sample forecasts $\widehat{\boldsymbol{y}}_{T+1}, \widehat{\boldsymbol{y}}_{T+2}, \cdots, \widehat{\boldsymbol{y}}_{T+S}$ as out-of-sample regressors $\boldsymbol{z}_T, \boldsymbol{z}_{T+1}, \ldots, \boldsymbol{z}_{T+S-1}$ are added to the information set.

(ii) **Expanding window:** For each out-of-sample time step $s = 0, \dots, S$, define $\widehat{W}_s^{\mathrm{EW}}$ as the estimate computed by "expanding" the sample window up to time $T + s$, given by $Y_s^{\mathrm{EW}} := (\boldsymbol{y}_2, \boldsymbol{y}_3, \dots, \boldsymbol{y}_T, \boldsymbol{y}_{T+1}, \dots, \boldsymbol{y}_{T+s})^\top$ and $Z_s^{\mathrm{EW}} := (\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_{T-1}, \boldsymbol{z}_T, \dots, \boldsymbol{z}_{T+s-1})^\top$. Coefficients $\widehat{W}_s^{\mathrm{EW}}$ are re-estimated and penalty strength λ is re-validated over windows Y_s^{EW}, Z_s^{EW}.

(iii) **Rolling window:** In this setup the within-window sample size is kept fixed across windows – that is, the sample window "rolls" over the data – by defining $\widehat{W}_s^{\mathrm{RW}}$ as the estimate over $Y_s^{\mathrm{RW}} := (\boldsymbol{y}_{2+s}, \boldsymbol{y}_{3+s}, \dots, \boldsymbol{y}_{T+s-1}, \boldsymbol{y}_{T+s})^\top$ and $Z_s^{\mathrm{RW}} := (\boldsymbol{z}_{1+s}, \boldsymbol{z}_{2+s}, \dots, \boldsymbol{z}_{T+s-2}, \boldsymbol{z}_{T+s-1})^\top$ for $s = 0, \dots, S$. Coefficients $\widehat{W}_s^{\mathrm{RW}}$ are re-estimated and penalty strength λ is re-validated over windows Y_s^{RW}, Z_s^{RW}.

In all three strategies, hyperparameters $\boldsymbol{\varphi} := (\alpha, \rho, \gamma, \omega)$ could also be re-tuned on corresponding windows as in Appendix B.2. The fixed-parameter setup is the most rigid one. It builds upon the idea that the initial sample contains sufficient information for correct model estimation and forecasting and that the model parameters are constant. Its theoretical analysis is relatively easy as there is no need to discuss the stability of the penalty and the hyperparameters across sample windows. An expanding window setup is based on the belief that newly available data contains key information to produce forecasts and, therefore, must be continuously incorporated. In essence, forecasters do this when they re-estimate a model at each data release cycle. In the case of a rolling window estimation strategy, one can theoretically handle model changes. Although proper structural break modeling would require a consistent identification of breakpoints, rolling window estimation can potentially accommodate slow drifts in model parameters over time by directly discarding old data, unlike with an expanding window. We do not explore the selection of an optimal window size, which in rolling window estimation has been shown to improve forecasting performance (Inoue et al. (2017)).

B.2 Hyperparameter Tuning

We now propose a general scheme for selection of hyperparameters $\boldsymbol{\varphi} := (\alpha, \rho, \gamma, \omega)$ in (3.7) for a model of the form (3.3)-(3.4). Our approach builds on the idea of leave-one-out cross-validation for time series models. Using a fixed, expanding, or rolling window over the training data, one can always compute the one-step forecasting errors committed by the ESN, given fixed normalized model matrices $(\overline{A}, \overline{C}, \overline{\boldsymbol{\zeta}})$ and a hyperparameter vector $\boldsymbol{\varphi}$. By choosing an appropriate loss function $\boldsymbol{\ell} : \mathbb{R}^J \times \mathbb{R}^J \to \mathbb{R}_+$, $J \in \mathbb{N}^+$, we can thus compute the empirical ESN forecasting error

$$\mathcal{L}_T(\boldsymbol{\varphi}) := \sum_{t=T_0}^{T-1} \boldsymbol{\ell}(\boldsymbol{y}_{t+1}, \widehat{W}_t(\boldsymbol{\varphi})^\top \boldsymbol{x}_t),$$

where $\widehat{W}_t(\boldsymbol{\varphi})$ is the readout coefficients estimator involving data available up to time t and $1 < T_0 < T - 1$ is the minimum number of observations used for fitting. Notice that if $\boldsymbol{\ell}(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_2^2$, $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^J$, then $\mathcal{L}_T(\boldsymbol{\varphi})$ is the cumulative squared error that is minimized in training (modulo a ridge penalty term). Here, however, the interest is not in estimating W, which minimizes $\mathcal{L}_T$, but rather finding the optimal hyperparameter vector

$$\boldsymbol{\varphi}^* \in \underset{\boldsymbol{\varphi} \in [0,1) \times [0,\overline{\rho}] \times [0,\overline{\gamma}] \times [0,\overline{\omega}]}{\arg\min} \mathcal{L}_T(\boldsymbol{\varphi}),$$

where upper bounds $\overline{\rho}$, $\overline{\gamma}$, and $\overline{\omega}$ can be appropriately chosen (in our empirical exercises we use 10 and verify that solutions are never on the boundary). We highlight that to tune $\boldsymbol{\varphi}$ one may choose $\boldsymbol{\ell}$ that is different from the one used in the estimation of the readout coefficients W.

We present the entire hyperparameter optimization routine in Algorithm 1. Note that step (i) might entail re-normalizing inputs and targets at each window t. This setup is general and allows applying

```
Algorithm 1: Hyperparameter tuning
Data: Sample $\boldsymbol{y}_{2:T} = \{\boldsymbol{y}_2, \boldsymbol{y}_3, \ldots, \boldsymbol{y}_T\}$, $\boldsymbol{z}_{1:T-1} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_{T-1}\}$, initial state $\boldsymbol{x}_0$, initial
      guess $\boldsymbol{\varphi}_0$, convergence criterion Crit, maximal algorithm iterations MaxIter. If ridge
      regression is used to estimate $W$, fixed regularization strength $\lambda > 0$.
Result: $\boldsymbol{\varphi}^*$
Fix $T$ and determine the model fit windows for $t = T_0, \ldots, T-1$. Choose whether the ESN
  model is estimated with a fixed or rolling window;
$j = 0$;
while ($\textbf{not}$ Crit) $\textbf{and}$ ($j <$ MaxIter) do
    (i) Given $\boldsymbol{\varphi}_j$, estimate coefficient matrices
            $$\left(\widehat{W}_t(\boldsymbol{\varphi}_j)\right)_{T_0:T-1},$$
      where possibly $\widehat{W}_t(\boldsymbol{\varphi}_j)$ does not depend on $t$, e.g. in the fixed estimation setup;
    (ii) Compute
            $$\mathcal{L}_T(\boldsymbol{\varphi}_j) := \sum_{t=T_0}^{T-1} \boldsymbol{\ell}(\boldsymbol{y}_{t+1}, \widehat{W}_t(\boldsymbol{\varphi}_j)^\top \boldsymbol{x}_t),$$
      the cumulative one-step-ahead forecasting loss;
    (iii) Update $\boldsymbol{\varphi}_{j+1} \leftarrow \boldsymbol{\varphi}_j$ with an appropriate rule (for example, the gradient descent of $\mathcal{L}_T$
      in the direction of $\boldsymbol{\varphi}_j$; in our applications, we use variants L-BFGS-B and pattern search);
    (iv) $j \leftarrow j + 1$, update Crit;
```

any global optimization routine to minimize $\mathcal{L}_T(\boldsymbol{\varphi})$. We construct the loss $\mathcal{L}_T(\boldsymbol{\varphi}_j)$ sequentially, that is by summing squared residuals of the model estimated in step (i) of Algorithm 1 when $\boldsymbol{\ell}$ is a quadratic loss. One can program $\mathcal{L}_T(\boldsymbol{\varphi}_j)$ via TensorFlow so that the gradient can be evaluated by backpropagation in Algorithm 1 (iii). Since there is no guarantee that the objective function is convex or even everywhere smooth, we suggest applying optimizers known to explore the parameter space efficiently. We emphasize that the lack of convexity guarantees is much more consequential for the other benchmarks, in particular for the MIDAS model (see Appendix H.1 for more details).

One issue with the state formulation in (3.3) and thus with the hyperparameter optimization routine in Algorithm 1, is that $\boldsymbol{\varphi}$ can not always be point identified. For example, if one considers identity activation σ and lets $\alpha = \omega = 0$, it is obvious that the ESN model is system isomorphic Grigoryeva and Ortega (2021) to $\boldsymbol{x}_t^* = d\rho \overline{A} \boldsymbol{x}_{t-1}^* + d\gamma \overline{C} \boldsymbol{z}_t$, $\boldsymbol{y}_t = d^{-1} W \boldsymbol{x}_t^* + \boldsymbol{\epsilon}_t$ for all $d \neq 0$. This issue also arises in nonlinear models, for example when σ is taken as a hyperbolic tangent and γ is sufficiently small. Parameter identification in nonlinear models has been extensively studied in semi- and nonparametric cross-sectional regressions. For instance, it is known that in certain setups, point identification requires a proper normalization to be imposed. The interested reader can refer to Section 6.3 of Horowitz (2009) for a discussion in a similar vein regarding nonparametric transformation models. Since often $\omega = 0$ is used, hyperparameter identification can be a significant issue when attempting model tuning. Whenever $\omega = 0$ we propose a helpful reparametrization given by

$$\boldsymbol{x}_t = \alpha \boldsymbol{x}_{t-1} + (1 - \alpha) \sigma \left(\psi \overline{A} \boldsymbol{x}_{t-1} + \overline{C} \boldsymbol{z}_t \right),$$

where $\psi = \rho / \gamma$. This prescription allows decoupling ρ and γ at the cost of the constant input scaling, which may be undesirable whenever one wants to attenuate the nonlinearity induced by the sigmoid map without also reducing the spectral radius.[6] It is immediate to modify the optimization scheme to

[6] One can fix $\overline{C}$ to have a different scaling before optimizing the hyperparameter ψ. However, this amounts to one

deal with the case $\widetilde{\boldsymbol{\varphi}} = (\alpha, \psi)$. In the sequel, we assume that the ESN models are estimated using the approaches proposed in this subsection and use the conventional ESN specification as in (3.3)-(3.4) to discuss the forecasting strategy.

B.3 Cross-validation

Because the initial cross-validation of λ uses an extended sample to try and improve generalization – specifically, our concern is for the fixed estimation setups – we use two slightly different approaches:

- In all setups – fixed, expanding, rolling – the *initial* ridge penalty cross-validation is done on the extended sample (starting January 1st, 1975 instead of January 1st, 1990). We construct 10 folds with 5 out-of-sample observations starting from the end of the sample. Each fold and out-of-sample observation set is re-normalized.

- Only in the expanding and rolling setups, for each subsequent window (the ones that now include at least one testing observation), we use the true sample (starting January 1st, 1990) and construct 5 folds, again with 5 out-of-sample observations. This is done to keep cross-validation computational complexity low and avoid making some folds too small, which could hurt larger MFESN models.

In practice, simple experiments show that there is not much difference between using 5 or 10 folds in the initial cross-validation.

C Performance measures

In this section we define the performance measures used throughout the paper to quantify the quality of forecasts produced by competing models. Suppose that a given model is used to produce a collection of forecasts $\{\widehat{\boldsymbol{y}}_s\}_{s\in S}$, $\widehat{\boldsymbol{y}}_s \in \mathbb{R}^J$. The ordered index set $S = \{s_1, \ldots, s_{|S|}\}$, where $|S|$ is the number of indices in S, can change depending on the setup. For example, in the case of 1-step ahead forecasting, $S = \{T+1, T+2, \ldots, T+\overline{T}\}$ where the 1-step ahead forecasts are constructed using the data up to $T, T+1, \ldots, T+\overline{T}-1$, respectively. For h-step ahead forecasts, we set $S = \{T+h, T+h+1, \ldots, T+\overline{T}-H+h\}$ where H is the maximal forecasting horizon. This ensures that the same number of forecasts are produced at each horizon and can be compared, for example, using the uniform Model Confidence Set (MCS) test described in Appendix D.

MSFE and RMSFE. The *root mean squared forecasting error* is given by

$$\mathrm{MSFE}(S) := \frac{1}{|S|} \sum_{s\in S} \|\boldsymbol{y}_s - \widehat{\boldsymbol{y}}_s\|_2^2,$$

while the *root mean squared forecasting error* is

$$\mathrm{RMSFE}(S) := \sqrt{\mathrm{MSFE}(S)}\,.$$

Cumulative SFE and Cumulative RMSFE. The *cumulative squared forecasting error* is given by the cumulative sum of squared errors. We define for any forecasting index $\tau \in S$,

$$\mathrm{CSFE}(\tau) := \sum_{\substack{s\in S\\ s\leq\tau}} \|\boldsymbol{y}_s - \widehat{\boldsymbol{y}}_s\|_2^2.$$

more *ex ante* model tuning step.

To define the *cumulative RMSFE* for any $\tau \in S$ we first define $\mathcal{T}_l(\tau) := \{s \in S : s \leq \tau\}$ and then write

$$\mathrm{CRMSFE}(\tau) := \sqrt{\frac{1}{|\mathcal{T}_l(\tau)|} \sum_{s \in \mathcal{T}_l(\tau)} \|\boldsymbol{y}_s - \widehat{\boldsymbol{y}}_s\|_2^2}.$$

Ahead RMSFE and 1-Year-Ahead RMSFE. If one wants to the evaluate performance *ahead* of a certain point of time, it is also possible to define the *ahead RMSFE*,

$$\mathrm{AheadRMSFE}(\tau) := \sqrt{\frac{1}{|\mathcal{T}_u(\tau)|} \sum_{s \in \mathcal{T}_u(\tau)} \|\boldsymbol{y}_s - \widehat{\boldsymbol{y}}_s\|_2^2},$$

where we introduce $\mathcal{T}_u(\tau) := \{s \in S : s \geq \tau\}$.

In the special case where the indices of S are associated to dates, one may also compare performance after a given amount of time has passed from the current time index. For example, one may evaluate how performance degrades after model estimation when parameters are fixed and not updated. In our empirical exercises where $\{y_t\}_{t\in\mathbb{Z}}$, $y_t \in \mathbb{R}$, is a quarterly GDP series, we can define the *1-year-ahead RMSFE* as

$$\mathrm{1YAheadRMSFE}(\tau) := \sqrt{\frac{1}{|\mathcal{T}_u(\tau+4)|} \sum_{s \in \mathcal{T}_u(\tau+4)} (y_s - \widehat{y}_s)^2}.$$

D Uniform Multi-Horizon MCS

We now give details on the implementation of the Uniform Multi-Horizon MCS test for the multi-horizon forecast comparisons in our empirical analysis. Our procedure follows closely the one originally provided by Quaedvlieg (2021): we provide R code for our functions, while the author's code was originally developed in the Ox programming language.

A main difference is that we prefer to use a Bartlett kernel to compute the sample uSPA statistic, whereas Quaedvlieg (2021) uses the quadratic spectral (QS) kernel of Andrews (1991). Our main reason for this choice is that the QS kernel features non-zero weights for all lags, while the Bartlett kernel has finite support. This is especially important since we only have a few forecasts in our case; thus, higher lag autocovariances between model losses can only be poorly estimated. It means our uMCS procedure implements the standard Newey-West HAC estimator. We use $B = 100$ replications for the outer and inner bootstraps. Finally, the inner bootstrap critical value is set at $\alpha = 0.1$.

E MIDAS

A state-of-the-art methodology for incorporating data of heterogeneous frequencies into one model is the MIDAS framework developed in Ghysels et al. (2004, 2007). Here we present MIDAS in its dynamic form, which allows the inclusion of target series autoregressive lags. We use our temporal notation given in Definition 2.1 throughout.

If the MIDAS model contains only one explanatory variable (z_r) with frequency multiplier κ, then it can be written as

$$y_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \beta \sum_{k=0}^{K} \varphi(\boldsymbol{\theta}, k) z_{t,-k|\kappa} + \epsilon_t, \tag{E.1}$$

where α_0 is a constant term, $\{\alpha_i\}_{i=1}^{p}$ are the autoregressive parameters, β is a scaling parameter, $\{\varphi(\boldsymbol{\theta}, k)\}_{k=0}^{K}$ are the MIDAS weights given as a parametric function of lag k and underlying parameter vector $\boldsymbol{\theta} \in \mathbb{R}^q$, and (ϵ_t) is a martingale difference process relative to the filtration $\{\mathcal{F}_t\}$ generated by $\{y_{t-1-j}, x_{t-j}, \ldots, x_{t-j,-K|\kappa}, \epsilon_{t-1-j} \mid j \geq 0\}$ and such that $\mathbb{E}[\epsilon_t^2] = \sigma_\epsilon^2 < \infty$.

The MIDAS weighting scheme is the core innovation of the model. It borrows parsimony from distributed lag models in the sense that, even if K is large, the vector $\boldsymbol{\theta} \in \mathbb{R}^q$ is usually restricted to contain only a handful of parameters. This greatly reduces the number of coefficients that need to be estimated, and a nonlinear least-squares estimator $\widehat{\boldsymbol{\theta}}$ can be readily implemented. There are alternative formulations of the MIDAS framework where $\varphi(\boldsymbol{\theta}, k) = \theta_k$ so that the above reduces to a full linear model, the so-called unrestricted MIDAS or U-MIDAS (Foroni and Marcellino, 2011).

