CRED WORKING PAPER n^o 2026-02

Tail-Aware Density Forecasting of Locally Explosive Time Series: A Neural Network Approach

February, 2026

ELENA DUMITRESCU[*] JULIEN PEIGNON[†] ARTHUR THOMAS[‡]

[*]Université Paris-Panthéon-Assas, CRED, France.
[†]CEREMADE, CNRS, UMR 7534, Université Paris-Dauphine PSL University, Paris, France
[‡]Université Paris-Dauphine, Université PSL LEDa, CNRS, IRD, 75016 Paris, France

Tail-Aware Density Forecasting of Locally Explosive Time Series: A Neural Network Approach

Elena Dumitrescu*
University Paris-Panthéon-Assas, CRED, 75005 Paris, France
and
Julien Peignon
CEREMADE, CNRS, UMR 7534, Université Paris-Dauphine
PSL University, Paris, France
and
Arthur Thomas
Université Paris-Dauphine, Université PSL
LEDa, CNRS, IRD, 75016 Paris, France

February 11, 2026

Abstract

This paper proposes a Mixture Density Network specifically designed for forecasting time series that exhibit locally explosive behavior. By incorporating skewed t-distributions as mixture components, our approach offers enhanced flexibility in capturing the skewed, heavy-tailed, and potentially multimodal nature of predictive densities associated with bubble dynamics modeled by mixed causal-noncausal ARMA processes. In addition, we implement an adaptive weighting scheme that emphasizes tail observations during training and hence leads to accurate density estimation in the extreme regions most relevant for financial applications. Equally important, once trained, the MDN produces near-instantaneous density forecasts. Through extensive Monte Carlo simulations and two empirical applications, on the natural gas price and inflation, we show that the proposed MDN-based framework delivers superior forecasting performance relative to existing approaches.

Keywords: Forecasting, Noncausal Models, Mixture Density Networks

*We are grateful to Serge Darolles, Christian Francq, Laurent Ferrara, Gaëlle le Fol, Daniel Velasquez Gaviria, Christian Gouriéroux, Alain Hecq, Loïc Henry, Joann Jasiak, Sébastien Laurent, Yannick Le Pen, Gabriele Mingoli, Aryan Manafi Neyazi, Fabrice Rossi and Jean-Michel Zakoian. We also thank the seminar participants at Maastricht University School of Business and Economics, as well as the participants at the 19th CFE conference, the 24th Conference Développements Récents de l'Econométrie Appliquée à la Finance (University of Paris Nanterre), and the First Workshop in Noncausal Econometrics for helpful comments and discussions.

1 Introduction

Time-series forecasts using causal ARMA models have been playing a crucial role in economic and financial decision-making processes for a long time. A conclusion often reached in this context where the current value of the variable of interest is forced to depend only on its past is that one- and multi-step ahead forecasting in periods of high instability or in presence of forward looking behavior of economic agents is particularly difficult. Indeed, these causal models are characterized by mean reversion, *i.e.*, their forecasts converge to the unconditional mean even after an extreme event occurs, regardless of whether such behavior reflects the true underlying dynamics. In the specific case of financial prices, which we are interested in, the traditional properties of heavy-tailed marginal distributions and volatility clustering are at the core of forecasting models. But a close look at the dynamics of various types of asset prices, sometimes called speculative assets, reveals the presence of phases of locally explosive behaviour: rising patterns followed by a burst, local trends and spikes. Such non-linear characteristics are, however, very poorly captured by standard financial econometric models.

Recently, (causal-)noncausal autoregressive processes have been found to be suitable for modelling such locally explosive behaviour as they allow for dependence on the future. These simple linear models produce rich non-linear patterns without requiring non-linearities to be imposed *ex ante*. Importantly, noncausal processes are grounded in macroeconomic theory, as they arise as stationary solutions of rational expectation models under infinite variance (Gourieroux et al. 2020). Relevant applications of these models range from asset prices (see Fries & Zakoian 2019, Gourieroux & Zakoian 2017, Gourieroux & Jasiak 2018, Hecq & Velasquez-Gaviria 2025), to macroeconomic data (see Lanne & Saikkonen 2011, Davis & Song 2020), commodity prices (Blasques et al. 2025), climate risk on El Niño and La Niña (de Truchis et al. 2025), green stock prices (Giancaterini et al. 2025), and to electronic currency exchange rates (Cavaliere et al. 2020).

In this context, our paper proposes a novel forecasting methodology for causal–noncausal autoregressive processes based on a specifically-designed neural network architecture and training procedure. Indeed, building predictions with this class of models has been shown to be particularly difficult due to their dependence on future values. In fact, it has long been thought that the conditional predictive density of mixed causal-noncausal processes does not have closed-form and inference can only be performed by simulation-based or Bayesian methods (see Lanne et al. 2012, Gourieroux & Jasiak 2016, Nyberg & Saikkonen 2014). Although these approaches constitute flexible alternatives for predicting general noncausal processes, Hecq & Voisin (2021) highlight two main drawbacks: they become computationally intensive for long-horizon forecasts and do not accurately capture the dynamics of extreme events concentrated in the tails of the distribution. This limitation is highly problematic given that modeling explosive tail behavior is the primary motivation for using noncausal processes in the first place.

More importantly, noncausal processes exhibit highly non-linear and process-specific tail behavior that must be properly accounted for in forecasting. For example, when the conditional predictive density becomes multimodal, point forecasts become meaningless, as they fail to represent the fundamental dichotomy between bubble continuation and collapse (see Gourieroux & Zakoian 2017, Fries 2022, de Truchis et al. 2025, Gourieroux et al. 2025). For this reason, the noncausal literature as a whole, and this paper in particular, focuses exclusively on density forecasting rather than point prediction. Additionally, this tail behavior may also explain the difficulties encountered by standard numerical approaches when modelling extreme values (see Hecq & Voisin 2021) as well as the failure of state-of-the-art machine learning methods to accurately forecast locally explosive dynamics (Saïdi 2023).

This paper takes a new approach to forecasting noncausal processes, *i.e.*, locally explosive dynamics. We develop a specifically-designed neural network architecture to accurately estimate the full conditional predictive density of univariate general mixed causal-noncausal autoregressive moving average (MARMA) processes. More precisely, we introduce a newly tailored Mixture Density Network (MDN) based on skewed t-distributions as mixture components, which naturally captures both the multimodality and heavy-tailed asymmetric nature of predictive densities during explosive episodes. This contrasts with traditional MDNs (à la Bishop 1994), whose underlying Gaussian assumption cannot accommodate such features. Furthermore, since tail observations are inherently rare, we develop an adaptive weighting scheme that emphasizes extreme regions during training, enabling accurate density estimation in the parts of the distribution most relevant for financial applications.