We follow the literature and use the most commonly applied weighting scheme that is based on the exponential Almon weighting polynomial map $\varphi : \mathbb{R}^q \times \mathbb{N}^+ \longrightarrow \mathbb{R}^+$ (see Almon (1965) for more details). In particular, for the case of $q = 2$, the two-parameter Almon weighting polynomial is given by

$$\varphi(\boldsymbol{\theta}, k) = \varphi((\theta_1, \theta_2), k) = \exp(\theta_1 k + \theta_2 k^2), \;\; k \in \mathbb{N}^+.$$

Since Almon weights need not sum up to a given constant for different values of θ_1 and θ_2, it is often common to consider the normalized Almon scheme

$$\overline{\varphi}(\boldsymbol{\theta}, k) = \frac{\exp(\theta_1 k + \theta_2 k^2)}{\sum_{k=0}^{K} \exp(\theta_1 k + \theta_2 k^2)}, \tag{E.2}$$

which together with (E.1) allows to treat β as a rescaling constant.

Let us now consider a more general model suitable for situations where time series of different frequencies are available and must be integrated into the MIDAS equation. Consider the case of L regressor time series. We assume that the lth time series is sampled at a frequency κ_l and contains observations $(z^{(l)}_{t,s|\kappa_l})_{t,s}$ with $z^{(l)}_{t,s|\kappa_l} \in \mathbb{R}$ for all $t \in \mathbb{Z}$ and $s \in \{0, \ldots, \kappa_l - 1\}$. It happens frequently in practice that κ_l, $l \in [L]$ takes values from a small set of integers. For example, in the case of yearly, quarterly, and monthly data $\kappa_l \in \{1, 4, 12\}$ even though L could be very large (often, hundreds or thousands of series might be of interest). The MIDAS model explaining low-frequency target variable y_t with L regressors $(z^{(l)}_{t,s|\kappa_l})_{t,s}$, $l \in [L]$ can be written as follows

$$y_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{l=1}^{L} \beta_l \sum_{k=0}^{K_l} \varphi(\boldsymbol{\theta}_l, k) z^{(l)}_{t,-k|\kappa_l} + \epsilon_t, \tag{E.3}$$

where the martingale difference process (ϵ_t) is relative to the filtration generated by sets as in (E.1), modified to include all the considered L regressors.

The MIDAS framework produces forecasts of the chosen target variable at the low frequency of the target. Yet, due to the MIDAS multi-frequency structure, *nowcasting* is also a straightforward exercise: if, for example, the high-frequency regressor is a single series (z_r) with frequency multiplier κ, one can construct exactly κ regression equations – one for each high-frequency release within a low-frequency period – and use these to produce high-frequency nowcasts of the target. In fact, due to the convenience of the MIDAS model, it is easy to define high-frequency regression specifications to study high-frequency forecasts and multicasts (see Section 2.2 and Appendix A).

In practice, implementing (E.3) demands some care. From a computational point of view, as long as the relevant regression matrices can be constructed, estimation amounts to a nonlinear least-squares problem, which can be readily solved. In Appendix E.1 and Appendix H.1 we discuss the technical aspects of our MIDAS implementation in more detail. One of the important issues of the MIDAS estimation is the non-convexity of the nonlinear least squares loss as a function of parameters. Often, a practitioner may obtain different estimation results depending on initialization and, more importantly, those that lead to a different quality of forecasts. Other weighting schemes that allow for convex estimation problems can be used. For example, one may adopt the Almon lag polynomial parametrization (Ghysels, 2016, Pettenuzzo et al., 2016) using a discrete polynomial basis for the transformation of high-frequency regressors. This specification allows for standard OLS estimation but requires careful choice of the polynomial order hyperparameter.

Another crucial disadvantage of the MIDAS specification is that practical implementations can be very challenging. This is caused mainly by the ragged edges of the "raw" macroeconomic data, incomplete observations, and uneven sampling frequencies. The relative inflexibility of MIDAS regression lag specifications makes integrating daily and weekly data at true calendar frequencies (i.e. without interpolation or aggregation) very complex.[7] State-space models effectively mitigate these issues.

Finally, as shown in Bai et al. (2013), exponential Almon MIDAS regressions have inherent connections to dynamic factor models, which we discuss in the next section. When the factor structure is not trivial, MIDAS can, however, only yield a finite-order approximation to a DFM data-generating process. Furthermore, Bai et al. (2013) prove that in well-identified setups the mapping between exponential Almon and factor model coefficients is highly nonlinear. Given the robustness evaluations in Appendix H.1, in practice, it appears hard to formally relate MIDAS and DFM forecasting performance.

E.1 MIDAS Implementation

While the MIDAS regression framework is straightforward to discuss in terms of equations, some care must be taken when implementing it computationally. A key assumption that can be imposed is that the integer frequencies $\boldsymbol{\kappa} := \{\kappa_1, \ldots, \kappa_L\}$ of L regressors are such that $\kappa_{\max} := \max(\boldsymbol{\kappa})$ is a multiple of each of the κ_l, $l \in [L]$. In this case, MIDAS parameter estimation can be written in matrix form, which allows for efficient numerical implementation, which we spell out in the following paragraphs.

Let $q_l = \kappa_{\max}/\kappa_l$, $l \in [L]$ denote the frequency ratios and define $\boldsymbol{y} := (y_1, y_2, \ldots, y_T)^\top$ the vector of target observations, where T is the sample length in reference time scale. Additionally, let $\boldsymbol{z}^{(l)} := (z_1^{(l)}, z_2^{(l)}, \ldots, z_{T_l}^{(l)})^\top$ be $T_l = T \cdot \kappa_l$ long vector which consists of observations of the lth covariate $z^{(l)}$ released with frequency κ_l. For the parameters of the MIDAS model in (E.3) to be identifiable, we assume that

$$T > 1 + p + \sum_{l=1}^{L} \left\lceil \frac{K_l}{\kappa_l} \right\rceil .$$

Since $\kappa_{\max}$ is a multiple of each of the L frequencies, for each series we introduce

$$\mathbf{Y} = \boldsymbol{y} \otimes \boldsymbol{i}_{\kappa_{\max}}, \quad \mathbf{Z}^{(l)} = \boldsymbol{z}^{(l)} \otimes \boldsymbol{i}_{q_l},$$

where $\boldsymbol{i}_{q_l}$ and $\boldsymbol{i}_{\kappa_{\max}}$ are vectors of ones of lengths q_l and $\kappa_{\max}$, respectively. In the absence of missing observations, we have that $\mathbf{Y}, \mathbf{Z}^{(l)} \in \mathbb{R}^{T_{\max}}$ with $T_{\max} = T \cdot \kappa_{\max}$ observations. We now construct preliminary regression matrices such that their maximal rows number is $T_{\max}$ without accounting for the lags structure of both the target (autoregressive lags) and regressors (MIDAS lags) and we

[7] One could set up a MIDAS regression with the full yearly calendar of weeks and working days as lags. However, ragged edges arising from holidays, leap years, etc. would still be non-trivial to handle coherently without resorting to downsampling, data re-alignment, or interpolation.

introduce zeros where no observations are available.[8] Define for $p \geq 1$ and for $K_l \geq 0$

$$Y_p = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ y_1 & 0 & \cdots & 0 \\ y_2 & y_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ y_{T-2} & y_{T-3} & \cdots & y_{T-p-1} \\ y_{T-1} & y_{T-2} & \cdots & y_{T-p} \end{pmatrix} \otimes \boldsymbol{i}_{\kappa_{\max}} \quad \text{and} \quad Z_{K_l} = \begin{pmatrix} z_1^{(l)} & 0 & \cdots & 0 \\ z_2^{(l)} & z_1^{(l)} & \cdots & 0 \\ z_3^{(l)} & z_2^{(l)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ z_{T_l-1}^{(l)} & z_{T_l-2}^{(l)} & \cdots & z_{T_l-K_l-1}^{(l)} \\ z_{T_l}^{(l)} & z_{T_l-1}^{(l)} & \cdots & z_{T_l-K_l}^{(l)} \end{pmatrix} \otimes \boldsymbol{i}_{q_l}.$$

In the special case $p = 0$ (the MIDAS model, in this case, is called *static*, since it does not contain an autoregressive term) we take Y_p as empty. We now follow by noticing that one should not use $Y_p \in \mathbb{M}_{T_{\max},p}$ and $Z_{K_l} \in \mathbb{M}_{T_{\max},K_l+1}$ as autoregressive and mixed-frequency regression matrices, respectively, since some observations are missing. To overcome this we introduce

$$n := \max\left\{p, \left\lceil \frac{K_1}{q_1} \right\rceil, \ldots, \left\lceil \frac{K_L}{q_L} \right\rceil \right\} \cdot \kappa_{\max} \tag{E.4}$$

and the so-called upper truncation (selection) matrix

$$U = \begin{pmatrix} \mathbb{O}_{T_{\max}-n+1,n-1} & \mathbb{I}_{T_{\max}-n+1,T_{\max}-n+1} \end{pmatrix}$$

with which we obtain the following required response vector and regression matrices

$$\begin{aligned} \mathbf{Y}^{\text{resp}} &= U\mathbf{Y} \in \mathbb{R}^{T_{\max}-n+1}, \\ Y_p^{\text{reg}} &= UY_p \in \mathbb{M}_{T_{\max}-n+1,p}, \\ Z_{K_l}^{\text{reg}} &= UZ_{K_l} \in \mathbb{M}_{T_{\max}-n+1,K_l+1}, \\ \mathbf{Z}^{\text{reg}} &= \begin{pmatrix} Y_p^{\text{reg}} & Z_{K_1}^{\text{reg}} & \cdots & Z_{K_L}^{\text{reg}} \end{pmatrix} \in \mathbb{M}_{T_{\max}-n+1,\sum_{l=1}^{L} K_l+L}, \end{aligned}$$

where Y_p^{reg} is empty whenever $p = 0$.

We can now observe that $\mathbf{Y}^{\text{resp}}$ and $\mathbf{Z}^{\text{reg}}$ are sufficient to construct all MIDAS forecasting and nowcasting regressions. In practice, some care needs to be taken to make sure that data is correctly aligned: for example, in the case of nowcasting exercise regressors in $\mathbf{Z}^{\text{reg}}$ and targets in $\mathbf{Y}^{\text{resp}}$ have to be aligned differently than in the case of forecasting exercises. Provided the aligned data is executed correctly, the estimation of MIDAS parameters can be carried out efficiently. An important thing to mention is that the truncation with the help of s in (E.4) may be too restrictive, as it may lead to excluding up to $K_{\max} - 1$ rows from $\mathbf{Z}^{\text{reg}}$ that could be used for estimation. This can be avoided at the time of implementation. In our repository available at [the address removed for anonymous submission] we consider this detail and exclude from the final regression matrices only those rows which cannot be used due to the lag requirements in the model. We warn the reader that this comes at a cost, namely the codes are lengthier and less elegant.

F Mixed-frequency DFM

Macroeconomic modeling based on dynamic factor models has been popular since their introduction in Geweke (1977) and Sargent et al. (1977). The proposition of DFMs is that a low-dimensional latent

[8] At the time of implementation of this procedure in any convenient coding environment it is more natural to introduce placeholders instead and to perform the subsequently discussed truncation via matrix manipulation rather than by using matrix multiplication.

factor $(\boldsymbol{f}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{f}_t \in \mathbb{R}^d$, drives a high-dimensional observable stochastic process $(\boldsymbol{y}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{y}_t \in \mathbb{R}^n$. We consider a time-inhomogeneous state-space model with dynamics

$$\boldsymbol{f}_{t+1}|\boldsymbol{f}_{1:t}, \boldsymbol{y}_{1:t} \sim h_{t+1,\boldsymbol{\theta}}(\cdot|\boldsymbol{f}_t) \tag{F.1}$$

$$\boldsymbol{y}_{t+1}|\boldsymbol{f}_{1:t+1}, \boldsymbol{y}_{0:t} \sim g_{t+1,\boldsymbol{\theta}}(\cdot|\boldsymbol{f}_{t+1}) \tag{F.2}$$

for some time-dependent state transition kernels $h_{t,\boldsymbol{\theta}}$ and observation densities $g_{t,\boldsymbol{\theta}}$ and some parameter vector $\boldsymbol{\theta}$ in a parameter space Θ. A common example in the literature (see Watson and Engle (1983) for more details) is linear Gaussian factor models with time-inhomogeneous state transitions that can be represented as

$$\boldsymbol{f}_{t+1} = A_{\boldsymbol{\theta}}\boldsymbol{f}_t + R_{\boldsymbol{\theta}}\boldsymbol{u}_t \tag{F.3}$$

$$\boldsymbol{y}_{t+1} = \Lambda_{t+1,\boldsymbol{\theta}}\boldsymbol{f}_{t+1} + S_{t+1,\boldsymbol{\theta}}\boldsymbol{w}_{t+1} \tag{F.4}$$

with state transition matrix $A_{\boldsymbol{\theta}} \in \mathbb{M}_d$, time-dependent factor loading matrices $\Lambda_t \in \mathbb{M}_{n,d}$, and where $\boldsymbol{u}_t$ and $\boldsymbol{w}_t$ are independent Gaussian vectors with zero mean and identity covariance matrix of dimension p and n, respectively, and $R_{\boldsymbol{\theta}}$ and $S_{t,\boldsymbol{\theta}}$ re matrices of appropriate dimensions. It is often assumed that the dimension p of the state noise vector $\boldsymbol{u}_t$ is smaller than the latent state space dimension d, which implies that $R_{\boldsymbol{\theta}}R_{\boldsymbol{\theta}}^\top$ is rank deficient, such as for AR(p) factor dynamics (Stock and Watson, 2016, Forni et al., 2005, Doz et al., 2011). In this case, $d = kp$ for some $k \in \mathbb{N}^+$,

$$A_{\boldsymbol{\theta}} = \begin{pmatrix} A_{\boldsymbol{\theta}}^{(1)} & A_{\boldsymbol{\theta}}^{(2)} & \cdots & A_{\boldsymbol{\theta}}^{(p-1)} & A_{\boldsymbol{\theta}}^{(p)} \\ \mathbb{I}_k & \mathbb{O}_k & \cdots & \mathbb{O}_k & \mathbb{O}_k \\ \mathbb{O}_k & \mathbb{I}_k & \cdots & \mathbb{O}_k & \mathbb{O}_k \\ \vdots & & \ddots & & \vdots \\ \mathbb{O}_k & \mathbb{O}_k & \cdots & \mathbb{I}_k & \mathbb{O}_k \end{pmatrix}, \quad \Lambda_{t,\boldsymbol{\theta}} = \begin{pmatrix} \Lambda_{t,\boldsymbol{\theta}}^{(1)} & \Lambda_{t,\boldsymbol{\theta}}^{(2)} & \cdots & \Lambda_{t,\boldsymbol{\theta}}^{(p)} \end{pmatrix} \tag{F.5}$$

with $A_{\boldsymbol{\theta}}^{(j)} \in \mathbb{M}_k$ and $\Lambda_{t,\boldsymbol{\theta}}^{(j)} \in \mathbb{M}_{n,k}$. Setting $\boldsymbol{f}_t = (\boldsymbol{v}_t^\top, \boldsymbol{v}_{t-1}^\top, \ldots, \boldsymbol{v}_{t-p+1}^\top)^\top$ implies that $(\boldsymbol{v}_t)_{t\in\mathbb{Z}}$ is a k-dimensional AR(p) process and it is commonly assumed that $\Lambda_{t,\boldsymbol{\theta}}^{(j)} = \mathbb{O}_{n,k}$ for $j > 1$. Let the initial state $\boldsymbol{f}_0$ be distributed according to ν. The joint density of the latent path $\boldsymbol{f}_{0:T}$ and observations $\boldsymbol{y}_{0:T}$ is then

$$p_{\boldsymbol{\theta},\nu}(\boldsymbol{f}_{0:T}, \boldsymbol{y}_{0:T}) = \nu(\boldsymbol{f}_0) g_{0,\boldsymbol{\theta}}(\boldsymbol{y}_0|\boldsymbol{f}_0) \prod_{t=1}^{T} h_{t,\boldsymbol{\theta}}(\boldsymbol{f}_t|\boldsymbol{f}_{t-1}) g_{t,\boldsymbol{\theta}}(\boldsymbol{y}_t|\boldsymbol{f}_t),$$

while the marginal likelihood of $\boldsymbol{y}_{0:T}$ is $p_{\boldsymbol{\theta},\nu}(\boldsymbol{y}_{0:T}) = \int p_{\boldsymbol{\theta},\nu}(\boldsymbol{f}_{0:T}, \boldsymbol{y}_{0:T}) \mathrm{d}\boldsymbol{f}_{0:T}$. Popular procedures for learning the static parameters $\boldsymbol{\theta} \in \Theta$ are based on gradient descent of the negative log-likelihood function $\ell_T \colon \Theta \to \mathbb{R}, \boldsymbol{\theta} \mapsto -\log p_{\boldsymbol{\theta},\nu}(\boldsymbol{y}_{0:T})$ or on the Expectation Maximization (EM) algorithm introduced in Dempster et al. (1977). We consider here gradient descent algorithmsbased on a sequence of step sizes $\gamma_k > 0$, that update the model parameters based on iterations of the form

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \gamma_{k+1} \nabla_{\boldsymbol{\theta}} \ell_T(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k},$$

for $k \in \mathbb{N}^+$.[9]

[9] Consistency of the maximum log-likelihood estimate for the dynamics (F.3)-(F.4) in the time-homogeneous case has been established for instance in Douc et al. (2011) under regularity assumptions, including, for instance, the full-rank of the noise covariance matrix $S_{\boldsymbol{\theta}}$, of the controllability matrix $C_{\boldsymbol{\theta}} = \big(R_{\boldsymbol{\theta}} | A_{\boldsymbol{\theta}} R_{\boldsymbol{\theta}} | \cdots | A_{\boldsymbol{\theta}}^{d-1} R_{\boldsymbol{\theta}}\big)$, and of the observability matrix $O_{\boldsymbol{\theta}} = \big(\Lambda_{\boldsymbol{\theta}}^\top | (\Lambda_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}})^\top | \cdots | (\Lambda_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^{d-1})^\top\big)^\top$. It is also possible to consider an online learning setting using a recursive decomposition of the score function as in LeGland and Mevel (1997). For general latent state dynamics (F.1) and observation densities (F.2) that can be non-linear with non-Gaussian noise, particle filtering algorithms are often utilized that make use of particle approximations in gradient-descent or EM learning approaches, see for instance Kantas et al. (2015).

Assuming a linear Gaussian setting where the transition density of the latent factor process is given by (F.5) to yield an AR(p) process $(\boldsymbol{v}_t)_{t\in\mathbb{Z}}$, $\boldsymbol{v}_t = (v_{1,t}, \dots, v_{k,t})^\top$, there remains some flexibility as to how the linear mappings

$$\mathsf{Agg}_{\boldsymbol{\theta},L} \colon \mathbb{M}_{k,p} \to \mathbb{R}, \quad (\boldsymbol{v}_{t-p+1} \dots, \boldsymbol{v}_t) \mapsto (\Lambda_{t,\boldsymbol{\theta}} \boldsymbol{f}_t)_i = (\Lambda_{t,\boldsymbol{\theta}})_{i,\cdot} \begin{bmatrix} \boldsymbol{v}_t \\ \boldsymbol{v}_{t-1} \\ \vdots \\ \boldsymbol{v}_{t-p+1} \end{bmatrix}$$

for some lag parameter $L \le p$ are chosen for each dimension $i \in [n]$.[10] We call this linear mapping $\mathsf{Agg}_{\boldsymbol{\theta},L}$ an *aggregation function* and consider specific examples below that yield different models for the factor loadings matrices $\Lambda_{t,\boldsymbol{\theta}}$. Notice that our aggregation functions are linear with respect to the latent factors in contrast to the non-linear approaches introduced in Proietti and Moauro (2006) that require approximations, such as resorting to extended Kalman filtering techniques.

Example F.1 (Stock aggregation) For $i \in [n]$, let $\boldsymbol{\beta}_i = (\beta_{i1}, \dots, \beta_{ik}) \in \mathbb{R}^k$ and consider

$$\mathsf{Agg}^{\mathrm{S}}_{\boldsymbol{\theta},1}(\boldsymbol{v}_{t-p+1} \dots, \boldsymbol{v}_t)_i = \sum_{m=1}^{k} \beta_{im} v_{m,t},$$

with $\boldsymbol{\theta} = \boldsymbol{\beta}_i$.

Example F.2 (Almon-Lag aggregation) For $i \in [n]$, let $\boldsymbol{\beta}_i \in \mathbb{R}^k$, $\boldsymbol{\psi}_i \in \mathbb{R}^{2k}$ and consider

$$\mathsf{Agg}^{\mathrm{AL}}_{\boldsymbol{\theta},L}(\boldsymbol{v}_{t-p+1} \dots, \boldsymbol{v}_t)_i = \sum_{m=1}^{k} \beta_{im} \sum_{\ell=0}^{L-1} \overline{\varphi}(\psi_{im}, \ell) v_{m,t-\ell},$$

with $\boldsymbol{\theta} = (\boldsymbol{\beta}_i, \boldsymbol{\psi}_i, \boldsymbol{\beta}_i, \boldsymbol{\psi}_i)$ and Almon-Lag weights $\overline{\varphi}$ given in (E.2).

Example F.3 (Trigonometric aggregation) For $i \in [n]$, let $\boldsymbol{\beta}_i \in \mathbb{R}^k$, and for $K \in \mathbb{N}$, let $\boldsymbol{\lambda} \in \mathbb{R}^K_+$, $\boldsymbol{\omega} \in [0,1]^K$, $\boldsymbol{\gamma} \in [-\pi, \pi]^K$ and $\tau \in \mathbb{R}_+$. Define

$$\mathsf{Agg}^{\sin}_{\boldsymbol{\theta},L}(\boldsymbol{v}_{t-p+1} \dots, \boldsymbol{v}_t)_i = \sum_{m=1}^{k} \beta_{im} \sum_{\ell=0}^{L-1} \overline{a}_p(\boldsymbol{\lambda}, \boldsymbol{\omega}, \boldsymbol{\gamma}, \tau, \ell) v_{m,t-\ell},$$

with $\boldsymbol{\theta} = (\boldsymbol{\beta}_i, \boldsymbol{\lambda}, \boldsymbol{\omega}, \boldsymbol{\gamma}, \tau)$ and

$$\overline{a}_p(\boldsymbol{\lambda}, \boldsymbol{\omega}, \boldsymbol{\gamma}, \tau, \ell) = \frac{\exp\left(\frac{1}{\tau} \sum_{j=1}^{K} \lambda_j^2 \cos(2\pi\omega_j \ell + \gamma_j)\right)}{\sum_{\ell'=0}^{p-1} \exp\left(\frac{1}{\tau} \sum_{j=1}^{K} \lambda_j^2 \cos(2\pi\omega_j \ell' + \gamma_j)\right)}.$$

This aggregation scheme is motivated by self-attention models (we refer the reader to Bahdanau et al. (2014), Vaswani et al. (2017) for more details), but to retain linearity only considers a relative positional encoding with a Toeplitz structure. Observe that the aggregation parameters are shared across all n dimensions in contrast to the Almon lag scheme in Example F.2.

Some authors (see for example Mariano and Murasawa (2003), Bańbura and Modugno (2014)) have imposed different restrictions on the form of the factor loadings matrices or aggregation function, particularly for one-dimensional mixed-frequency factor models of quarterly GDP growth rates and

[10] The Markovian representation (F.1)-(F.2), that is, the companion form, is based on the autoregressive order p, however, one can set $A^{(\ell)}_{\boldsymbol{\theta}} = \mathbb{O}_k$ for $\ell > p$.

monthly covariates, which are motivated by approximations of growth rates. We do not pursue this additional restriction in this work.

Kalman filtering techniques have been used routinely for handling missing observations in multi-frequency DFMs, see Harvey et al. (1998). In this work, we leverage modern auto-differentiation libraries (Abadi et al., 2016, Dillon et al., 2017) to compute the gradient of the log-likelihood based on Kalman filtering formulae and estimate the static parameters $\boldsymbol{\theta}$ by gradient ascent of the log-likelihood. For alternative estimation approaches using EM that could be extended to this setting, we refer the reader to Bańbura and Modugno (2014). Nonlinear or non-Gaussian dynamic factor models in a mixed frequency setting have been considered in Gagliardini et al. (2017), Leippold and Yang (2019) that rely on particle filtering methods in conjunction with backward simulation algorithms as in Godsill et al. (2004), while Schorfheide et al. (2018) consider a Bayesian approach using particle MCMC (see Andrieu et al. (2010)). Such approaches can become computationally expensive and are not considered for benchmarking purposes.

While previous mixed-frequency DFMs (see Mariano and Murasawa (2003), Bańbura and Modugno (2014) for a more thorough discussion) often consider time series which are sampled at two frequencies, we introduce here a flexible mixed-frequency DFM that describes $L \in \mathbb{N}^+$ collections of distinct time series sampled at frequencies $\{\kappa_1, \ldots, \kappa_L\}$ and each consisting of $\{n_1, \ldots, n_L\}$ series, respectively. In the same setting as in Section 4, each group of n_l, $l \in [L]$, time series sampled at frequency κ_l contains observations $(\boldsymbol{y}^{(l)}_{t,s|\kappa_l})$ with $\boldsymbol{y}^{(l)}_{t,s|\kappa_l} \in \mathbb{R}^{n_l}$ for all $t \in \mathbb{Z}$ and $s \in \{0, \ldots, \kappa_l - 1\}$. Let $\kappa_{\max} = \max_l \kappa_l$. Suppose that the latent factor dynamics are updated at the highest sampling frequency based on the linear transition

$$\boldsymbol{f}_{t,s+1|\kappa_{\max}} = A_{\boldsymbol{\theta}} \boldsymbol{f}_{t,s|\kappa_{\max}} + R_{\boldsymbol{\theta}} \boldsymbol{u}_{t,s+1|\kappa_{\max}}, \tag{F.6}$$

where

$$\boldsymbol{f}_{t,s|\kappa_{\max}} = \left(\boldsymbol{v}^\top_{t,s|\kappa_{\max}}, \ldots, \boldsymbol{v}^\top_{t,s-p+1|\kappa_{\max}}\right)^\top,$$

with $A_{\boldsymbol{\theta}}$ given in (F.5) for the special case where $A^{(\ell)}_{\boldsymbol{\theta}} = \mathbb{O}_k$ for $\ell \geq 2$, $p = \kappa_{\max}$ and

$$A^{(1)}_{\boldsymbol{\theta}} = \bar{A} \frac{\rho}{\max\left\{\rho, |\lambda_1(\bar{A})|\right\}}$$

with parameters $\rho \in (0,1)$, $\bar{A} \in \mathbb{M}_k$ and with $\lambda_1(\bar{A})$ denoting the largest eigenvalue of $\bar{A}$. In the simplified scenario of first-order autoregressive dynamics, we parameterize $R_{\boldsymbol{\theta}} \in \mathbb{M}_k$ to be positive definite and diagonal and $\boldsymbol{u}_{t,s+1|\kappa_{\max}}$ are a sequence of IID k-dimensional standard Gaussian variables.

Notice that Kalman filtering formulas yield the first moment

$$\widehat{\boldsymbol{f}}_{t,s|\kappa_{\max}} = \mathbb{E}\left[\boldsymbol{f}_{t,s|\kappa} \middle| \boldsymbol{y}_{1,0|\kappa_{\max}}, \ldots, \boldsymbol{y}_{t,s|\kappa_{\max}}\right]$$

recursively online, see for example Appendix F.1 for details in the general time-inhomogeneous case. Due to the linearity in (F.6), for any $h \in \mathbb{N}$,

$$\widehat{\boldsymbol{f}}_{t,s+h|\kappa_{\max}} = \mathbb{E}\left[\boldsymbol{f}_{t,s+h|\kappa_{\max}} \middle| \boldsymbol{y}_{1,0|\kappa_{\max}}, \ldots, \boldsymbol{y}_{t,s|\kappa_{\max}}\right] = A^h_{\boldsymbol{\theta}} \widehat{\boldsymbol{f}}_{t,s|\kappa_{\max}}.$$

Furthermore, from the linearity of the aggregation scheme, we obtain the forecasts for any $s, h \in \mathbb{N}$,

$$\mathbb{E}\left[\boldsymbol{y}^{(l)}_{t,s+h|\kappa_l} \middle| \boldsymbol{y}_{1,0|\kappa_l}, \ldots, \boldsymbol{y}_{t,s|\kappa_l}\right] = \mathsf{Agg}_{\boldsymbol{\theta}^{(l)}}\left(\widehat{\boldsymbol{f}}_{t,(s+h)q_l|\kappa_{\max}}\right), \tag{F.7}$$

where $q_l = \kappa_{\max}/\kappa_l$ (c.f. Section 4.1) and $\mathsf{Agg}_{\boldsymbol{\theta}^{(l)}}$ is the aggregation scheme for frequency l. We observe that there is a single latent factor process that describes the observations at all frequencies, in contrast, for instance, to hierarchical Hidden Markov Models (HMM) (Hihi and Bengio, 1995) where the latent variables evolve a priori at different time-scales. This time evolution of states is similar to the SMFESN models also developed in this paper.

It is possible to write the following mixed-frequency DFM model in Example F.4 as a general time-inhomogeneous state-space system (F.1)-(F.2) by suitably parameterizing the time dependencies in the aggregation matrices. We provide more details on implementing our mixed frequency DFM in Appendix F.1 below. The standard Kalman filtering recursions utilized therein for parameter estimation have a cubic complexity in the dimension d or n of the Markovian factor process $\boldsymbol{f}$ or the observation process $\boldsymbol{y}$, respectively, at every time step. The marginal log-likelihood is optimized based on stochastic gradient methods with adaptive step sizes (Kingma and Ba, 2014) and is generally not a concave function of the parameter values.[11]

Example F.4 (Quarterly-Monthly-Daily DFM Model) We consider $n_{(\mathtt{6d})}$ time series that result from averaging daily time series over 6 days, yieling 12 observations per quarter that are denoted as $\boldsymbol{y}^{(\mathtt{6d})}$. Furthermore, we consider $n_{(\mathtt{m})}$ monthly $\boldsymbol{y}^{(\mathtt{m})}$ as well as $n_{(\mathtt{q})}$ quarterly time series $\boldsymbol{y}^{(\mathtt{q})}$. We let $\kappa_{\max} = 72/6 = 12$ and update the k-dimensional latent factor process every 6 days in sync with $\boldsymbol{y}^{(\mathtt{6d})}$. We aggregate 6 days to significantly decrease the computational cost of the factor model. The latent factors are assumed to have the VAR(1) dynamics,[12]

$$\boldsymbol{v}_{t,s+1|12} = A^{(1)}\boldsymbol{v}_{t,s|12} + R\boldsymbol{u}_{t,s+1|12},$$

for any $s, t \in \mathbb{N}$, $A^{(1)} \in \mathbb{M}_{k,k}$ $R \in \mathbb{M}_k$ and IID k-dimensional standard Gaussian variables $\boldsymbol{u}_{t,s|12}$. The averaged daily data is described by

$$\boldsymbol{y}^{(\mathtt{6d})}_{t,s|12} = \beta^{(\mathtt{6d})}\boldsymbol{v}_{t,s|12} + S^{(\mathtt{6d})}\boldsymbol{w}^{(\mathtt{6d})}_{t,s|12}$$

for any $s, t \in \mathbb{N}$, $\beta^{(\mathtt{6d})} \in \mathbb{M}_{n_{(\mathtt{6d})},k}$, $S^{(\mathtt{6d})} \in \mathbb{M}_{n_{(\mathtt{6d})}}$ and IID $n_{(\mathtt{6d})}$-dimensional standard Gaussian variables $\boldsymbol{w}^{(\mathtt{6d})}_{t,s|12}$. The monthly data in the stock aggregation scheme is modeled as

$$\boldsymbol{y}^{(\mathtt{m})}_{t,s|3} = \beta^{(\mathtt{m})}\boldsymbol{v}_{t,4s|12} + S^{(\mathtt{m})}\boldsymbol{w}^{(\mathtt{m})}_{t,s|3},$$

with $\beta^{(\mathtt{m})} \in \mathbb{M}_{n_{(\mathtt{m})},k}$, $S^{(\mathtt{m})} \in \mathbb{M}_{n_{(\mathtt{m})}}$ and IID $n_{(\mathtt{m})}$-dimensional standard Gaussian variables $\boldsymbol{w}^{(\mathtt{m})}_{t,s|3}$. Alternatively, an Almon aggregation scheme yields the model

$$\boldsymbol{y}^{(\mathtt{m})}_{t,s|3} = \beta^{(\mathtt{m})}\sum_{\ell=0}^{3}\overline{\boldsymbol{\varphi}}(\boldsymbol{\psi}^{(\mathtt{m})}, \ell) \odot \boldsymbol{v}_{t,(4s-\ell)|12} + S^{(\mathtt{m})}\boldsymbol{w}^{(\mathtt{m})}_{t,s|3},$$

with $\beta^{(\mathtt{m})} \in \mathbb{M}_{n_{(\mathtt{m})},k}$, $S^{(\mathtt{m})} \in \mathbb{M}_{n_{(\mathtt{m})}}$, IID $n_{(\mathtt{m})}$-dimensional standard Gaussian variables $\boldsymbol{w}^{(\mathtt{m})}_{t,s|3}$ and $\overline{\boldsymbol{\varphi}}(\boldsymbol{\psi}^{(\mathtt{m})}, \ell) = \left(\overline{\varphi}(\psi^{(\mathtt{m})}{}_1, \ell), \ldots, \overline{\varphi}(\psi^{(\mathtt{m})}{}_k, \ell)\right)^\top \in \mathbb{R}^k$. The symbol $\odot$ stands for the Hadamard or componentwise matrix product.

The quarterly components can be analogously described as

$$\boldsymbol{y}^{(\mathtt{q})}_t = \beta^{(\mathtt{q})}\boldsymbol{v}_{t,0|12} + S^{(\mathtt{q})}\boldsymbol{w}^{(\mathtt{q})}_t$$

[11] We compute gradients of the marginal log-likelihood using a Kalman filter implementation for a time-inhomogeneous linear Gaussian state space model in TensorFlow Probability (Dillon et al., 2017).

[12] Because of the AR(1) dynamics, we do not write it in the companion form of the latent factor. However, unless one uses the stock aggregation scheme, one still needs to keep track of the past factor values for modeling monthly or quarterly observables.

for a stock aggregation scheme, while the Almon scheme writes as

$$\boldsymbol{y}_t^{(\mathtt{q})} = \beta^{(\mathtt{q})} \sum_{\ell=0}^{11} \overline{\boldsymbol{\varphi}}(\boldsymbol{\psi}^{(\mathtt{q})}, \ell) \odot \boldsymbol{v}_{t,-\ell|12} + S^{(\mathtt{q})} \boldsymbol{w}_t^{(\mathtt{q})},$$

with $\beta^{(\mathtt{q})} \in \mathbb{M}_{n_{(\mathtt{q})},k}$, $S^{(\mathtt{m})} \in \mathbb{M}_{n_{(\mathtt{q})}}$, IID $n_{(\mathtt{q})}$-dimensional standard Gaussian variables $\boldsymbol{w}_t^{(\mathtt{m})}$ and $\overline{\boldsymbol{\varphi}}(\boldsymbol{\psi}^{(\mathtt{q})}, \ell) = \left(\overline{\varphi}(\psi^{(\mathtt{q})}{}_1, \ell), \dots, \overline{\varphi}(\psi^{(\mathtt{q})}{}_k, \ell)\right)^\top \in \mathbb{R}^k$.

F.1 Mixed-frequency DFM Implementation

This section gives additional details on implementing non-homogeneous dynamic factor models, such as the mixed frequency model introduced in the main text. We notice that the conditioning notation in this section should not be confused with our temporal notation in Definition 2.1.