Another significant advantage of our approach is that, once trained, the MDN produces near-instantaneous density forecasts for all forecasting horizons under analysis. By leveraging neural networks, this paper contributes to the growing literature on machine learning methods for economic and financial applications (Athey & Imbens 2019), and specifically for uncertainty quantification in forecasting. However, we do not pursue a strictly statistical path aimed at identifying true predictive densities or establishing formal properties of the forecasts, such as identifiability, asymptotic convergence, or Markov structure (see, e.g., Gouriéroux & Monfort 2025). We rather exploit the approximation power of neural networks to directly estimate the predictive densities. Our procedure accommodates short time series by proceeding in two steps: first, estimate the underlying noncausal model, and then learn the associated predictive density from simulated paths of this estimated process.

Nevertheless, one could directly apply our MDN approach to the raw data, provided that it is sufficiently long.

We compare our MDN with noncausal-specific density forecasting methods and with more general techniques through both extensive Monte Carlo simulations and two empirical applications, on U.S. natural gas price and inflation. In all settings, our approach outperforms competing methods while requiring substantially less computational time. It also achieves superior point-forecast accuracy compared to standard benchmarks in both applications.

The paper is structured as follows. Section 2 details our machine learning forecasting method. The Monte Carlo analysis is detailed in Section 3. Section 4 summarizes the two empirical illustrations, while Section 5 concludes.

2 Forecasting with Mixture Density Networks

Traditional Mixture Density Networks (Bishop 1994) provide a flexible framework for modeling conditional probability distributions by outputting the parameters of a Gaussian mixture model through a neural network. However, Gaussian mixtures struggle to adequately capture the heavy-tailed behavior of financial prices and lead to systematic underestimation of extreme event probabilities. To address this deficiency, we propose a new MDN, which relies on skewed t-distribution components instead of Gaussian ones to provide flexible parametric control over both skewness and tail heaviness. More precisely, it models the conditional density as:

$$p_h(X_{t+h}|\mathbf{X}_t) = \sum_{j=1}^{K} \pi_{j,h}(\mathbf{X}_t) \cdot f(X_{t+h}; \mu_{j,h}(\mathbf{X}_t), \sigma_{j,h}(\mathbf{X}_t), \xi_{j,h}(\mathbf{X}_t), \nu_{j,h}(\mathbf{X}_t)),$$

with $f(y;\mu,\sigma,\xi,\nu) = \frac{2}{\sigma} t\left(\frac{y-\mu}{\sigma};\nu\right) T\left(\xi \frac{y-\mu}{\sigma}\sqrt{\frac{\nu+1}{\nu+\frac{y-\mu}{\sigma}}};\nu+1\right)$ the skewed t-distribution probability density function, where $t(\cdot;\nu)$ denotes the Student-t density and $T(\cdot;\nu)$ its cumulative distribution function, both with ν degrees of freedom that control tail thickness. We denote the forecasting horizon by h and let $\mathbf{X}_t = (X_t, X_{t-1}, \ldots, X_{t-L+1})$ be a vector of L consecutive observations up to time t.[1] The other parameters are the location $\mu \in \mathbb{R}$, the scale $\sigma > 0$, and the shape $\xi \in \mathbb{R}$, which governs asymmetry.[2] These predictive densities provide a complete characterization of forecast uncertainty: they can be used to construct prediction intervals, assess tail risk probabilities during explosive episodes, and derive point forecasts. We develop our own network architecture and training strategy as follows.

2.1 Network Architecture

Our network architecture is lightweight and parsimonious, consisting of a fully connected multilayer perceptron (MLP) with two hidden layers of dimension 64 and five parallel output heads, one for each parameter of the skewed t-distribution mixture $(\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\xi}, \boldsymbol{\nu})$. This makes it efficient to train even on standard CPU hardware.[3] The hidden layers use rectified linear unit (ReLU) activations, a standard choice in deep learning due to its computational efficiency and ability to mitigate vanishing gradient issues (Goodfellow et al. 2016):

$$\begin{aligned}
\mathbf{z}^{(0)}(\mathbf{X}_t) &= \mathbf{X}_t \in \mathbb{R}^L, \\
\mathbf{z}^{(1)}(\mathbf{X}_t) &= \mathrm{ReLU}\left(\mathbf{W}^{(0)}\mathbf{z}^{(0)}(\mathbf{X}_t) + \mathbf{b}^{(0)}\right) \in \mathbb{R}^{64}, \\
\mathbf{z}^{(2)}(\mathbf{X}_t) &= \mathrm{ReLU}\left(\mathbf{W}^{(1)}\mathbf{z}^{(1)}(\mathbf{X}_t) + \mathbf{b}^{(1)}\right) \in \mathbb{R}^{64},
\end{aligned}$$

where $\mathbf{W}^{(0)} \in \mathbb{R}^{64\times L}$, $\mathbf{W}^{(1)} \in \mathbb{R}^{64\times 64}$, $\mathbf{b}^{(0)}, \mathbf{b}^{(1)} \in \mathbb{R}^{64}$, and $\mathrm{ReLU}(x) = \max(0, x)$ is applied element-wise. The five output heads map the final hidden representation $\mathbf{z}^{(2)}(\mathbf{X}_t)$ to the

[1] Intuitively, machine learning approaches can naturally leverage vectors of multiple past observations. Kernel estimators could also theoretically be extended to incorporate multiple conditioning lags, but at the expense of an exponential increase in computational complexity due to the curse of dimensionality. Throughout the rest of the paper, we mainly use $\mathbf{X}_t = X_t$ to insure a fair comparison across methods.

[2] We found this parameterization of the MDN to be numerically more robust than the Tukey g-and-h component-based approach of Guillaumin & Efremova (2024), as it does not need computing numerical inverse transforms via binary search, which can cause training instability when dealing with extreme values in the tails. Additionally, the skewed t-distribution admits well-established multivariate extensions (see Azzalini & Dalla Valle 1996), providing a natural pathway for extending our framework to vector-valued time series in future work.

[3] While recurrent architectures such as RNNs or LSTMs could capture additional temporal dependencies, we opt for this simpler feedforward structure to maintain computational efficiency and ease of implementation in applied settings.

mixture parameters:

$$
\begin{aligned}
\boldsymbol{\pi}(\mathbf{X}_t) &= \text{Softmax}(\mathbf{W}_\pi \mathbf{z}^{(2)}(\mathbf{X}_t) + \mathbf{b}_\pi) \in [0,1]^K, \ \sum_{j=1}^{K} \pi_j = 1 \\
\boldsymbol{\mu}(\mathbf{X}_t) &= \mathbf{W}_\mu \mathbf{z}^{(2)}(\mathbf{X}_t) + \mathbf{b}_\mu \in \mathbb{R}^K \\
\boldsymbol{\sigma}(\mathbf{X}_t) &= \text{Softplus}(\mathbf{W}_\sigma \mathbf{z}^{(2)}(\mathbf{X}_t) + \mathbf{b}_\sigma) \in \mathbb{R}_+^K \\
\boldsymbol{\xi}(\mathbf{X}_t) &= \mathbf{W}_\xi \mathbf{z}^{(2)}(\mathbf{X}_t) + \mathbf{b}_\xi \in \mathbb{R}^K \\
\boldsymbol{\nu}(\mathbf{X}_t) &= \text{Softplus}(\mathbf{W}_\nu \mathbf{z}^{(2)}(\mathbf{X}_t) + \mathbf{b}_\nu) \in \mathbb{R}_+^K,
\end{aligned}
$$