Kalman filtering and computational complexity. The sufficient statistics of the posterior distribution of the latent factor $\boldsymbol{f}_t | \boldsymbol{y}_{0:t}$ can be updated recursively by the Kalman filter updates in the linear Gaussian setting. First, propagate the prior

$$\begin{aligned} \widehat{\boldsymbol{f}}_{t+1|t,\boldsymbol{\theta}} &= A_{\boldsymbol{\theta}} \widehat{\boldsymbol{f}}_{t|t,\boldsymbol{\theta}} \\ \widehat{\Sigma}_{t+1|t,\boldsymbol{\theta}} &= A_{\boldsymbol{\theta}} \widehat{\Sigma}_{t+1|t,\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top + S_{t+1,\boldsymbol{\theta}} S_{t+1,\boldsymbol{\theta}}^\top. \end{aligned}$$

Compute the innovation covariance

$$\Gamma_{t+1,\boldsymbol{\theta}} = \Lambda_{t+1} \widehat{\Sigma}_{t+1|t,\boldsymbol{\theta}} \Lambda_{t+1}^\top + R_{\boldsymbol{\theta}} R_{\boldsymbol{\theta}}^\top$$

and the Kalman gain

$$K_{t+1,\boldsymbol{\theta}} = \widehat{\Sigma}_{t+1|t,\boldsymbol{\theta}} \Lambda_{t+1,\boldsymbol{\theta}}^\top \Gamma_{t+1,\boldsymbol{\theta}}^{-1}.$$

Then, update the statistics with the new information y_{t+1},

$$\begin{aligned} \widehat{\boldsymbol{f}}_{t+1|t+1,\boldsymbol{\theta}} &= \widehat{\boldsymbol{f}}_{t+1|t,\boldsymbol{\theta}} - K_{t+1,\boldsymbol{\theta}} \left(y_{t+1} - \Lambda_{t+1,\boldsymbol{\theta}} \widehat{\boldsymbol{f}}_{t+1|t,\boldsymbol{\theta}} \right) \\ \widehat{\Sigma}_{t+1|t+1,\boldsymbol{\theta}} &= \left(\mathbb{I} - K_{t+1,\boldsymbol{\theta}} \Lambda_{t+1,\boldsymbol{\theta}}\right) \widehat{\Sigma}_{t+1|t,\boldsymbol{\theta}}. \end{aligned}$$

Notice that the inverse of the log-determinant of the innovation matrices $\Gamma_{t,\boldsymbol{\theta}}$ are required for computing the Kalman gains and the marginal log-likelihood, respectively, which yield a cubic computational complexity in the dimension of the observation process. Alternatively, one can apply matrix inversion or determinant lemmas to obtain a computational complexity that is cubic in the dimension of the Markovian factor process $\boldsymbol{f}_t$. For an alternative approach in high-dimensions that imposes a dynamic factor structure after a projection of the observations onto a low-dimensional space, see Jungbacker and Koopman (2015), and Bräuning and Koopman (2014) for a collapsed mixed-frequency DFM.

Model selection. The model parameters $\boldsymbol{\theta}$ are learned to jointly maximize the log-likelihood of the observed data for all frequencies. This is in contrast to the parameter estimation approach for MIDAS, which minimizes the MSE of low-frequency predictions conditional on observing the high-frequency series. We remark that a different log-likelihood weighting for the different frequencies in DFMs has been suggested in Blasques et al. (2016), but requires cross-validation to optimize such weightings. Nevertheless, the introduced DFM contains several hyperparameters that need to be chosen, such as the latent state space dimension k or the order p of the latent Markov process. One possibility is to select such hyperparameters by evaluating the low-frequency predictions on a validation set.

Approaches for choosing the dimensions of the latent factor process have been under-explored in the mixed-frequency setting, but see Bai and Ng (2007), Hallin and Liška (2007) for possible criteria in general dynamic factor models. In our implementation, we choose $p = 1$, as this allows for a differentiable model parametrization with stationary factor dynamics. We set $k = 5$ for the small dataset and $k = 10$ for the medium dataset.

Parameter estimation and forecasting. Based on the results from the Kalman filtering recursions, the model parameters $\boldsymbol{\theta}$ are learned by maximizing the marginal log-likelihood using $\ell_t(\boldsymbol{\theta}) = -\log p_{\boldsymbol{\theta}}(\boldsymbol{y}_{0:t}) = -\sum_{s=0}^{t} \log p_{\boldsymbol{\theta}}(\boldsymbol{y}_s|\boldsymbol{y}_{0:s-1})$ where $p_{\boldsymbol{\theta}}(\boldsymbol{y}_s|\boldsymbol{y}_{0:s-1})$ is Gaussian with mean $\Lambda_{s,\boldsymbol{\theta}}\widehat{\boldsymbol{f}}_{s|s-1,\boldsymbol{\theta}}$ and covariance $\Gamma_{s,\boldsymbol{\theta}}$. Gradients of $\ell_t(\boldsymbol{\theta})$ can be computed using algorithmic differentiation.

For fixed $\boldsymbol{\theta} \in \Theta$ and $h \in \mathbb{N}$, let

$$\mu_{t+h|t,\boldsymbol{\theta}}(\boldsymbol{y}_{t+h}|\boldsymbol{y}_{0:t}) = \int g_{t+h,\boldsymbol{\theta}}(\boldsymbol{y}_{t+h}|\boldsymbol{f}_{t+h}) \prod_{\ell=1}^{h} h_{t+\ell,\boldsymbol{\theta}}(\boldsymbol{f}_{t+\ell}|\boldsymbol{f}_{t+\ell-1})\mathrm{d}\boldsymbol{f}_{t+\ell}\pi_{t|t,\boldsymbol{\theta}}(\boldsymbol{f}_t|\boldsymbol{y}_{0:t})\mathrm{d}\boldsymbol{f}_t$$

be the h-step predictive distribution of the data, while $\pi_{t|t,\boldsymbol{\theta}}(\boldsymbol{f}_t|\boldsymbol{y}_{0:t})$ is the filtering distribution of the latent state $\boldsymbol{f}_t|\boldsymbol{y}_{0:t}$. The mean of $\mu_{t+h|t,\boldsymbol{\theta}}(\cdot|\boldsymbol{y}_{0:t})$ is $\widehat{\boldsymbol{y}}_{t+h|t,\boldsymbol{\theta}} = \mathbb{E}_{\boldsymbol{\theta}}\left[\boldsymbol{y}_{t+h}|\boldsymbol{y}_{0:t}\right]$. For some $t, \tau \geq 0$, let us write $\widehat{\boldsymbol{f}}_{t+\tau|t,\boldsymbol{\theta}} = \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{f}_{t+\tau}|\boldsymbol{y}_{0:t}]$ and $\Sigma_{t+\tau|t,\boldsymbol{\theta}} = \mathrm{Cov}_{\boldsymbol{\theta}}[\boldsymbol{f}_{t+\tau} - \widehat{\boldsymbol{f}}_{t+\tau|t,\boldsymbol{\theta}}|\boldsymbol{y}_{0:t}]$ for the mean and covariance of the latent process, respectively. For linear Gaussian dynamics, Kalman filtering allows for computing the filtered mean $\widehat{\boldsymbol{f}}_{t|t,\boldsymbol{\theta}}$ and covariance matrices $\widehat{\Sigma}_{t|t,\boldsymbol{\theta}}$ analytically.

For fixed $\boldsymbol{\theta}$, the τ-step ahead prediction function $H^{\tau}_{t,\boldsymbol{\theta}}(\boldsymbol{y}_{0:t}) = \widehat{\boldsymbol{y}}_{t+\tau|t,\boldsymbol{\theta}} = \Lambda_{t+\tau,\boldsymbol{\theta}}\widehat{\boldsymbol{f}}_{t+\tau|t,\boldsymbol{\theta}}$ is linear due to the Kalman filtering recursion. For $s \leq t$, consider also the prediction $H^{\star\tau}_{s,t}(\boldsymbol{y}_{0:t}) = \mathbb{E}_{\boldsymbol{\theta}^\star(\boldsymbol{y}_{0:s})}[\boldsymbol{y}_{t+\tau}|\boldsymbol{y}_{0:t}]$ that is based on the sample $\boldsymbol{y}_{0:t}$, but where $\boldsymbol{\theta}^\star(\boldsymbol{y}_{0:s}) = \arg\min_{\boldsymbol{\theta}} \ell_s(\boldsymbol{\theta})$ maximizes the marginal likelihood of data $\boldsymbol{y}_{0:s}$ only. This setting allows to implement different parameter estimation setups from Section B.1. For instance, the fixed parameter setup corresponds to fixing s, which yields a fixed training set $\boldsymbol{y}_{0:s}$ to estimate $\boldsymbol{\theta}$. In the expanding window setup, both s and t are expanded, while a rolling window setting updates the dataset $\boldsymbol{y}_{0:s}$ by rolling over the data.

G High-Frequency Forecasts

To better understand how the use of high-frequency data impacts forecasting, as an additional empirical experiment we investigate high-frequency (HF) forecasts of all models in the Small-MD dataset. We restrict our analysis to this dataset because the computational burden to construct HF forecasts can be high: when using daily data and using our suggested 24 days-per-month interpolation, one quarter amounts to 72 daily frequency observations, which means HF forecasts can involve thousands of data points, and for DFM and M-MFESN models this setup can be quite computationally onerous.

Constructing HF forecasts with MIDAS is trivial once the aggregation weights have been estimated, even though a practical implementation requires care in constructing the appropriate lag matrices. Recall for Section E that the MIDAS equation with L regressors $(z^{(l)}_{t,s|\kappa_l})_{t,s}$ with $z^{(l)}_{t,s|\kappa_l} \in \mathbb{R}$, $l \in [L]$ for all $t \in \mathbb{Z}$ and $s \in \{0, \ldots, \kappa_l - 1\}$ can be written as

$$y_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{l=1}^{L} \beta_l \sum_{k=0}^{K_l} \varphi(\boldsymbol{\theta}_l, k) z^{(l)}_{t,-k|\kappa_l} + \epsilon_t.$$

For clarity, we suppress the dynamic autoregressive component, as it has the same frequency as the target. Now assume that we include $n_{(\mathtt{m})}$ monthly and $n_{(\mathtt{d})}$ daily frequency regressors in the model that are sampled $\kappa_{(\mathtt{m})} = 3$ and $\kappa_{(\mathtt{d})} = 72$ times per quarter and hence $\kappa_{\max} = 72$. Therefore we can partition the regression above in the following way

$$y_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{l=1}^{n_{(\mathtt{m})}} \beta_l \sum_{k=0}^{K_l} \varphi(\boldsymbol{\theta}_l, k) z^{(l)}_{t,-k|3} + \sum_{l=n_{(\mathtt{m})}+1}^{L} \beta_l \sum_{k=0}^{K_l} \varphi(\boldsymbol{\theta}_l, k) z^{(l)}_{t,-k|72} + \epsilon_t$$

with $L = n_{(\mathtt{m})} + n_{(\mathtt{d})}$.

Assuming parameter estimates $\widehat{\alpha}_0, \widehat{\alpha}_1, \ldots, \widehat{\alpha}_p$ and $\{(\widehat{\beta}_l, \widehat{\boldsymbol{\theta}}_l)\}_{l=1}^{L}$ are available, the HF forecast $\widehat{y}_{t+1,s|72}$ is given by

$$\widehat{y}_{t+1,s|72} = \widehat{\alpha}_0 + \sum_{i=1}^{p} \widehat{\alpha}_i y_{t-i} + \sum_{l=1}^{n_{(\mathtt{m})}} \widehat{\beta}_l \sum_{k=0}^{K_l} \varphi(\widehat{\boldsymbol{\theta}}_l, k) z^{(l)}_{t,\lfloor s/24 \rfloor - k|3} + \sum_{l=n_{(\mathtt{m})}+1}^{L} \widehat{\beta}_l \sum_{k=0}^{K_l} \varphi(\widehat{\boldsymbol{\theta}}_l, k) z^{(l)}_{t,s-k|72}.$$

For DFMs, high-frequency forecasts can be constructed using (F.6) and (F.7) to iterate factors forward in time and then aggregate them according to estimated loadings or a weighting scheme.

Multi-frequency ESN models are also able to yield high-frequency forecasts in a straightforward manner. For simplicity, let us consider the case as in Example 4.1 of an aligned S-MFESN model that has been fit to a quarterly target with monthly and daily input data. The reservoir is run in high-frequency, $\kappa_{\max}$ steps per quarter, according to state equation

$$\boldsymbol{x}^{(\mathtt{m},\mathtt{d})}_{t,s|72} = \alpha \boldsymbol{x}^{(\mathtt{m},\mathtt{d})}_{t,s-1|72} + (1-\alpha)\sigma(A\boldsymbol{x}^{(\mathtt{m},\mathtt{d})}_{t,s-1|72} + C\boldsymbol{z}^{(\mathtt{m},\mathtt{d})}_{t,s|72} + \boldsymbol{\zeta}).$$

Suppose a coefficient matrix $\widehat{W}$ has been estimated. Then, as states between low-frequency periods t and $t+1$ are collected, we can immediately construct the high-frequency one-step-ahead forecasts

$$\widetilde{y}_{t+1,s|72} = \widehat{W}^{\top} \boldsymbol{x}^{(\mathtt{m},\mathtt{d})}_{t,s|72}.$$

For M-MFESN models HF forecasts require slightly more care. For example, when dealing with the multi-reservoir MFESN model of Example 4.2, we must repeat the most recent monthly state at daily frequency correctly.

H Robustness Analysis

H.1 MIDAS

As we discuss briefly in the main text, parameter optimization is a principal problem in implementing any MIDAS model. Even though explicit formulas exist for both gradient and Hessian of the MIDAS loss objective when an Almon weighting scheme is used (see Kostrov (2021)), there is no known guarantee that the loss itself is convex or even locally convex. In practice, for a given starting point (or point set) a numerical optimizer might only converge to a local minimum.

We observe this in practice, and we explore its effects on the robustness of MIDAS forecasts. We report summary results for our simulations in Figure 20. Our proposal is, given a MIDAS model specification and a set of starting points for evaluating the loss, to run an optimizer (for example, L-BFGS-B with explicit gradient) and select the smallest local minimum. By repeating this procedure multiple times, we collect a set of MIDAS parameters and study both the variation between the parameter vectors and the implied one-step ahead forecasts.

To be precise, our procedure is as follows:

1. For a total of B repetitions:

   (a) Choose M initialization points for the optimizer. We draw 64 points inside the hypercube of edge length 0.025 using a low-discrepancy Sobol sequence. The choice of a down-scaled hypercube as a domain comes from the empirical fact that the Almon exponential scheme may produce extremely large values even for relatively small coefficients. A straightforward way to see this is to notice that given any arbitrary small value for θ_1 and θ_2 in (E.2), for lag index k sufficiently large weight $\exp(\theta_1 k + \theta_2 k^2)$ will overflow at any given numerical precision. This means that one should adjust the MIDAS optimization domain based on the number of lags in the model.

(b) For each initialization point, run the optimizer of choice.

(c) Among the resulting M (local) loss minimizers, select and store the one with the lowest loss value.

2. With the resulting B stored minimizer:

   - Construct a low-dimensional projection of the high-dimensional minima to see their relative location in the parameter space and to compare their gradient and loss values, see Figure 20 (a)-(b).
   - Use each minimizer to produce MIDAS one-step ahead forecasts and plot quantile frequency plots of the forecast variation due to initialization; see Figure 20 (c).

Figure 20 shows that the best minimizers among initial Sobol sets are clustered together. To construct this 2D projection of the high-dimensional Almon coefficient space (including autoregressive lags and intercept), we use the well-known t-SNE procedure developed in van der Maaten (2009), which is an unsupervised dimensionality reduction algorithm capable of preserving the latent high-dimensional structure. This approach naturally implies that the Euclidean distances in the plot are suggestive of “clustering” rather than the actual latent distance between points. In Figures 20 (a)-(b), we see that the L-BFGS-B optimizer with explicit gradient achieves good convergence in terms of gradient norm and also that the resulting cluster of minimizers has close loss values. However, one can see that there is no single loss minimum: Figure 20 instead suggests that the local structure of the MIDAS loss function is very uneven, and therefore many distinct local minima can be achieved even when choosing a large number of initialization values for the optimizer. This means that the “multi-start” strategy suggested in Kostrov (2021) to alleviate issues in MIDAS model estimation is insufficient.

The effects of non-negligible variation in parameter values on forecasts appear to be significant. Looking at Figure 20 (c), we can see wide frequency bands for the one-step ahead forecasts constructed using the Small-MD dataset and fixed parameter values. In particular, the Financial Crisis period seems to induce larger deviations in forecasts, consistent with the intuition that data with larger variation causes stronger model sensitivity when making forecasts.

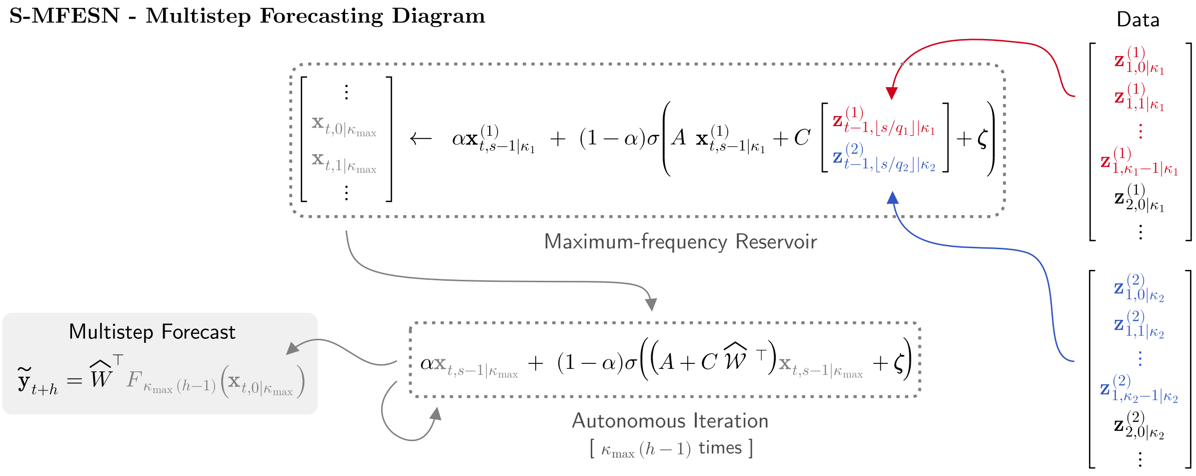
H.2 MFESN

Since ESN models, and thus MFESN models, require random sampling of parameter matrices, the size of which is often large, there is inherent variability in any ESN model forecast. In theory, because all MFESN state parameters $(\widetilde{A}, \widetilde{C}, \widetilde{\zeta})$ are drawn independently of each other, one could try to decompose the variance of any MFESN into the share due to parameter sampling and the share due to data sampling. Unfortunately, in practice, such decomposition is hard to derive. Cross-validation of ridge penalties and rolling and expanding window estimation are non-trivial data-dependent operations that complicate inference. In this work, we limit ourselves to numerically evaluating the effect of reservoir coefficient sampling on MFESN forecast variance.