where $K = 10$ is the number of mixture components, and each output head has weight matrix $\mathbf{W}_\bullet \in \mathbb{R}^{K \times 64}$ and bias $\mathbf{b}_\bullet \in \mathbb{R}^K$. The softmax function, $\text{Softmax}(\mathbf{x})_j = e^{x_j} / \sum_{i=1}^{K} e^{x_i}$, maps an unconstrained K-vector to valid probability weights summing to one. The softplus function, $\text{Softplus}(x) = \log(1 + e^x)$, is applied element-wise and provides a smooth transformation ensuring strictly positive outputs for the scale (σ) and degree of freedom (ν) parameters. The location (μ) and skewness (ξ) parameters remain unconstrained to allow for arbitrary centering and both left and right asymmetry.

The model is implemented in PyTorch with weights initialized via the Kaiming uniform scheme (He et al. 2015), and trained by stochastic gradient descent (SGD) with the Adam optimizer (Kingma & Ba 2015), minimizing the negative log-likelihood of the mixture density. We add noise regularization during training by injecting small Gaussian perturbations to the input data. This smooths the estimated density by implicitly penalizing the Hessian of the log-likelihood, an approach promoted in the conditional density estimation literature (Rothfuss et al. 2019, 2020). This architecture has approximately 8,000 parameters, exceeding the number of observations available in both our simulations and empirical applications, and is therefore not identifiable. However, for a purely forecasting-oriented approach, this overparameterization is not a limitation but rather an advantage, as it can improve generalization without overfitting (Belkin et al. 2019).

2.2 Adaptive Weighting Function

We focus on financial prices rather than returns, as differencing eliminates the explosive dynamics that we aim to forecast. Prices naturally exhibit imbalance between normal market conditions and extreme events, yet learning algorithms prioritize frequent observations over rare tail events that are crucial for risk assessment. To address this, we combine resampling strategies (Avelino et al. 2024) with cost-sensitive learning (Steininger et al. 2021), which encourage the MDN to focus on extreme events during training.

Let $\mathcal{D}$ denote the training sample of size $T = |\mathcal{D}|$, and let the subset of tail events in the training sample be given by $\mathcal{E} = \{t \in \mathcal{D} : X_t < \ell \text{ or } X_t > u\}$. The set $\mathcal{E}$ is defined using only the scalar X_t, not the entire lagged vector $\mathbf{X}_t$ used as input. The boundaries ℓ and u are determined using the generalized boxplot methodology of Bruffaerts et al. (2014).[4] Then, the adaptive weight function is given by:

$$w_t = \begin{cases} \sqrt{\frac{T}{|\mathcal{E}|}}, & \text{if } t \in \mathcal{E} \\ 1, & \text{otherwise.} \end{cases} \tag{1}$$

The weighting mechanism operates at two levels during training, which allows the model to focus on rare tail events without requiring modifications to the underlying MDN architecture. First, during mini-batch construction, we give more weight to rare cases by employing a weighted random sampling with replacement approach.[5] Under this scheme, the probability that observation t is sampled in a mini-batch $\mathcal{B}$ is given by $P(t) = w_t / \sum_{i=1}^{T} w_i$. Second, within each mini-batch $\mathcal{B}$, we apply cost-sensitive learning through a weighted loss function. At each gradient step, the model parameters $\boldsymbol{\theta}$ are updated by minimizing the weighted

[4] $\ell = \xi^{\text{TGH}}_{\delta/2}, \quad u = \xi^{\text{TGH}}_{1-\delta/2}$, for a fixed detection rate δ, where ξ^{TGH} is the empirical Tukey g-h CDF fitted to a rank-preserving transformation of the observed data to accurately capture skewness and tail heaviness of the data.

[5] A mini-batch $\mathcal{B} \subset \mathcal{D}$ is a small random subset of training observations used to compute gradient updates at each iteration of the SGD.

negative log-likelihood,

$$\mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{\sum_{j\in\mathcal{B}} w_j} \sum_{j\in\mathcal{B}} w_t \log p_h(X_{j+h}|\mathbf{X}_j;\boldsymbol{\theta}),$$

where j indexes observations in mini-batch $\mathcal{B}$. This dual application of weights, both in sampling and loss computation, results in an effective weight of $T/|\mathcal{E}|$ for tail events. This justifies the square-root scaling in (1) to achieve the desired inverse proportion weighting.

2.3 Post-hoc Calibration

By reweighting the training distribution to emphasize extreme events, we train the model on a distribution that differs from the data-generating process. Specifically, when we assign higher weights to tail observations, the model learns to predict conditional distributions $\hat{p}_h(X_{t+h}|\mathbf{X}_t)$ that reflect this reweighted empirical distribution rather than the original empirical distribution. This shift causes systematic biases: the model overestimates the probability of extreme events across the entire input space, leading to miscalibrated predictions when evaluated with respect to the original, unweighted distribution.

To correct for miscalibration, we use the method of Dey et al. (2022) based on the uniformity property of the Probability Integral Transform (PIT), $\mathrm{PIT}_h(X_{t+h}, \mathbf{X}_t) = \int_{-\infty}^{X_{t+h}} \hat{p}_h(z|\mathbf{X}_t)\, dz = \hat{F}_h(X_{t+h}|\mathbf{X}_t)$, where $\hat{F}_h$ is the predicted cumulative distribution function. This recalibration procedure is applied after the initial model training on $\mathcal{D}$ and requires a distinct calibration dataset, $\mathcal{D}_{\mathrm{cal}}$. First, we use the trained model to compute the PIT values for all couples of observations $(X_{t+h}^{\mathrm{cal}}, \mathbf{X}_t^{\mathrm{cal}}) \in \mathcal{D}_{\mathrm{cal}}$. Then, the conditional distribution of PIT values is estimated as follows: for each threshold τ on a grid $\mathcal{G} \subset [0,1]$, a separate XGBoost classifier (Chen & Guestrin 2016) is trained to predict the binary outcome $\mathbf{1}\{\mathrm{PIT}_h(X_{t+h}, \mathbf{X}_t) \leq \tau\}$ from the calibration features $\mathbf{X}_t^{\mathrm{cal}}$. The predicted probability from each classifier provides an estimate of

$$\hat{\beta}_h(\tau|\mathbf{X}_t^{\mathrm{cal}}) = \mathbb{P}(\mathrm{PIT}_h(X_{t+h}, \mathbf{X}_t) \leq \tau|\mathbf{X}_t^{\mathrm{cal}}) = \mathbb{P}(\hat{F}_h(X_{t+h}^{\mathrm{cal}}|\mathbf{X}_t^{\mathrm{cal}}) \leq \tau|\mathbf{X}_t^{\mathrm{cal}}), \quad \tau \in [0,1].$$

Intuitively, $\hat{\beta}_h(\tau|\mathbf{X}_t^{\text{cal}})$ measures the empirical frequency with which the model's predicted CDF, evaluated at the true outcome, falls below τ for observations with similar features. Perfect calibration corresponds to $\hat{\beta}_h(\tau|\mathbf{X}_t^{\text{cal}}) = \tau$ for all τ, which would indicate that the PIT values are uniformly distributed conditionally on $\mathbf{X}_t^{\text{cal}}$.