Our approach is straightforward: given an MFESN model specification, c.f. Table 5.2, and a forecasting setup (fixed parameters, expanding or rolling window), we resample the reservoir state matrix parameters, perform cross-validation and possibly train-test sample windowing, and finally construct pointwise forecasts. Once a sufficiently large set of resampling forecasts has been computed, we plot frequency intervals in Figures 22 and 24. From Figure 22, we can see that the single-reservoir MFESN model with reservoir size 30 produces forecasts with a meaningful amount of variability induced by parameter resampling. Forecasts exhibit more variation when using an expanding or rolling window estimation strategy, even though the overall forecasts align with the GDP realizations. A similar discussion to that of MIDAS applies here: forecast sensitivity increases with underlying data variation, exacerbated in periods of systemic economic crisis.

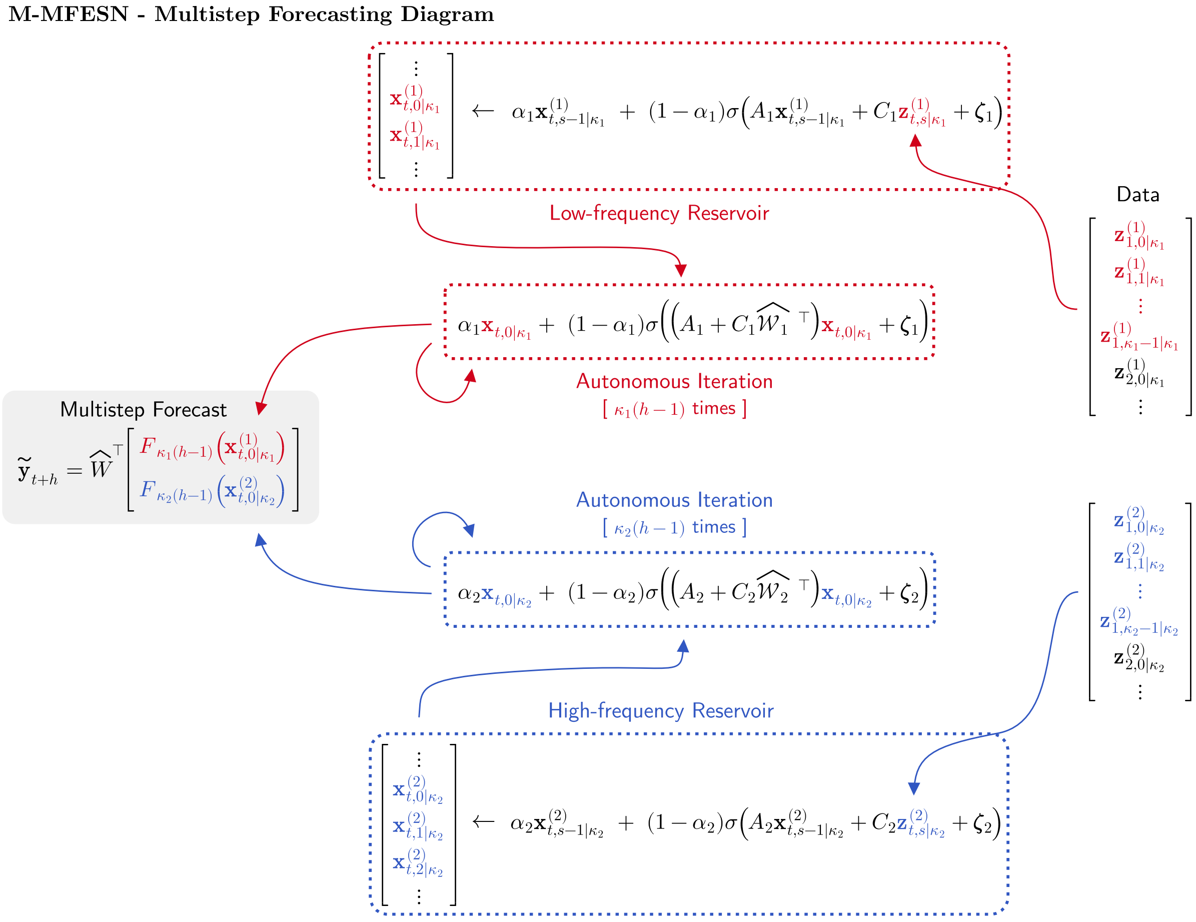
Figure 24 suggests that larger MFESN models produce significantly more stable forecasts regarding model resampling. Note that the M-MFESN model [A] has a monthly frequency reservoir that is approximately 3 times the size of the S-MFESN model [A]. This stability is preserved even in expanding or rolling window settings, even though a slightly higher variation is apparent at the height of the 2008 Financial Crisis. We hypothesize that this reduction in variance due to model parameter sampling is due to the concentration of measure phenomena that prevail in high-dimensional spaces.
Figure 24 suggests that larger MFESN models produce significantly more stable forecasts regarding model resampling. Note that the M-MFESN model [A] has a monthly frequency reservoir that is approximately 3 times the size of the S-MFESN model [A]. This stability is preserved even in expanding or rolling window settings, even though a slightly higher variation is apparent at the height of the 2008 Financial Crisis. We hypothesize that this reduction in variance due to model parameter sampling is due to the concentration of measure phenomena that prevail in high-dimensional spaces.

I Additional Figures


S-MFESN - Multistep Forecasting Diagram

Data
$\mathbf{z}^{(1)}_{1,0|\kappa_1}$ $\mathbf{z}^{(1)}_{1,1|\kappa_1}$ $\vdots$ $\mathbf{z}^{(1)}_{1,\kappa_1-1|\kappa_1}$ $\mathbf{z}^{(1)}_{2,0|\kappa_1}$ $\vdots$
$\mathbf{z}^{(2)}_{1,0|\kappa_2}$ $\mathbf{z}^{(2)}_{1,1|\kappa_2}$ $\vdots$ $\mathbf{z}^{(2)}_{1,\kappa_2-1|\kappa_2}$ $\mathbf{z}^{(2)}_{2,0|\kappa_2}$ $\vdots$
$\begin{bmatrix} \vdots \\ \mathbf{x}_{t,0|\kappa_{\max}} \\ \mathbf{x}_{t,1|\kappa_{\max}} \\ \vdots \end{bmatrix} \leftarrow \alpha \mathbf{x}^{(1)}_{t,s-1|\kappa_1} + (1-\alpha)\sigma\left(A\ \mathbf{x}^{(1)}_{t,s-1|\kappa_1} + C \begin{bmatrix} \mathbf{z}^{(1)}_{t-1,\lfloor s/q_1 \rfloor|\kappa_1} \\ \mathbf{z}^{(2)}_{t-1,\lfloor s/q_2 \rfloor|\kappa_2} \end{bmatrix} + \zeta\right)$
Maximum-frequency Reservoir
$\alpha \mathbf{x}_{t,s-1|\kappa_{\max}} + (1-\alpha)\sigma\left(\left(A + C\,\widehat{W}^\top\right)\mathbf{x}_{t,s-1|\kappa_{\max}} + \zeta\right)$
Autonomous Iteration
[$\kappa_{\max}(h-1)$ times]
Multistep Forecast
$\widetilde{\mathbf{y}}_{t+h} = \widehat{W}^\top F_{\kappa_{\max}(h-1)}\left(\mathbf{x}_{t,0|\kappa_{\max}}\right)$

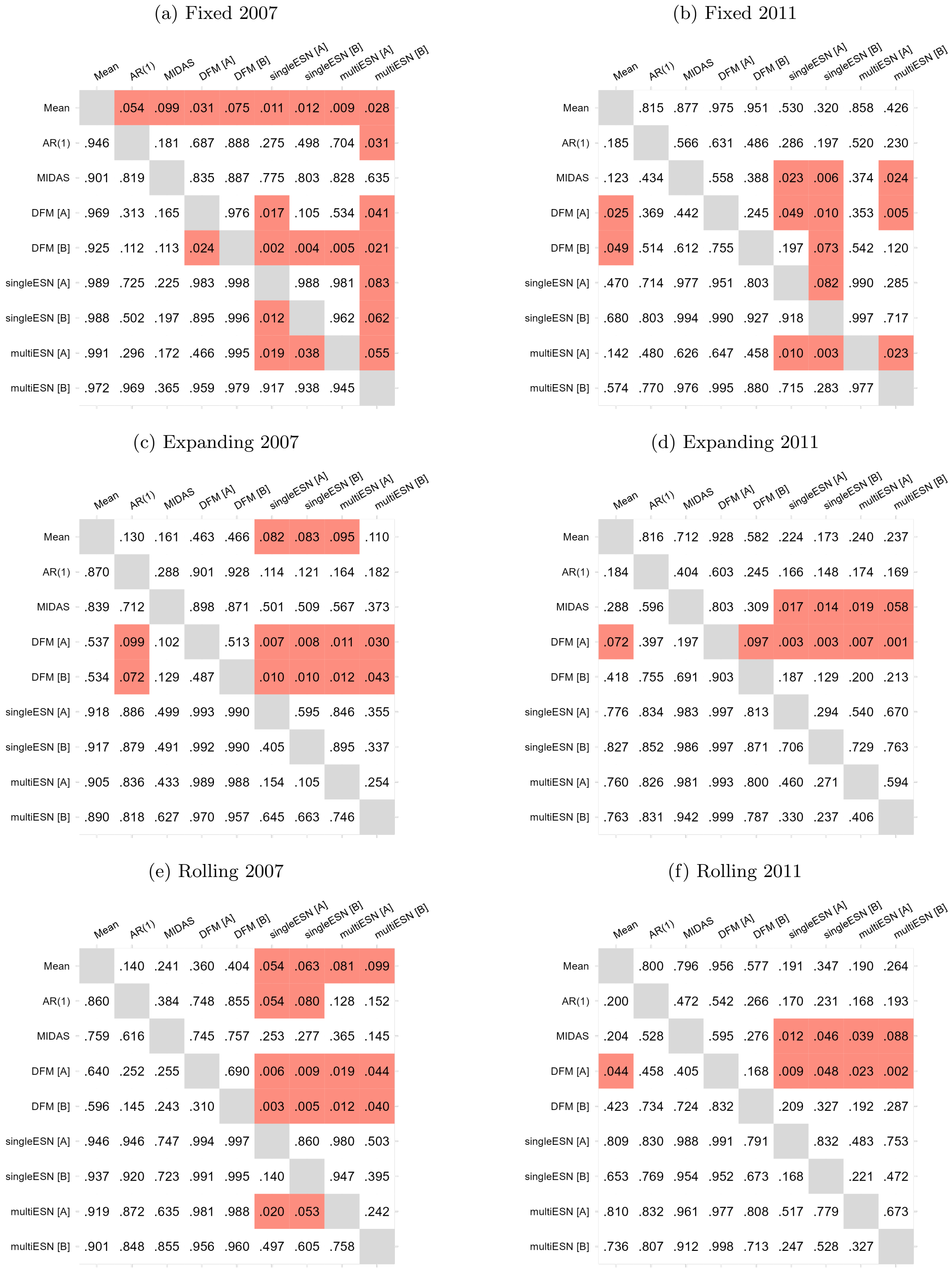
Figure 12


M-MFESN - Multistep Forecasting Diagram

$\begin{bmatrix} \vdots \\ \mathbf{x}^{(1)}_{t,0|\kappa_1} \\ \mathbf{x}^{(1)}_{t,1|\kappa_1} \\ \vdots \end{bmatrix} \leftarrow \alpha_1 \mathbf{x}^{(1)}_{t,s-1|\kappa_1} + (1-\alpha_1)\sigma\left(A_1 \mathbf{x}^{(1)}_{t,s-1|\kappa_1} + C_1 \mathbf{z}^{(1)}_{t,s|\kappa_1} + \zeta_1\right)$
Low-frequency Reservoir
Data
$\mathbf{z}^{(1)}_{1,0|\kappa_1}$ $\mathbf{z}^{(1)}_{1,1|\kappa_1}$ $\vdots$ $\mathbf{z}^{(1)}_{1,\kappa_1-1|\kappa_1}$ $\mathbf{z}^{(1)}_{2,0|\kappa_1}$ $\vdots$
$\alpha_1 \mathbf{x}_{t,0|\kappa_1} + (1-\alpha_1)\sigma\left(\left(A_1 + C_1 \widehat{W}_1^\top\right)\mathbf{x}_{t,0|\kappa_1} + \zeta_1\right)$
Autonomous Iteration
[$\kappa_1(h-1)$ times]
Multistep Forecast
$\widetilde{\mathbf{y}}_{t+h} = \widehat{W}^\top \begin{bmatrix} F_{\kappa_1(h-1)}\left(\mathbf{x}^{(1)}_{t,0|\kappa_1}\right) \\ F_{\kappa_2(h-1)}\left(\mathbf{x}^{(2)}_{t,0|\kappa_2}\right) \end{bmatrix}$
Autonomous Iteration
[$\kappa_2(h-1)$ times]
$\alpha_2 \mathbf{x}_{t,0|\kappa_2} + (1-\alpha_2)\sigma\left(\left(A_2 + C_2 \widehat{W}_2^\top\right)\mathbf{x}_{t,0|\kappa_2} + \zeta_2\right)$
$\mathbf{z}^{(2)}_{1,0|\kappa_2}$ $\mathbf{z}^{(2)}_{1,1|\kappa_2}$ $\vdots$ $\mathbf{z}^{(2)}_{1,\kappa_2-1|\kappa_2}$ $\mathbf{z}^{(2)}_{2,0|\kappa_2}$ $\vdots$
High-frequency Reservoir
$\begin{bmatrix} \vdots \\ \mathbf{x}^{(2)}_{t,0|\kappa_2} \\ \mathbf{x}^{(2)}_{t,1|\kappa_2} \\ \mathbf{x}^{(2)}_{t,2|\kappa_2} \\ \vdots \end{bmatrix} \leftarrow \alpha_2 \mathbf{x}^{(2)}_{t,s-1|\kappa_2} + (1-\alpha_2)\sigma\left(A_2 \mathbf{x}^{(2)}_{t,s-1|\kappa_2} + C_2 \mathbf{z}^{(2)}_{t,s|\kappa_2} + \zeta_2\right)$

Figure 13

Figure 14: 1-Step-ahead GDP Forecasting – Modified Diebold-Mariano – Small-MD Dataset



(a) Fixed 2007

	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.054	.099	.031	.075	.011	.012	.009	.028
AR(1)	.946		.181	.687	.888	.275	.498	.704	.031
MIDAS	.901	.819		.835	.887	.775	.803	.828	.635
DFM [A]	.969	.313	.165		.976	.017	.105	.534	.041
DFM [B]	.925	.112	.113	.024		.002	.004	.005	.021
singleESN [A]	.989	.725	.225	.983	.998		.988	.981	.083
singleESN [B]	.988	.502	.197	.895	.996	.012		.962	.062
multiESN [A]	.991	.296	.172	.466	.995	.019	.038		.055
multiESN [B]	.972	.969	.365	.959	.979	.917	.938	.945	

(b) Fixed 2011

	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.815	.877	.975	.951	.530	.320	.858	.426
AR(1)	.185		.566	.631	.486	.286	.197	.520	.230
MIDAS	.123	.434		.558	.388	.023	.006	.374	.024
DFM [A]	.025	.369	.442		.245	.049	.010	.353	.005
DFM [B]	.049	.514	.612	.755		.197	.073	.542	.120
singleESN [A]	.470	.714	.977	.951	.803		.082	.990	.285
singleESN [B]	.680	.803	.994	.990	.927	.918		.997	.717
multiESN [A]	.142	.480	.626	.647	.458	.010	.003		.023
multiESN [B]	.574	.770	.976	.995	.880	.715	.283	.977	

(c) Expanding 2007

	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.130	.161	.463	.466	.082	.083	.095	.110
AR(1)	.870		.288	.901	.928	.114	.121	.164	.182
MIDAS	.839	.712		.898	.871	.501	.509	.567	.373
DFM [A]	.537	.099	.102		.513	.007	.008	.011	.030
DFM [B]	.534	.072	.129	.487		.010	.010	.012	.043
singleESN [A]	.918	.886	.499	.993	.990		.595	.846	.355
singleESN [B]	.917	.879	.491	.992	.990	.405		.895	.337
multiESN [A]	.905	.836	.433	.989	.988	.154	.105		.254
multiESN [B]	.890	.818	.627	.970	.957	.645	.663	.746	

(d) Expanding 2011

	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.816	.712	.928	.582	.224	.173	.240	.237
AR(1)	.184		.404	.603	.245	.166	.148	.174	.169
MIDAS	.288	.596		.803	.309	.017	.014	.019	.058
DFM [A]	.072	.397	.197		.097	.003	.003	.007	.001
DFM [B]	.418	.755	.691	.903		.187	.129	.200	.213
singleESN [A]	.776	.834	.983	.997	.813		.294	.540	.670
singleESN [B]	.827	.852	.986	.997	.871	.706		.729	.763
multiESN [A]	.760	.826	.981	.993	.800	.460	.271		.594
multiESN [B]	.763	.831	.942	.999	.787	.330	.237	.406	