Once the correction function $\hat{\beta}_h$ is learned on $\mathcal{D}_{\text{cal}}$, it can be applied to any new couple of observations $(X_{t+h}^{\text{test}}, \mathbf{X}_t^{\text{test}}) \in \mathcal{D}_{\text{test}}$ from the test set or future data. The final recalibrated PDF, $\hat{p}_h^{\text{recal}}$, is obtained by applying the correction and then renormalizing,

$$\hat{p}_h^{\text{recal}}(X_{t+h}^{\text{test}}|\mathbf{X}_t^{\text{test}}) = \frac{c_h(X_{t+h}^{\text{test}}|\mathbf{X}_t^{\text{test}}) \cdot \hat{p}_h(X_{t+h}^{\text{test}}|\mathbf{X}_t^{\text{test}})}{\int c_h(z|\mathbf{X}_t^{\text{test}}) \cdot \hat{p}_h(z|\mathbf{X}_t^{\text{test}})\, dz},$$

with correction factor $c_h(X_{t+h}^{\text{test}}|\mathbf{X}_t^{\text{test}}) = \frac{d\hat{\beta}_h}{d\tau}\Big|_{\tau=\hat{F}_h(X_{t+h}^{\text{test}}|\mathbf{X}_t^{\text{test}})}$. The correction factor is derived from a change-of-variables formula which ensures that the recalibrated density produces uniformly distributed PIT values.

3 Monte Carlo Simulations

In this section, we evaluate the forecasting performance of our MDN approach through Monte Carlo simulations on various MARMA specifications. These controlled settings allow us to benchmark against theoretical predictive densities when available, or against theoretical predictive moments. We assess the relative performance of our method by conducting a horse race against established forecasting approaches.

3.1 MARMA Processes

Mixed Causal-Noncausal Autoregressive Moving Average processes naturally capture key distributional features of financial price series: multimodality, skewness, and heavy tails. Let X_t $(t = 0, \pm 1, \pm 2, \ldots)$ be a stochastic process generated by

$$\psi(F)\phi(B)X_t = \theta(F)H(B)\varepsilon_t, \tag{2}$$

where F (resp. $B = F^{-1}$) denotes the forward (resp. backward) operator, $\psi(z) = 1 - \psi_1 z - \ldots - \psi_p z^p$, $\phi(z) = 1 - \phi_1 z - \ldots - \phi_q z^q$, $\theta(z) = 1 - \theta_1 z - \ldots - \theta_r z^r$, and $H(z) = 1 - H_1 z - \ldots - H_s z^s$ are polynomials with roots outside the unit circle, and $(\varepsilon_t)_{t\in\mathbb{Z}}$ is a sequence of i.i.d. variables. Equation (2) admits a unique stationary solution, called a $MARMA(p, q, r, s)$, if $\phi(z) \neq 0$ and $\psi(z) \neq 0$ for all $\mid z \mid \leq 1$, and if ψ (resp. ϕ) has no common root with θ (resp. H). A sufficient condition for the identification of the model in (2) is that ε_t is i.i.d. non-gaussian (see Rosenblatt 2000).[6]

The literature has been using either α-stable or t-distributed innovations, the choice of one or another driving the forecasting algorithms proposed so far in the literature. We focus on α-stable MARMA models for two reasons. First, the α-stable family offers considerable flexibility, encompassing distributions ranging from Gaussian ($\alpha = 2$, $\beta = 0$) to Cauchy ($\alpha = 1$, $\beta = 0$) as special cases. Second, and most importantly, unlike the t-distribution case, theoretical results on predictive densities and moments are available in this setting. Specifically, Gourieroux & Zakoian (2017) derived a closed-form expression for the predictive conditional density of a noncausal MAR(0, 1) when $\varepsilon_t \overset{\text{i.i.d.}}{\sim}$ Cauchy. More generally, when ε_t follows an α-stable law, $\varepsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{S}(\alpha, \beta, \sigma, \mu)$, with $\alpha > 1$, $\beta \in [-1, 1]$, and $\sigma > 0$, the theoretical results of Fries (2022) yield closed-form expressions for higher-order conditional moments: $\mathbb{E}\Big[X_{t+h}^p \mid X_t = x\Big]$ for any integer p satisfying $1 \leq p < 2\alpha + 1$. This provides a rigorous framework for evaluating density forecasts in the following subsections.

[6] To the best of our knowledge, our MDN is the first noncausal-specific method capable of forecasting the conditional predictive density of general MARMA processes, as existing approaches are restricted to MAR specifications.

3.2 Simulation Design

We generate time series of length 5,000 from the following MARMA data generating processes (de Truchis & Thomas 2025):

$$\text{MAR(0,1):} \quad (1-0.9F)X_t = \varepsilon_t, \tag{3}$$

$$\text{MAR(0,2):} \quad (1-0.7F-0.1F^2)X_t = \varepsilon_t, \tag{4}$$

$$\text{MAR(1,1):} \quad (1-0.9F)(1-0.1B)X_t = \varepsilon_t, \tag{5}$$

$$\text{MARMA(1,1,1,1):} \quad (1-0.9F)(1+0.3B)X_t = (1-0.4F)(1+0.3B)\varepsilon_t, \tag{6}$$

where $\varepsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{S}(\alpha, 0, 0.5, 0)$.[7] In the simulations, we set $\alpha \in \{1.0, 1.2, 1.4, 1.8\}$, where smaller values correspond to fatter tails, which allows us to examine how model performance changes with the degree of tail heaviness.

The forecasting abilities of our MDN approach are compared with those of a set of established conditional density forecasting methods spanning different methodological paradigms: the nonparametric Nadaraya-Watson kernel density estimator (Rosenblatt 1969), the simulation-based approach of Lanne et al. (2012) and the closed-form predictive densities of Gourieroux & Jasiak (2025), both designed for MAR processes, as well as the learning-based FlexZBoost method of Izbicki (2017) and Dalmasso et al. (2020).[8] An additional calibration set $\mathcal{D}_{\text{cal}}$ of 5,000 observations, generated by the same data-generating process, is used to perform the local recalibration procedure described in Section 2.3 for the MDN.

To ensure a fair comparison across all methods, the kernel densities required by Nadaraya-Watson and the noncausal state density needed for the closed-form approach of Gourieroux & Jasiak (2025) are both estimated using the same 5,000-observation training set. The simulation-based method of Lanne et al. (2012) does not require additional historical

[7] The parameter values are chosen by following de Truchis et al. (2025) for the three MAR models and based on Fries (2022) for the MARMA specification. All specifications satisfy the stationarity conditions.