(e) Rolling 2007

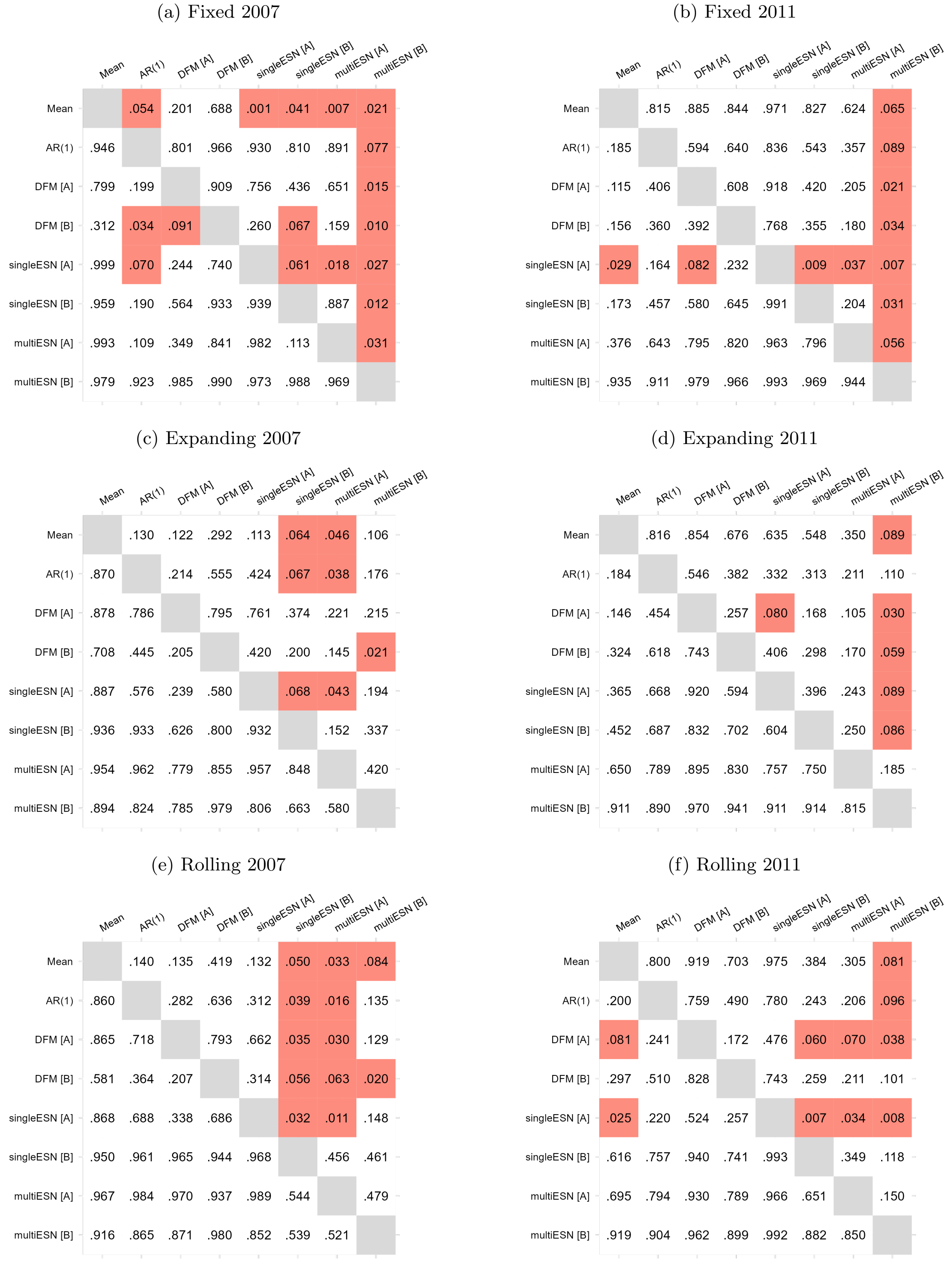
	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.140	.241	.360	.404	.054	.063	.081	.099
AR(1)	.860		.384	.748	.855	.054	.080	.128	.152
MIDAS	.759	.616		.745	.757	.253	.277	.365	.145
DFM [A]	.640	.252	.255		.690	.006	.009	.019	.044
DFM [B]	.596	.145	.243	.310		.003	.005	.012	.040
singleESN [A]	.946	.946	.747	.994	.997		.860	.980	.503
singleESN [B]	.937	.920	.723	.991	.995	.140		.947	.395
multiESN [A]	.919	.872	.635	.981	.988	.020	.053		.242
multiESN [B]	.901	.848	.855	.956	.960	.497	.605	.758	

(f) Rolling 2011

	Mean	AR(1)	MIDAS	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.800	.796	.956	.577	.191	.347	.190	.264
AR(1)	.200		.472	.542	.266	.170	.231	.168	.193
MIDAS	.204	.528		.595	.276	.012	.046	.039	.088
DFM [A]	.044	.458	.405		.168	.009	.048	.023	.002
DFM [B]	.423	.734	.724	.832		.209	.327	.192	.287
singleESN [A]	.809	.830	.988	.991	.791		.832	.483	.753
singleESN [B]	.653	.769	.954	.952	.673	.168		.221	.472
multiESN [A]	.810	.832	.961	.977	.808	.517	.779		.673
multiESN [B]	.736	.807	.912	.998	.713	.247	.528	.327	

Figure 15: p-values of the pairwise Modified Diebold-Mariano tests between models of Table 5.4. Tests are one-sided and carried out row-wise: the null hypothesis for row i and column j reads as "the ith-row model forecasts *more accurately* than the jth-column model". Rejections at the 10% level are highlighted in red.

Figure 16: 1-Step-ahead GDP Forecasting – Modified Diebold-Mariano – Medium-MD Dataset



(a) Fixed 2007

	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.054	.201	.688	.001	.041	.007	.021
AR(1)	.946		.801	.966	.930	.810	.891	.077
DFM [A]	.799	.199		.909	.756	.436	.651	.015
DFM [B]	.312	.034	.091		.260	.067	.159	.010
singleESN [A]	.999	.070	.244	.740		.061	.018	.027
singleESN [B]	.959	.190	.564	.933	.939		.887	.012
multiESN [A]	.993	.109	.349	.841	.982	.113		.031
multiESN [B]	.979	.923	.985	.990	.973	.988	.969	

(b) Fixed 2011

	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.815	.885	.844	.971	.827	.624	.065
AR(1)	.185		.594	.640	.836	.543	.357	.089
DFM [A]	.115	.406		.608	.918	.420	.205	.021
DFM [B]	.156	.360	.392		.768	.355	.180	.034
singleESN [A]	.029	.164	.082	.232		.009	.037	.007
singleESN [B]	.173	.457	.580	.645	.991		.204	.031
multiESN [A]	.376	.643	.795	.820	.963	.796		.056
multiESN [B]	.935	.911	.979	.966	.993	.969	.944	

(c) Expanding 2007

	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.130	.122	.292	.113	.064	.046	.106
AR(1)	.870		.214	.555	.424	.067	.038	.176
DFM [A]	.878	.786		.795	.761	.374	.221	.215
DFM [B]	.708	.445	.205		.420	.200	.145	.021
singleESN [A]	.887	.576	.239	.580		.068	.043	.194
singleESN [B]	.936	.933	.626	.800	.932		.152	.337
multiESN [A]	.954	.962	.779	.855	.957	.848		.420
multiESN [B]	.894	.824	.785	.979	.806	.663	.580	

(d) Expanding 2011

	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.816	.854	.676	.635	.548	.350	.089
AR(1)	.184		.546	.382	.332	.313	.211	.110
DFM [A]	.146	.454		.257	.080	.168	.105	.030
DFM [B]	.324	.618	.743		.406	.298	.170	.059
singleESN [A]	.365	.668	.920	.594		.396	.243	.089
singleESN [B]	.452	.687	.832	.702	.604		.250	.086
multiESN [A]	.650	.789	.895	.830	.757	.750		.185
multiESN [B]	.911	.890	.970	.941	.911	.914	.815	

(e) Rolling 2007

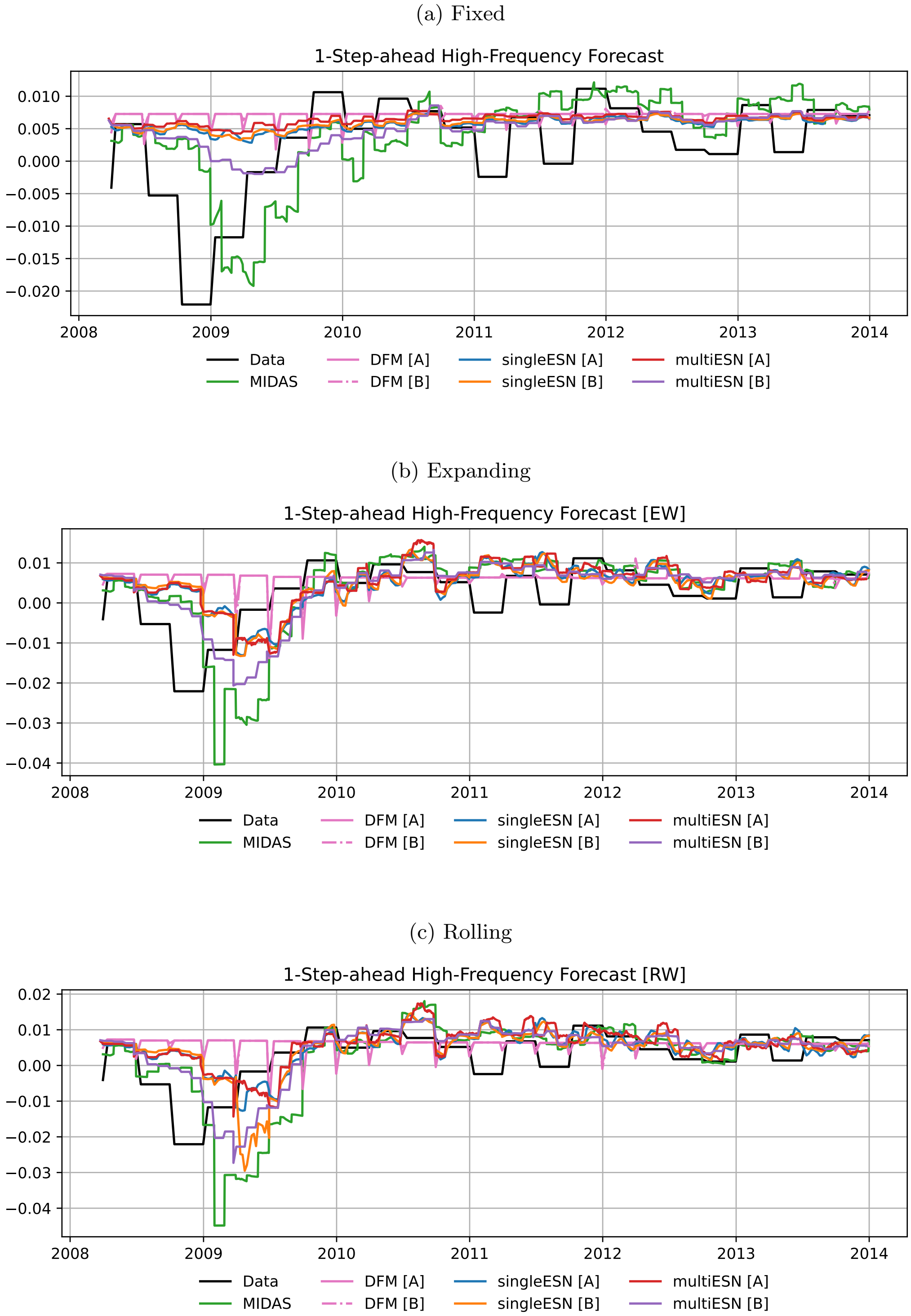
	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.140	.135	.419	.132	.050	.033	.084
AR(1)	.860		.282	.636	.312	.039	.016	.135
DFM [A]	.865	.718		.793	.662	.035	.030	.129
DFM [B]	.581	.364	.207		.314	.056	.063	.020
singleESN [A]	.868	.688	.338	.686		.032	.011	.148
singleESN [B]	.950	.961	.965	.944	.968		.456	.461
multiESN [A]	.967	.984	.970	.937	.989	.544		.479
multiESN [B]	.916	.865	.871	.980	.852	.539	.521	

(f) Rolling 2011

	Mean	AR(1)	DFM [A]	DFM [B]	singleESN [A]	singleESN [B]	multiESN [A]	multiESN [B]
Mean		.800	.919	.703	.975	.384	.305	.081
AR(1)	.200		.759	.490	.780	.243	.206	.096
DFM [A]	.081	.241		.172	.476	.060	.070	.038
DFM [B]	.297	.510	.828		.743	.259	.211	.101
singleESN [A]	.025	.220	.524	.257		.007	.034	.008
singleESN [B]	.616	.757	.940	.741	.993		.349	.118
multiESN [A]	.695	.794	.930	.789	.966	.651		.150
multiESN [B]	.919	.904	.962	.899	.992	.882	.850	

Figure 17: p-values of pairwise Modified Diebold-Mariano tests between models of Table 5.4. Tests are one-sided and carried out row-wise: the null hypothesis for row i and column j reads as "the ith-row model forecasts *more accurately* than the jth-column model". Rejections at the 10% level are highlighted in red.

Figure 18: 1-Step-ahead High-Frequency GDP Forecasting – 2007 Sample – Small-MD Dataset



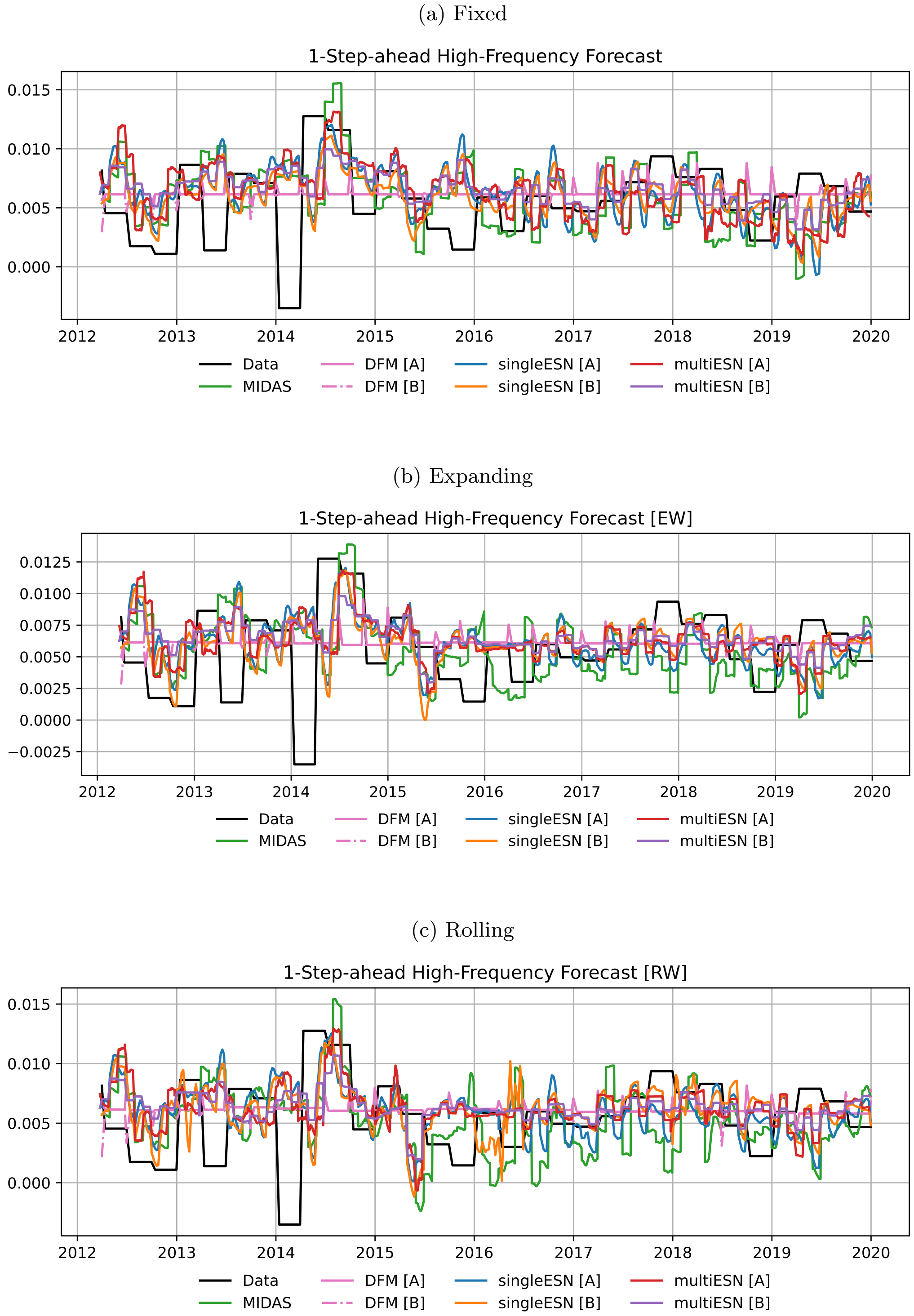
(a) Fixed

(b) Expanding

(c) Rolling

Note: Forecasts for the 2007 sample are presented up to Q4 2013 to better display the high-frequency behavior of models during the Financial Crisis period.