[8] For a thorough presentation of these methods, see the Online Appendix.

observations beyond the parameter estimation step and can be applied directly to the evaluation grid once the MAR and α-stable parameters are obtained. Note that for Lanne et al. (2012) and Gourieroux & Jasiak (2025), the MAR specification is assumed known, thereby circumventing model identification issues. We estimate the MAR process parameters using the Generalized Covariance (GCov) estimator (Gourieroux & Jasiak 2023), a semi-parametric approach that minimizes a portmanteau statistic based on the autocovariances of transformed residuals. The parameters of the α-stable distribution are then obtained by fitting the characteristic function-based estimator of Nolan (2020) to the filtered residuals.[9] In contrast, the nonparametric and learning-based methods, Nadaraya-Watson, FlexZBoost, and our MDN, do not require explicit parametric model estimation and directly learn the predictive density from the data.[10]

3.3 Bimodality Analysis and Sampled Trajectories

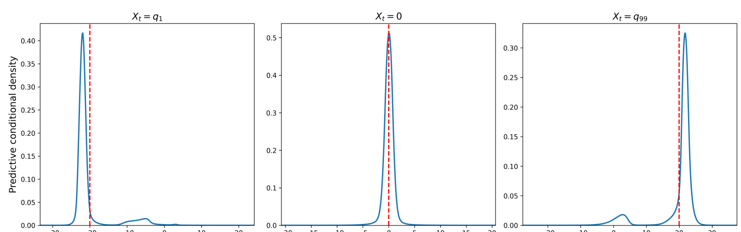
As shown by de Truchis et al. (2025), for any MAR process with a single anticipative root, when conditioning on a single observation, the conditional predictive density theoretically exhibits bimodality in the tail regions. This reflects the fundamental dichotomy of locally explosive dynamics: the bubble either continues or crashes, with probability $|\psi_1|^{\alpha h}$ and $1-|\psi_1|^{\alpha h}$ respectively (see Section 2.2 in the Online Appendix for a discussion). Accordingly, the predictive density features two modes: one near the unconditional mean (zero in our simulations), corresponding to the crash scenario, and one further from the conditioning value, corresponding to bubble continuation.

[9]This is an optimal framework to account for parameter estimation uncertainty, whereas in real-life applications model risk also matters and can further hamper the forecasting abilities of these approaches.

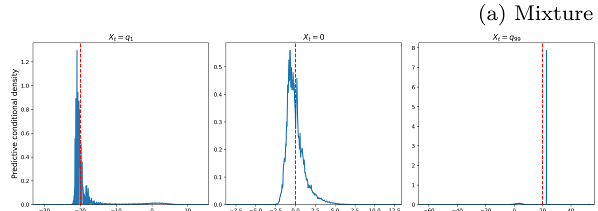
[10]To assess the impact of parameter estimation uncertainty, we conducted additional simulations where the methods of Lanne et al. (2012) and Gourieroux & Jasiak (2025) were evaluated using the true data-generating parameters rather than estimated ones. The relative performance rankings remained unchanged, indicating that the observed differences in forecasting accuracy are not primarily attributable to parameter estimation error.

Before implementing the forecast evaluation horse-race, we investigate whether the competing approaches capture well this theoretical bimodality of the conditional predictive density, particularly in the tail region. The procedure relies on a grid of 5,000 equispaced conditioning values $\{x_1, \ldots, x_{5000}\}$ spanning the quantile range $[q_{0.01}, q_{0.99}]$, where q_τ denotes the τ-quantile of the theoretical marginal distribution. For each point x_i on the grid, we estimate the conditional predictive density $\hat{p}_h(X_{t+h} \mid X_t = x_i)$ using each of the competing methods and visually assess their ability to recover the bimodal structure.

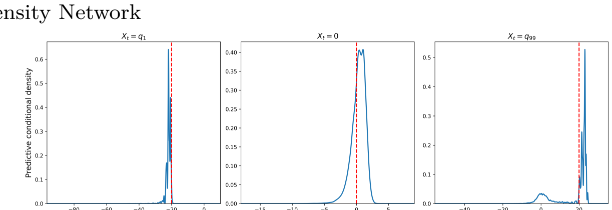
Figure 1 displays the one-step-ahead conditional predictive density for a MAR(0,1) process under three conditioning scenarios: $X_t = q_{0.01}$ (left tail), $X_t = 0$ (center), and $X_t = q_{0.99}$ (right tail). The MDN (panel a) successfully captures the expected bimodality,



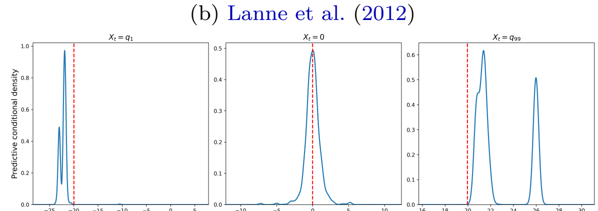
(a) Mixture Density Network



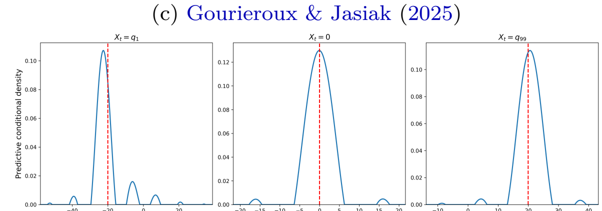
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



(d) Nadaraya-Watson



(e) FlexZBoost

Figure 1: 1-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process

Notes: This figure displays the estimated conditional predictive density $\hat{p}_h(X_{t+h} \mid \mathbf{X}_t)$ for a purely noncausal MAR(0,1) process with α-stable innovations at three conditioning values: $X_t = q_1$ (left tail), $X_t = 0$ (center), and $X_t = q_{99}$ (right tail), where $q_.$ denotes the percentiles. The red dashed line indicates the conditioning value X_t. The tail-index is fixed at $\alpha = 1.4$.

exactly as predicted by the geometric decay of $|\psi_1|^{\alpha h}$.[11] Moreover, employing skewed t-

[11]The bimodality becomes increasingly pronounced as the horizon extends from $h = 1$ to $h = 5$, and it is best captured by the MDN approach (see Section 3 in the Online Appendix).

distribution mixture components proves beneficial: the MDN naturally outputs asymmetric and heavy-tailed predictive densities. In contrast, Nadaraya-Watson (panel d) produces a highly erratic predictive density, with substantial gaps and spikes, especially in data-sparse tail regions. The approaches of Lanne et al. (2012) and Gourieroux & Jasiak (2025) exhibit similar irregular behavior. FlexZBoost (panel e) yields smoother estimates than kernel methods but displays small spurious bumps due to its inability to set certain cosine basis coefficients to zero. This prevents it from achieving the clean bimodal structure predicted by theory and confirms that general-purpose density estimators require substantial adaptation to capture locally explosive patterns. The Online Appendix, includes GIFs (Graphics Interchange Format) depicting the evolution of our MDN predictive densities across the entire grid of 5,000 conditioning points, for $h \in \{1, 2, 5\}$.