Figure 19: 1-Step-ahead High-Frequency GDP Forecasting – 2011 Sample – Small-MD Dataset

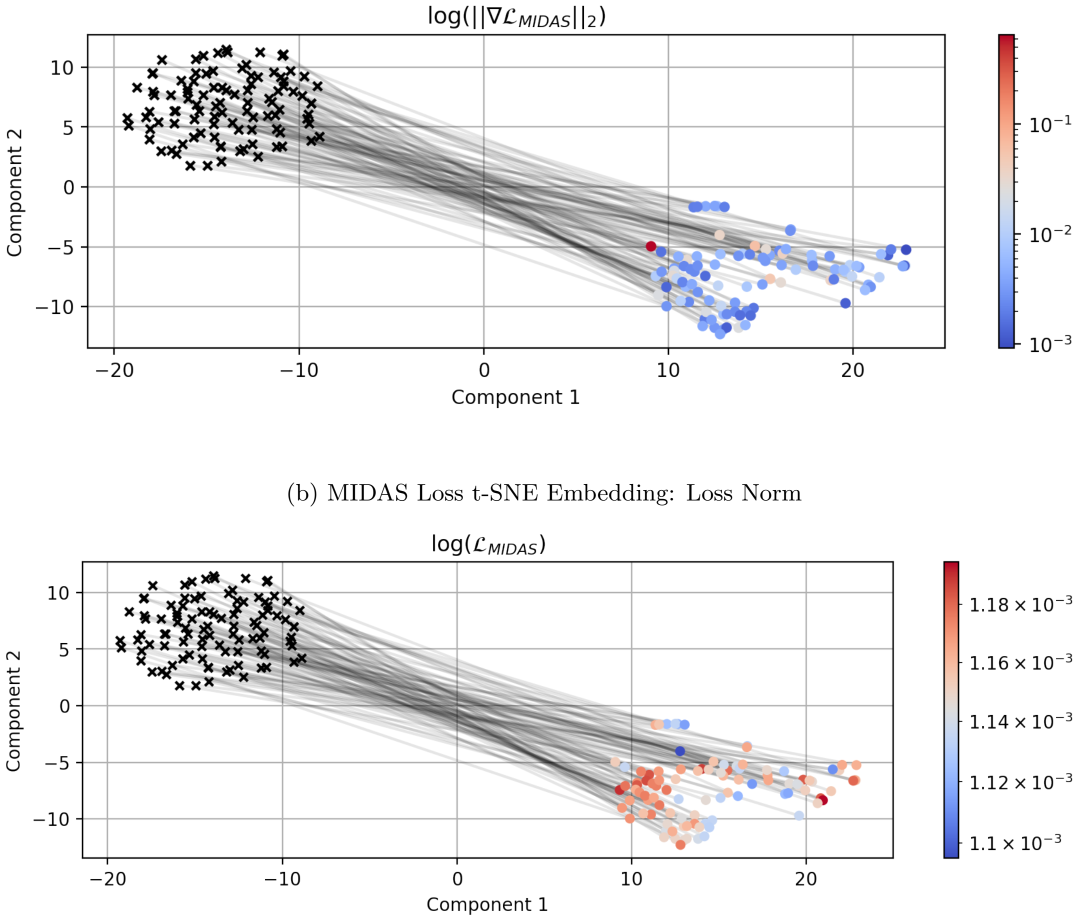


(a) Fixed

(b) Expanding

(c) Rolling

Figure 20: MIDAS Robustness Plots – 2007 Sample – Small-MD Dataset

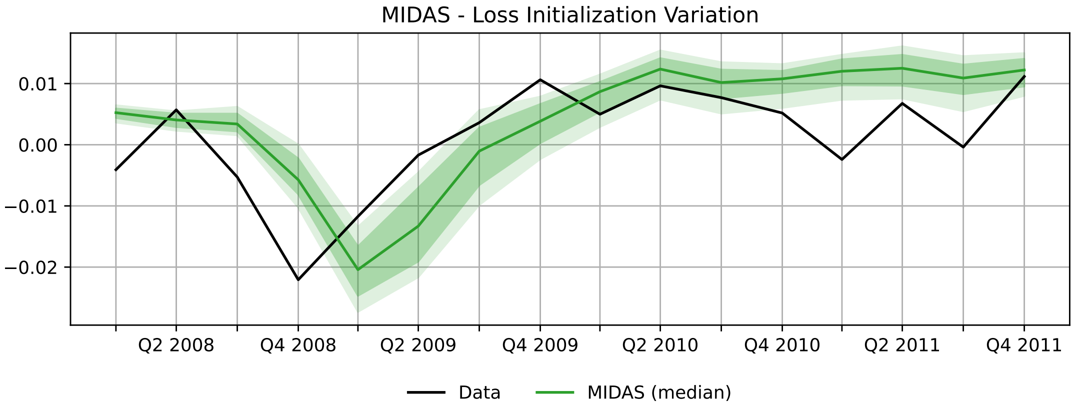


(a) MIDAS Loss t-SNE Embedding: Gradient Norm

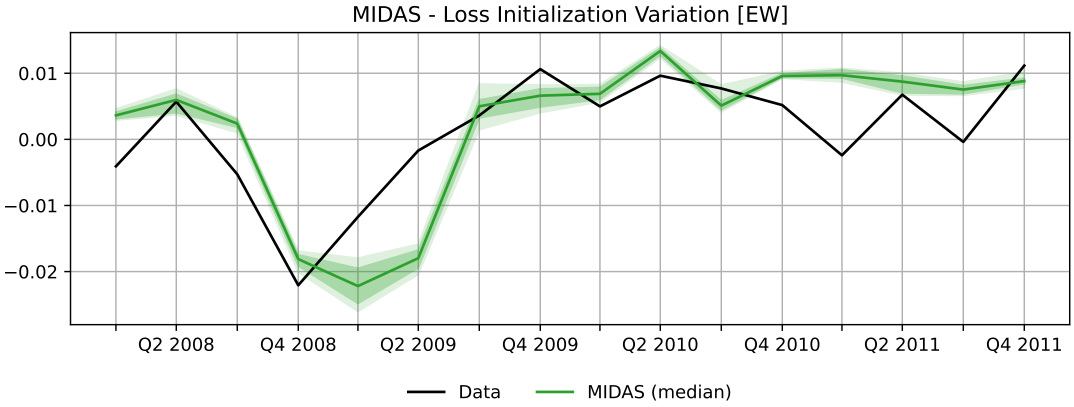
(b) MIDAS Loss t-SNE Embedding: Loss Norm

Figure 21: MIDAS Robustness Plots – 2007 Sample – Small-MD Dataset

(a) Fixed Parameters



(b) Expanding



(c) Rolling

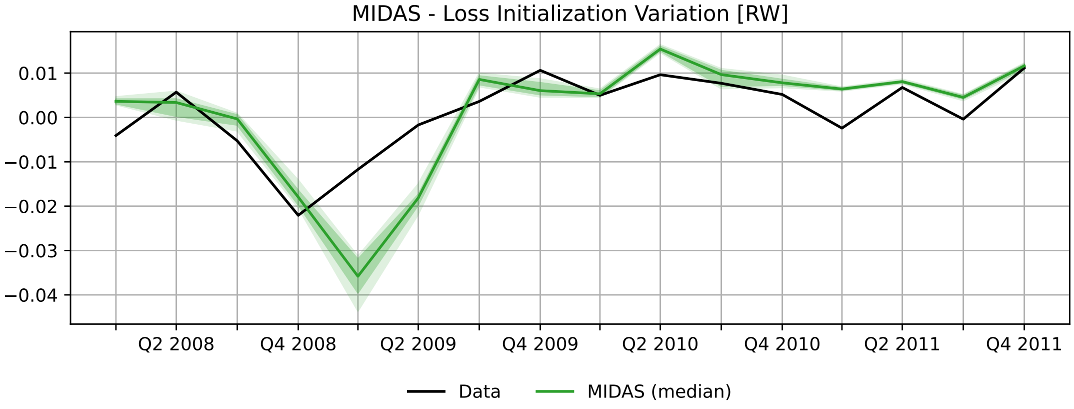
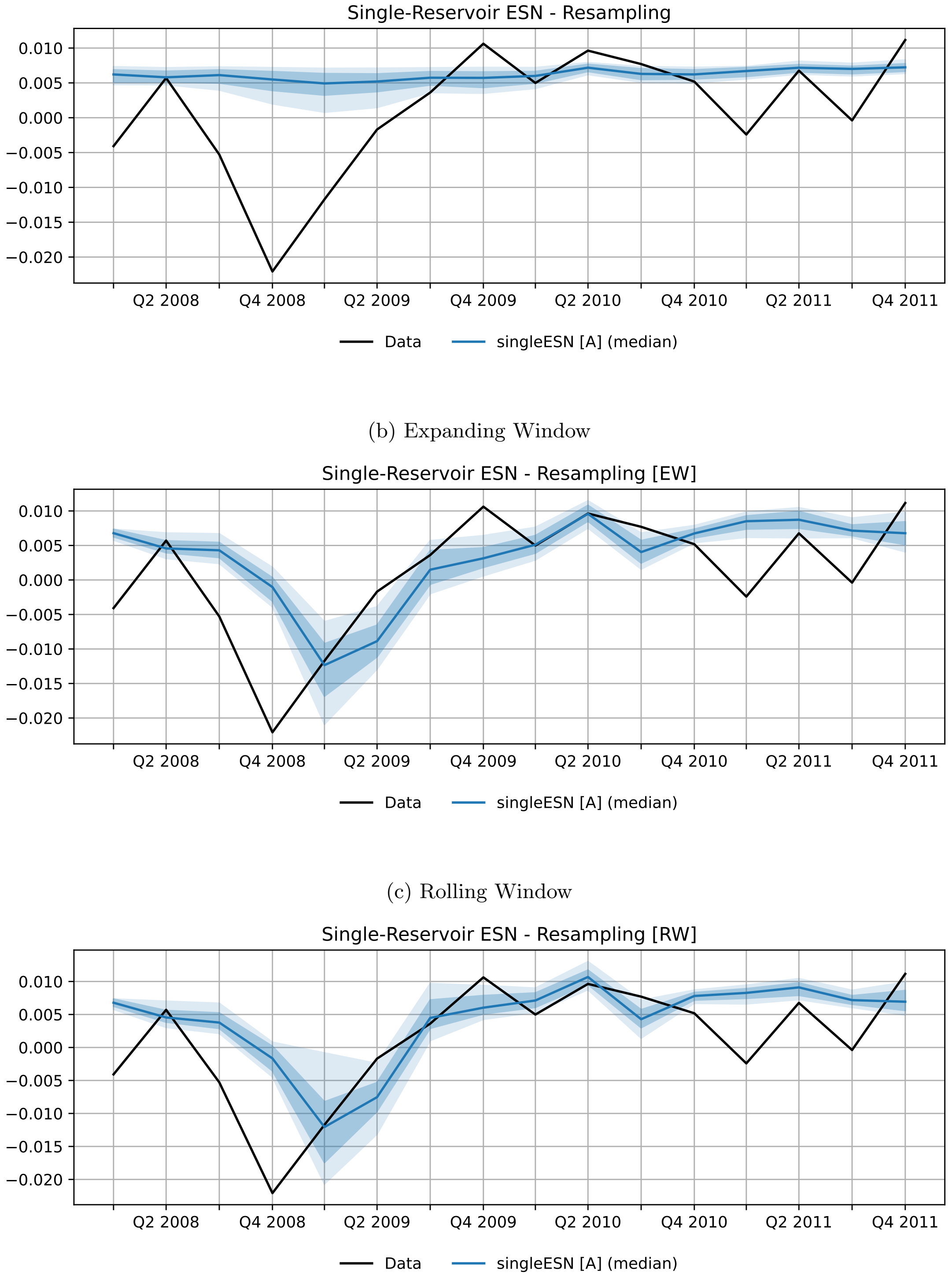


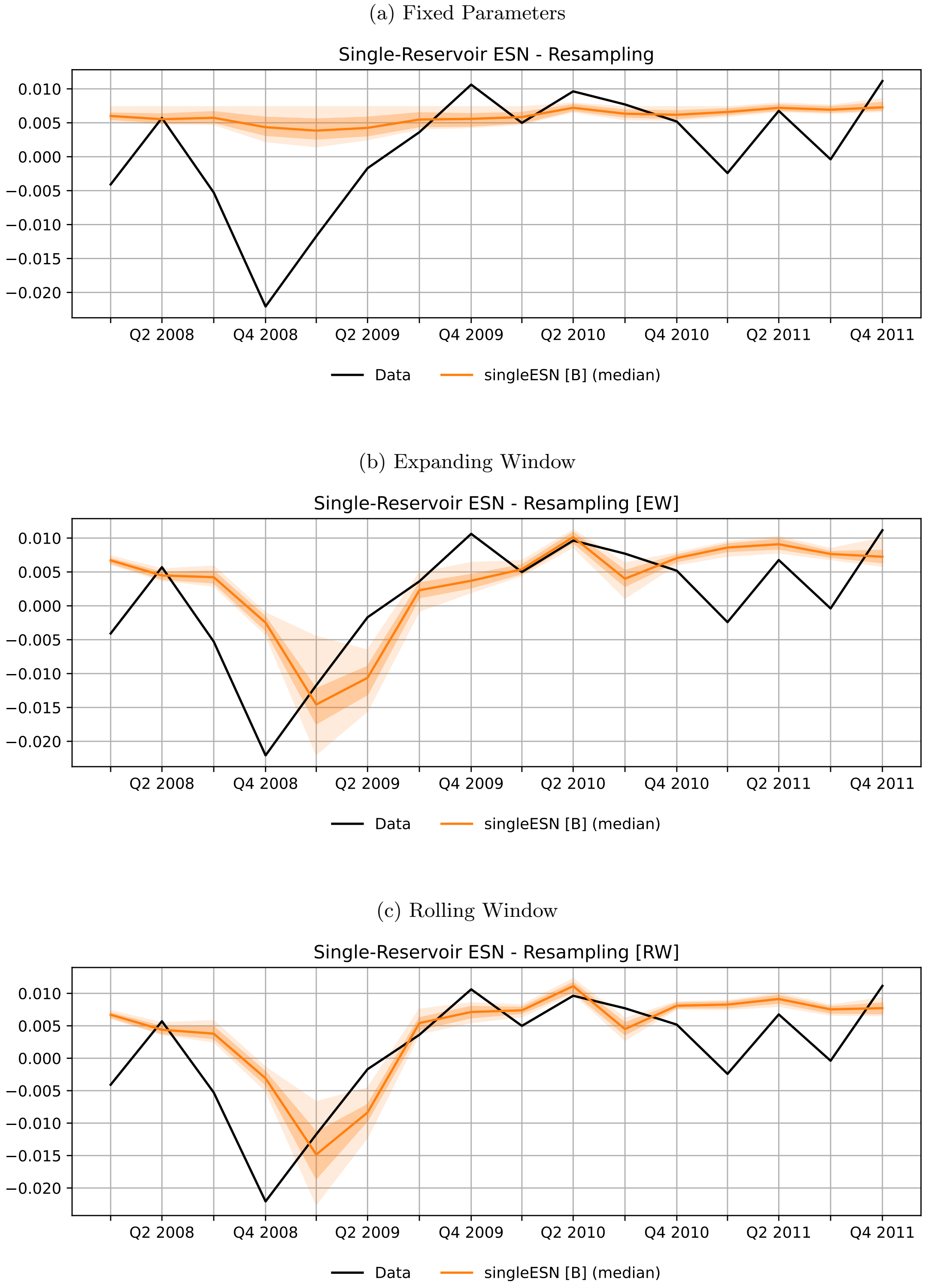
Figure 22: ESN Robustness Plots – 2007 Sample – Small-MD Dataset