Beyond density estimation, the MDN's ability to capture bimodality translates directly into realistic trajectory generation. Figure 2 displays a trajectory obtained by iteratively sampling from the one-step-ahead predictive density, starting from an initial $X_0 = 0$ and using each sampled value as the next conditioning point. The resulting path exhibits both locally explosive dynamics and abrupt reversals that define noncausal processes, mirroring the behavior of a true MAR(0,1) simulation. Remarkably, when estimating the parameters of the MDN-sampled trajectory, one recovers a MAR(0,1) structure with coefficients close to the true data-generating process (see the Online Appendix for estimation details). We further demonstrate in the same appendix that this sampling procedure extends to higher-order anticipative models by providing an illustration for a MAR(0,2) process conditional on $\mathbf{X}_t = [X_t, X_{t-1}]$. This underscores our model's capacity to effectively utilize multiple conditioning variables.

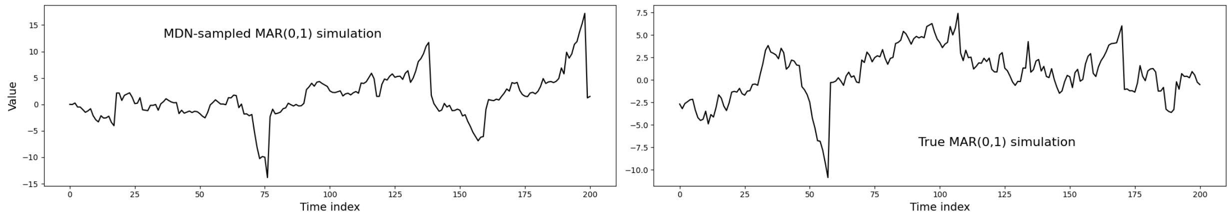


Figure 2: MDN-Sampled vs. True MAR(0,1) Trajectories

Notes: Left panel: trajectory generated by iteratively sampling from the MDN's one-step-ahead predictive density. Right panel: true MAR(0,1) simulation with $\alpha = 1.4$. The underlying MDN has been trained on a simulated sample of 5,000 realizations from the true MAR(0,1) specification in Equation (3), with $\alpha = 1.4$.

3.4 Simulation Results

Ideally, comparing the alternative forecasting approaches introduced in Section 3.2 would involve evaluating the competing predictive densities against the true theoretical one. However, since the latter is unavailable for most noncausal processes, our simulation analysis proceeds along three complementary directions.

First, we exploit a unique feature of the Cauchy MAR(0,1) process ($\alpha = 1.0$, $\beta = 0$): the availability of a closed-form expression for the conditional predictive density (Gourieroux & Zakoian 2017). This special case provides a direct benchmark for assessing how well each method approximates the true density. Second, for the general case of MARMA processes where closed-form densities are unavailable, we rely on theoretical conditional moments to evaluate predictive accuracy. Specifically, we compute the estimated conditional moments by numerically integrating the estimated predictive densities, and compare them to their theoretical counterparts. Note that our grid-based approach (see Section 3.3) is naturally suited to univariate conditioning, *i.e.*, predicting X_{t+h} based solely on the current value X_t, which is consistent with the theoretical conditional predictive densities of Gourieroux & Zakoian (2017), and the conditional moments of Fries (2022). Accordingly, in all simulations, the different methods condition only on the last observed value, $\mathbf{X}_t = X_t$. Third, we complement the grid-based evaluation with an out-of-sample forecasting exercise that assesses the predictive densities against realized outcomes, thereby better reflecting real-world forecasting applications.

3.4.1 Case 1: Cauchy MAR(0,1) model

Table 1 compares the estimated predictive densities with the true density of a MAR(0,1) process with $\alpha = 1.0$ and $\beta = 0$, using Kullback-Leibler (KL) divergence and Integrated Squared Error (ISE) metrics.[12]

Table 1: KL Divergence and ISE Between the True Predictive Density and the Predicted Ones: the Case of MAR(0, 1)

Horizon	Model	KL Divergence			ISE		
		Center	Tails	Total	Center	Tails	Total
$h = 1$	Nadaraya-Watson	**0.578**	11.45	10.39	**0.020**	0.751	0.680
	Lanne et al. (2012)	2.339	13.10	12.06	0.418	0.416	0.416
	Gourieroux and Jasiak (2025)	2.512	8.392	7.823	0.441	0.600	0.585
	FlexZBoost	2.877	**3.410**	**3.359**	0.268	**0.225**	**0.229**
	Mixture Density Network	**0.927**	**1.220**	**1.192**	**0.197**	**0.182**	**0.184**
$h = 2$	Nadaraya-Watson	**0.751**	14.47	13.14	**0.017**	0.701	0.635
	Lanne et al. (2012)	2.457	11.81	10.90	0.191	0.230	0.226
	Gourieroux and Jasiak (2025)	2.924	8.685	8.127	0.292	0.474	0.456
	FlexZBoost	2.236	**3.294**	**3.192**	0.121	**0.084**	**0.087**
	Mixture Density Network	**0.482**	**0.731**	**0.707**	**0.063**	**0.057**	**0.058**
$h = 5$	Nadaraya-Watson	**1.109**	18.23	16.58	**0.018**	0.760	0.688
	Lanne et al. (2012)	2.698	10.50	9.749	0.071	0.150	0.143
	Gourieroux and Jasiak (2025)	2.445	8.702	8.097	0.211	0.393	0.375
	FlexZBoost	1.485	**1.368**	**1.380**	0.041	**0.017**	**0.019**
	Mixture Density Network	**0.129**	**0.546**	**0.506**	**0.008**	**0.012**	**0.011**

Note: This Table reports the Kullback-Leibler (KL) divergence and Integrated Squared Error (ISE) between the true predictive density and estimated densities. Metrics are evaluated over three spatial regions: Center $[q_{0.1}, q_{0.9}]$, Tails $[q_{0.01}, q_{0.1}] \cup [q_{0.9}, q_{0.99}]$, and Total $[q_{0.01}, q_{0.99}]$, where q_p represents the p-th quantile. Best method in red, second best in **bold black**.

Using the grid-based methodology described above, we estimate the conditional predictive density $\hat{p}_h(X_{t+h} \mid X_t = x_i)$ for each competing method and directly compare it with the true density $p_h(X_{t+h} \mid X_t = x_i)$ across three distinct regions: the center $[q_{0.10}, q_{0.90}]$, the tails $[q_{0.01}, q_{0.10}] \cup [q_{0.90}, q_{0.99}]$, and the total range $[q_{0.01}, q_{0.99}]$.

[12]The KL divergence is defined as $\mathrm{KL}(p_h \| \hat{p}_h) = \int p_h(X_{t+h} \mid \mathbf{X}_t) \log\left(\frac{p_h(X_{t+h}|\mathbf{X}_t)}{\hat{p}_h(X_{t+h}|\mathbf{X}_t)}\right) dX_{t+h}$, while the ISE is given by $\mathrm{ISE} = \int \left(p_h(X_{t+h} \mid \mathbf{X}_t) - \hat{p}_h(X_{t+h} \mid \mathbf{X}_t)\right)^2 dX_{t+h}$.

The MDN exhibits the lowest KL divergence and ISE in the tail region and over the full distribution across all forecast horizons. In the center of the distribution, Nadaraya-Watson remains competitive, achieving the lowest KL divergence and ISE at $h = 1$, and the lowest ISE at $h = 2$. However, it performs substantially worse for density forecasting in the tails, especially at longer horizons. At horizon $h = 5$, the MDN achieves a tail KL divergence of 0.546 compared to 18.23 for Nadaraya-Watson and 8.702 for Gourieroux & Jasiak (2025). Across all horizons and regions examined, the MDN consistently delivers either the best or second-best performance.

3.4.2 Case 2: Predictive Moments Approach

The root mean square error (RMSE) criterion is now employed to compare the estimated conditional moments, $\hat{\mathbb{E}}[X_{t+h}^k \mid X_t = x_i]$ to the theoretical ones, $\mathbb{E}[X_{t+h}^k \mid X_t = x_i]$, of Fries (2022) and measure the forecasting performance for each tail index $\alpha \in \{1.0, 1.2, 1.4, 1.8\}$ and each moment order $k \in \{1, 2, 3, 4\}$, for all four DGPs in the same three regions of the distribution as in the first case-scenario. To formally assess the statistical significance of relative performance differences, we employ the Model Confidence Set (MCS) procedure of Hansen et al. (2011) with a 90% confidence level.

Note that, the forecast methods of Gourieroux & Jasiak (2025) and Lanne et al. (2012) require conditioning on multiple lags when the autoregressive order exceeds one, making direct comparison with Fries (2022)'s theoretical moments inappropriate beyond MAR(0,1) and for MARMA models. For this reason, this second-case comparative analysis is structured as follows. For the MAR(0,1) process, the MDN is compared with the full set of alternatives at all forecast horizons. For higher-order processes, MAR(0,2), MAR(1,1) and MARMA(1,1,1,1), the comparison is restricted to Nadaraya-Watson and FlexZBoost due to the conditioning set discrepancy.

Tables 2, 3, 4, and 5 present the one-period-ahead RMSE results for all four data-generating processes. The MDN approach almost always exhibits the lowest RMSE in the tail region and over the full distribution, regardless of the tail index α and the conditional moment order.[13] In the center of the distribution, Nadaraya-Watson consistently achieves the lowest RMSE across almost all configurations, benefiting from regions where the conditional distribution exhibits more regular behavior. Importantly, when the MDN does not achieve the lowest RMSE in the center, it consistently ranks as the second-best method. The more traditional forecasting methods of Lanne et al. (2012) and Gourieroux & Jasiak (2025), as well as FlexZBoost are almost always dominated in all regions.

Table 2: Root Mean Squared Error of Predictive Moments: MAR(0, 1) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.434***	43.72	41.55	**10.80***	6619	6291	–	–	–	–	–	–
	1.2	**0.097***	7.182	6.607	**1.360***	341.8	314.4	**12.75***	1.671e+04	1.538e+04	–	–	–
	1.4	**0.075***	2.313*	2.038*	**0.431***	49.86*	43.92*	**2.798***	1125	991.2	–	–	–
	1.6	**0.047***	**0.879**	**0.727**	**0.167***	12.20	10.09	**0.523***	149.7	123.8	**4.584***	1786	1477
	1.8	**0.037***	**0.164**	**0.125***	**0.095***	**1.149**	**0.863***	**0.334***	**6.941***	**5.201***	**1.642***	**41.28***	**30.92***
Lanne et al. (2012)	1.0	1.115	39.51	37.55	133.1	7320	6956	–	–	–	–	–	–
	1.2	0.391	12.90	11.87	5.406	649.5	597.6	46.55	3.127e+04	2.877e+04	–	–	–
	1.4	0.218	4.369	3.850*	1.602	96.61	85.10	9.985	2027	1785	–	–	–
	1.6	0.121	1.010	0.838	0.679	14.71	12.17	3.171	184.1	152.3	17.51	2234	1848
	1.8	0.076	0.220	0.172	**0.380**	2.482	1.875	1.468	20.93	15.69	5.417	159.3	119.3
Gourieroux and Jasiak (2025)	1.0	4.040	19.37	18.45	81.40	**2666**	**2534**	–	–	–	–	–	–
	1.2	0.698	**4.649**	**4.286**	6.326	**186.1**	**171.2**	46.11	**8597**	**7910**	–	–	–
	1.4	0.285	**2.150***	**1.898***	1.590	**37.05***	**32.65***	6.841	**683.2***	**601.9***	–	–	–
	1.6	0.170	1.197	0.995	0.787	**11.76**	**9.736**	2.597	**123.7**	**102.3**	**11.09**	**1299**	**1075**
	1.8	0.092	0.449	0.342	0.385	2.526	1.908	**1.060**	15.13	11.35	**3.584**	87.15	65.28
FlexZBoost	1.0	2.107	**16.39**	**15.59**	2481	6082	5831	–	–	–	–	–	–
	1.2	0.740	4.858	4.479	98.26	482.4	445.5	332.8	4.168e+04	3.835e+04	–	–	–
	1.4	0.292	2.381*	2.102*	10.03	75.54	66.71	87.52	2221	1957	–	–	–
	1.6	0.182	1.038	0.864	1.629	18.51	15.34	9.190	318.7	263.7	89.79	5745	4751
	1.8	0.179	0.256	0.225	0.963	1.965	1.604	7.424	17.58	14.05	93.36	200.1	162.0
Mixture Density Network	1.0	**0.902**	**7.461***	**7.096***	**37.73**	**1091***	**1037***	–	–	–	–	–	–
	1.2	**0.305**	**1.705***	**1.574***	**2.980**	**67.51***	**62.12***	**38.00**	**2614***	**2405***	–	–	–
	1.4	**0.124**	**0.640***	**0.567***	**0.747**	**8.409**	**7.416**	**4.118**	**133.6**	**117.7**	–	–	–
	1.6	**0.082**	**0.306***	**0.258***	**0.427**	**2.745***	**2.283***	**1.932**	**31.10***	**25.75***	24.66	**390.1***	**323.0***
	1.8	**0.056**	**0.113***	**0.093***	0.421	**0.628***	**0.547***	1.624	**4.679***	**3.664***	15.36	**26.48***	**22.29***

Notes: This Table reports the root mean squared error (RMSE) of estimated predictive moments relative to theoretical values. α denotes the tail index of the stable distribution. Predictive moments are evaluated over three spatial regions: Center $[q_{0.1}, q_{0.9}]$, Tails $[q_{0.01}, q_{0.1}] \cup [q_{0.9}, q_{0.99}]$, and Total $[q_{0.01}, q_{0.99}]$, where q_p represents the p-th quantile. Best method in red, second best in **bold black**. An asterisk (*) designates model(s) which belong to the Model Confidence Set at the 90% confidence level.