(a) Fixed Parameters

(b) Expanding Window

(c) Rolling Window

Figure 23: ESN Robustness Plots – 2007 Sample – Small-MD Dataset

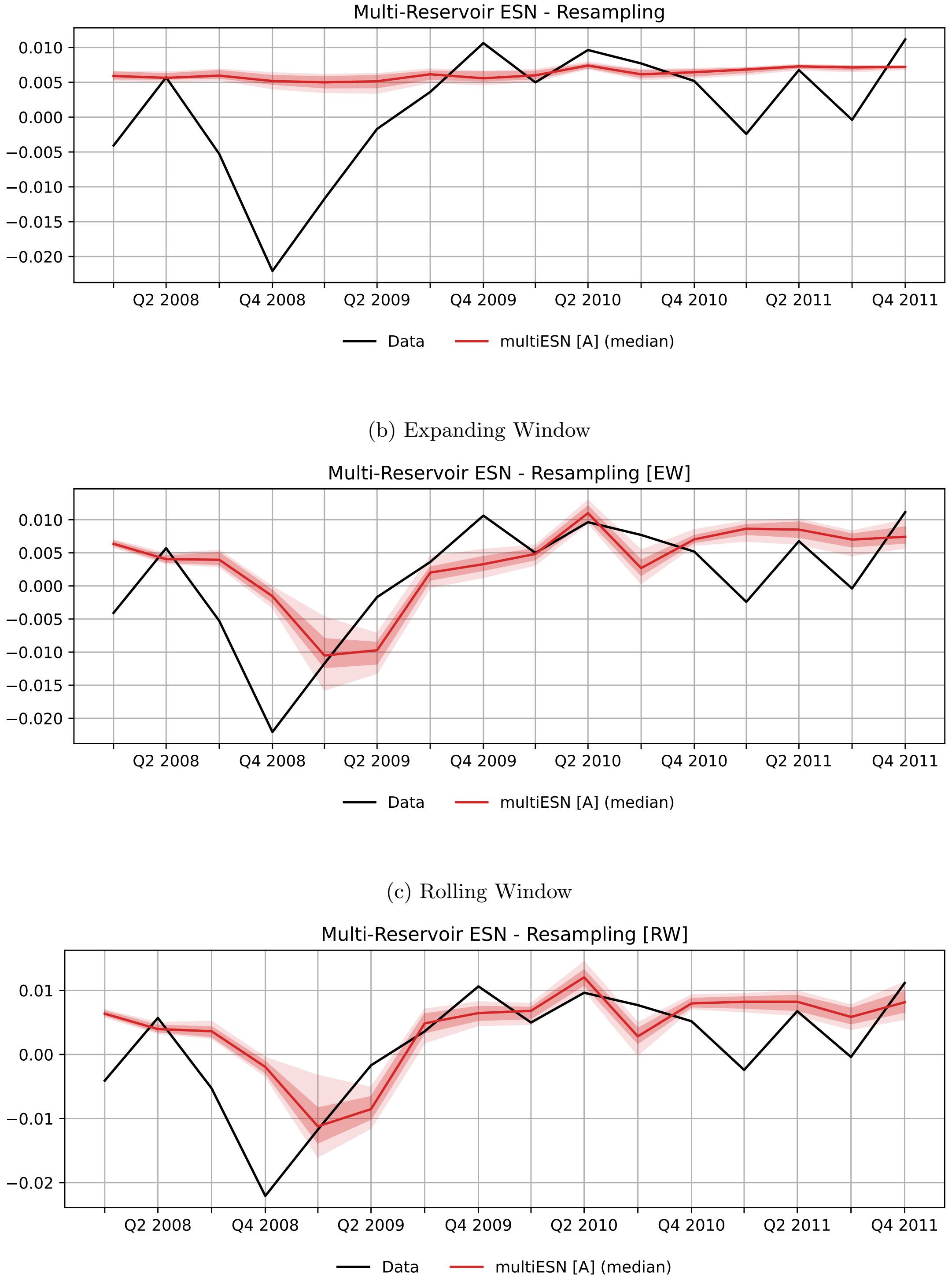


(a) Fixed Parameters

(b) Expanding Window

(c) Rolling Window

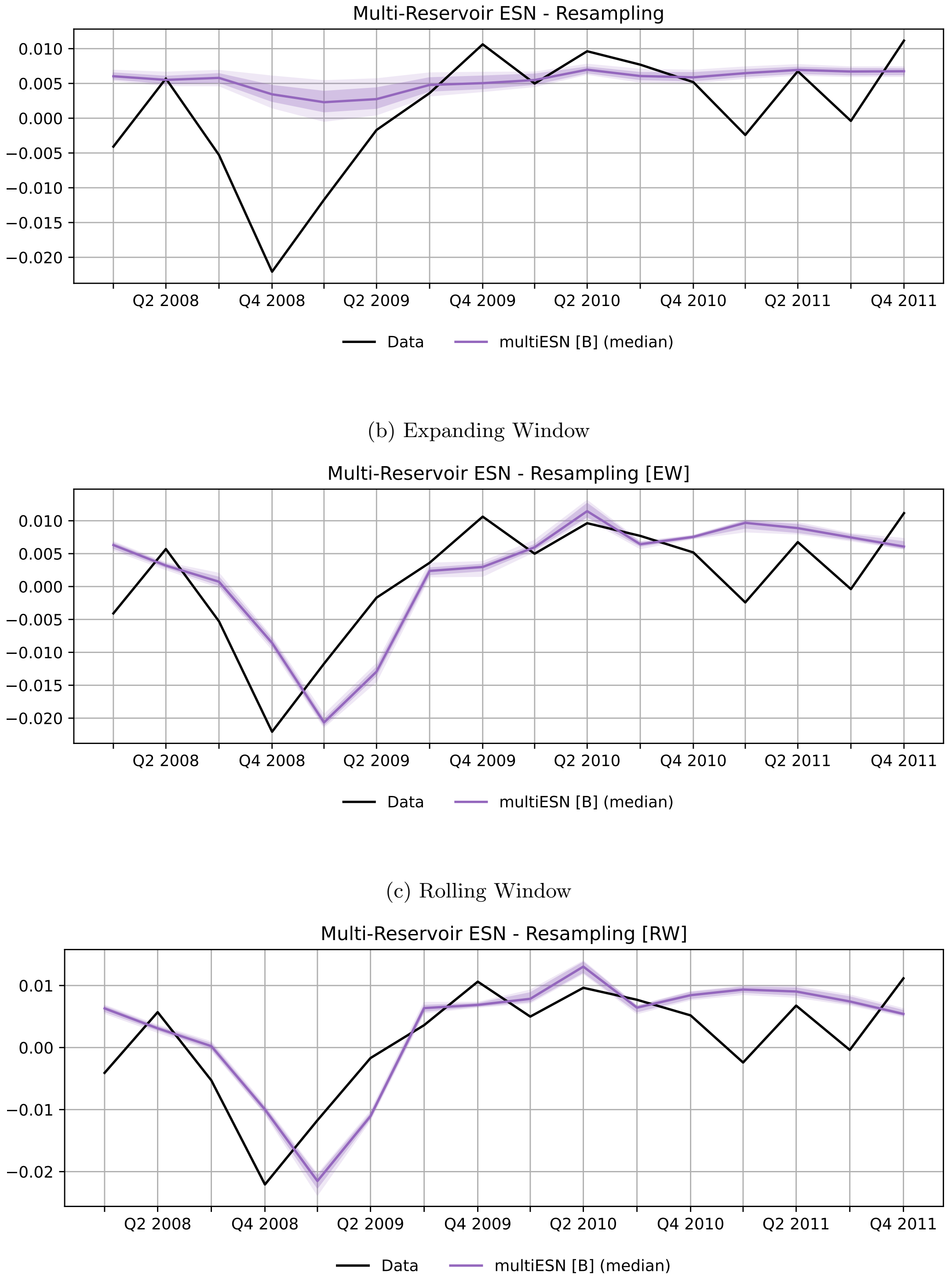
Figure 24: ESN Robustness Plots – 2007 Sample – Small-MD Dataset


(a) Fixed Parameters

(b) Expanding Window

(c) Rolling Window

Figure 25: ESN Robustness Plots – 2007 Sample – Small-MD Dataset

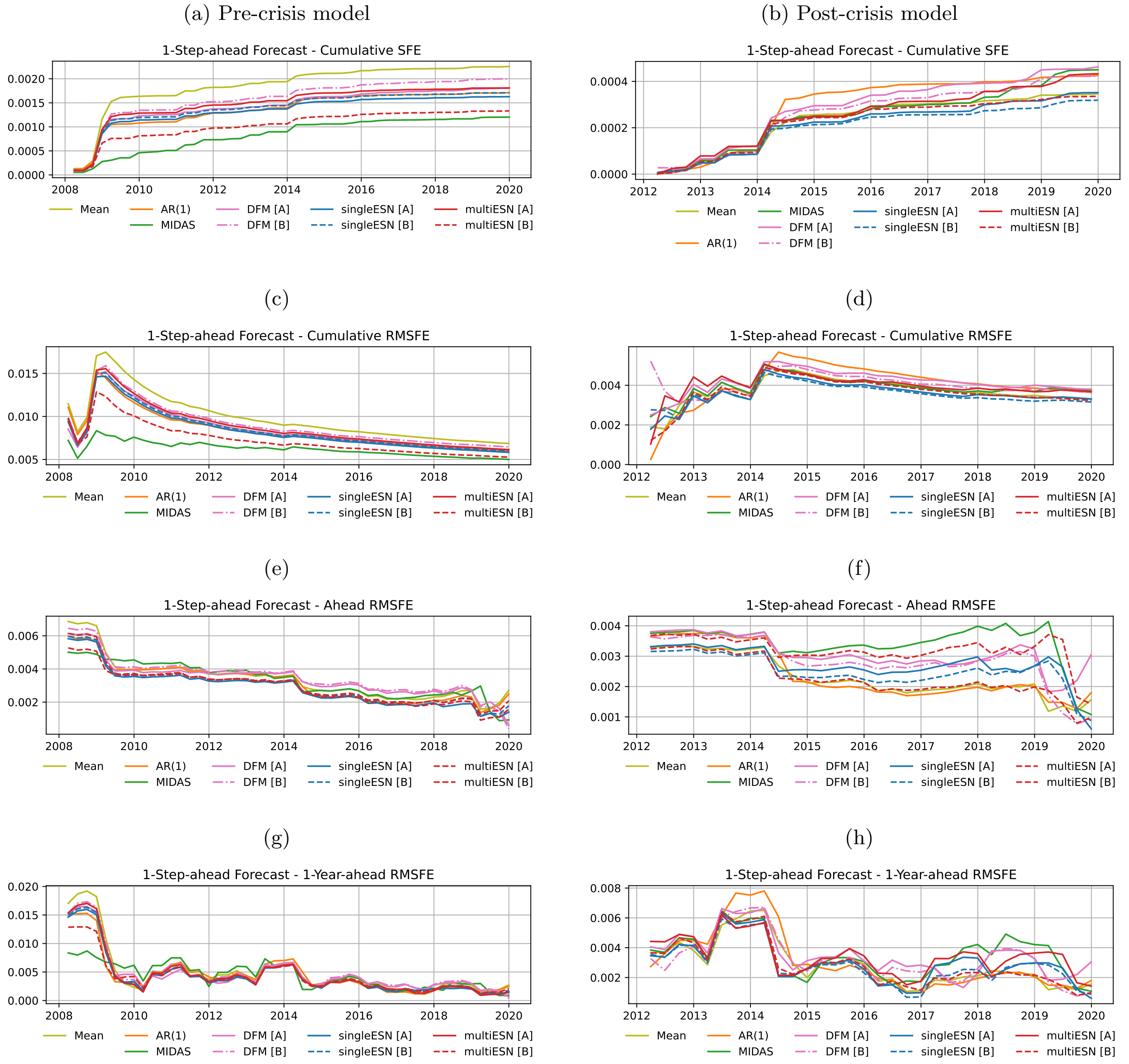


(a) Fixed Parameters

(b) Expanding Window

(c) Rolling Window

Figure 26: 1-Step-ahead GDP Forecasting, Fixed Parameters - Small-MD Dataset



(a) Pre-crisis model

(b) Post-crisis model

(c)

(d)

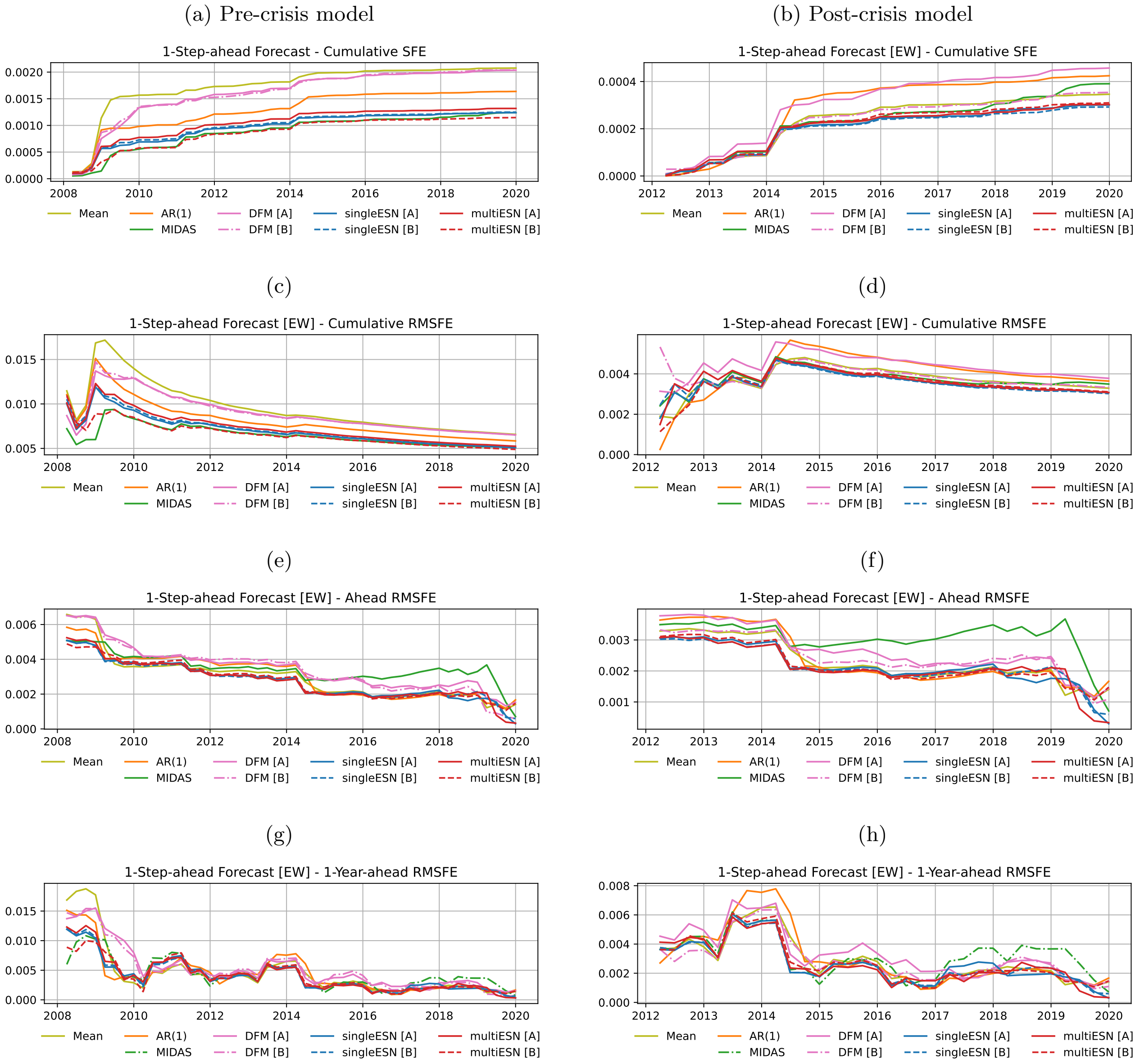
(e)

(f)

(g)

(h)

Figure 27: 1-Step-ahead GDP Forecasting, Expanding Window - Small-MD Dataset



(a) Pre-crisis model

(b) Post-crisis model

(c)

(d)

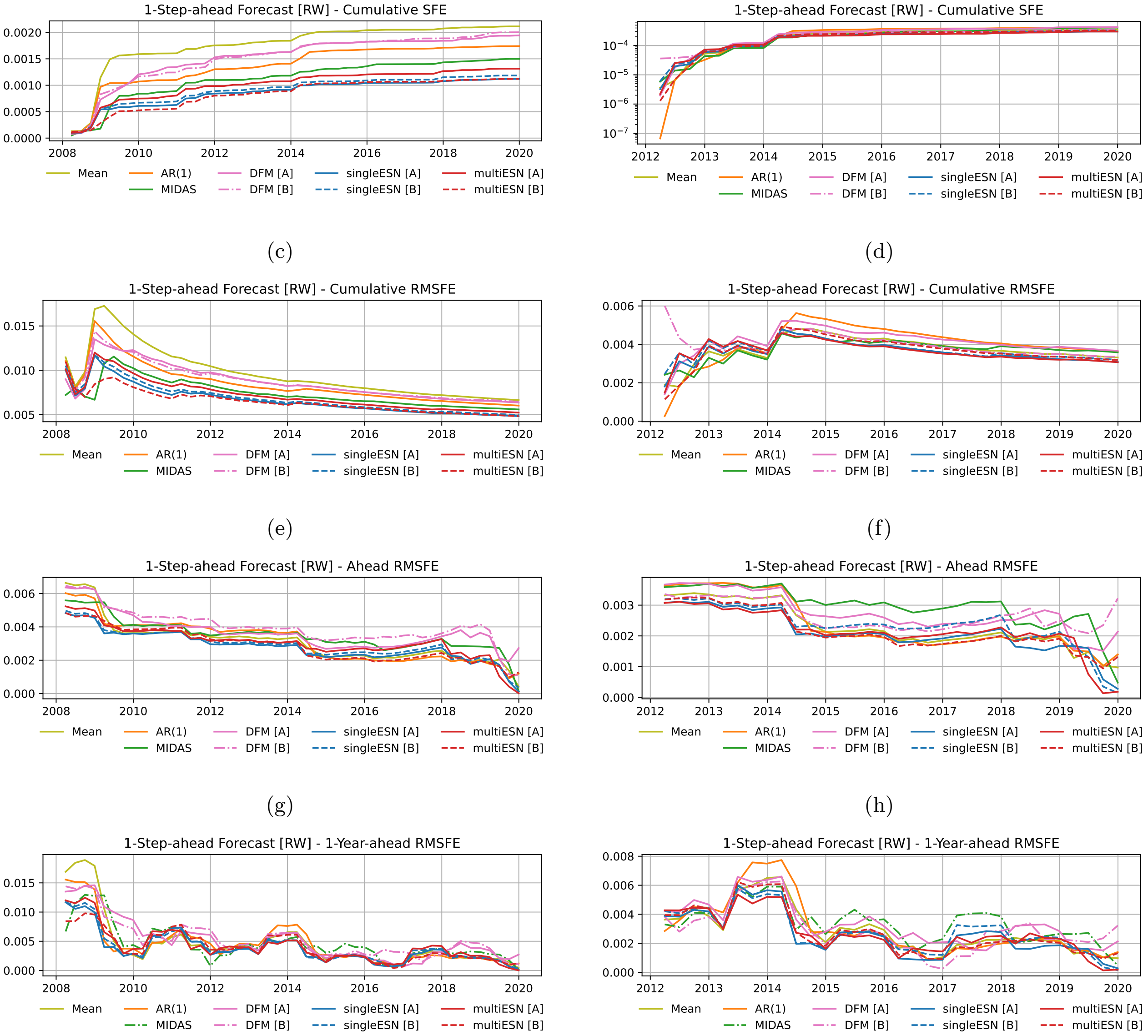
(e)

(f)

(g)

(h)

Figure 28: 1-Step-ahead GDP Forecasting, Rolling Window - Small-MD Dataset



(a) Pre-crisis model

(b) Post-crisis model

1-Step-ahead Forecast [RW] - Cumulative SFE

(c)

(d)

1-Step-ahead Forecast [RW] - Cumulative RMSFE

(e)

(f)

1-Step-ahead Forecast [RW] - Ahead RMSFE

(g)

(h)

1-Step-ahead Forecast [RW] - 1-Year-ahead RMSFE