The statistical significance of these differences in performance is confirmed by the MCS tests. In most cases, only one forecasting approach belongs to the $MCS_{90\%}$: Nadaraya-Watson for the central region, and the MDN for the tail region and overall distribution.[14]

[13]Notable exceptions occur at $\alpha = 1.8$: for MAR(0,2) and MAR(1,1), Nadaraya-Watson performs better for higher-order moments ($k \geq 2$), while for MARMA(1,1,1,1), Nadaraya-Watson outperforms only for the fourth moment.

Table 3: Root Mean Squared Error of Predictive Moments: MAR(0, 2) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.244***	20.23	19.23	**2.252***	**1697**	**1613**	–	–	–	–	–	–
	1.2	**0.114***	5.137	4.726	**0.610***	**162.9**	**149.9**	**5.464***	**6199**	**5703**	–	–	–
	1.4	**0.086***	**1.568**	**1.382**	**0.274***	**21.01**	**18.51**	**0.787***	**339.5**	**299.1**	–	–	–
	1.6	**0.044***	**0.583**	**0.483**	**0.118***	4.763	3.940	**0.347***	**38.99***	**32.25***	**1.540***	**368.0***	**304.4***
	1.8	**0.029***	**0.188***	**0.142***	**0.085***	**0.962***	**0.722***	**0.159***	**4.597***	**3.443***	**0.578***	**23.65***	**17.71***
FlexZBoost	1.0	1.164	**9.557**	**9.090**	617.9	1886	1803	–	–	–	–	–	–
	1.2	0.455	**4.648**	**4.280**	29.97	226.7	208.9	59.39	1.322e+04	1.216e+04	–	–	–
	1.4	0.265	1.789	1.581	3.475	39.33	34.68	19.30	1070	942.4	–	–	–
	1.6	0.200	0.695	0.586	1.345	**4.626**	**3.900**	12.98	47.36	39.85	226.6	502.1	434.4
	1.8	0.187	0.267	0.235	0.703	1.092	0.941	3.668	9.215	7.315	32.50	72.83	58.63
Mixture Density Network	1.0	**0.733**	**5.713***	**5.435***	**21.11**	**291.6***	**277.2***	–	–	–	–	–	–
	1.2	**0.129***	**2.285***	**2.103***	**1.647**	**41.24***	**37.94***	**41.53**	**2067***	**1902***	–	–	–
	1.4	**0.084***	**0.607***	**0.536***	**0.773**	**8.158***	**7.196***	**2.672**	**161.3***	**142.1***	–	–	–
	1.6	**0.084**	**0.276***	**0.233***	**0.575**	**2.696***	**2.253***	**2.873**	**28.51***	**23.63***	**30.92**	**309.6***	**256.6***
	1.8	**0.049**	**0.168***	**0.130***	**0.345**	**1.032***	**0.806**	**1.053**	**6.553**	**4.955**	**8.499**	**59.19**	**44.66**

Notes: For details on variable definitions and methodology, refer to Table 2.

Table 4: Root Mean Squared Error of Predictive Moments: MAR(1, 1) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.506***	40.06	38.07	**13.94***	**7557**	**7182**	–	–	–	–	–	–
	1.2	**0.104***	6.755	6.215	**1.469***	**378.8**	**348.5**	**16.37***	**2.040e+04**	**1.877e+04**	–	–	–
	1.4	**0.083***	2.312	2.037	**0.566***	**59.61**	**52.51**	**4.169***	**1513**	**1333**	–	–	–
	1.6	**0.047***	**0.798**	**0.660**	**0.199***	**12.24**	**10.12**	**0.688***	**170.4**	**140.9**	**5.156***	**2312**	**1912**
	1.8	**0.040***	**0.159***	**0.122***	**0.113***	**1.335***	**1.002***	**0.428***	**9.769***	**7.319***	**2.135***	**70.68***	**52.93***
FlexZBoost	1.0	2.320	**17.85**	**16.98**	3065	7661	7343	–	–	–	–	–	–
	1.2	0.791	**5.266**	**4.855**	120.7	616.9	569.5	462.4	5.894e+04	5.423e+04	–	–	–
	1.4	0.349	**2.174**	**1.922**	12.40	84.16	74.37	120.5	2932	2583	–	–	–
	1.6	0.191	1.306	1.086	1.996	27.51	22.78	12.41	529.1	437.6	119.7	1.049e+04	8678
	1.8	0.194	0.353	0.294	1.015	3.798	2.922	8.195	38.76	29.52	108.6	424.0	325.5
Mixture Density Network	1.0	**0.967**	**3.751***	**3.578***	**47.44**	**863.9***	**821.2***	–	–	–	–	–	–
	1.2	**0.328**	**1.789***	**1.651***	**4.245**	**86.04***	**79.17***	**62.46**	**3998***	**3678***	–	–	–
	1.4	**0.094***	**1.021***	**0.901***	**0.844**	**19.21***	**16.92***	**4.898**	**367.2***	**323.5***	–	–	–
	1.6	**0.061***	**0.302***	**0.252***	**0.479**	**2.472***	**2.062***	**2.441**	**23.10***	**19.16***	**31.12**	**300.8***	**249.4***
	1.8	**0.067**	**0.154***	**0.123***	**0.467**	**1.367***	**1.069***	**1.927**	**12.52***	**9.462***	**18.88**	**118.4**	**89.51**

Notes: For details on variable definitions and methodology, refer to Table 2.

Table 5: Root Mean Squared Error of Predictive Moments: MARMA(1, 1, 1, 1) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.711***	41.50	39.44	**27.39***	**3236**	**3076**	–	–	–	–	–	–
	1.2	**0.134***	6.808	6.264	**1.149***	**194.5**	**178.9**	**26.24***	**8876**	**8166**	–	–	–
	1.4	**0.082***	2.129	1.876	**0.383***	**42.68**	**37.59**	**3.275***	**1044**	**919.7**	–	–	–
	1.6	**0.056***	**0.478***	**0.396***	**0.127***	**4.996**	**4.132**	**0.984***	**46.74**	**38.66***	**10.97***	**631.4**	**522.2**
	1.8	**0.042***	**0.290**	**0.219***	**0.116***	**1.566**	**1.175***	**0.207***	**9.357***	**7.006***	**1.216***	**48.67***	**36.44***
FlexZBoost	1.0	1.679	**20.73**	**19.71**	1770	6375	6083	–	–	–	–	–	–
	1.2	0.592	**4.634**	**4.269**	66.67	344.6	318.1	185.0	4.014e+04	3.693e+04	–	–	–
	1.4	0.296	**1.736**	**1.536**	6.192	68.48	60.39	40.52	2015	1775	–	–	–
	1.6	0.201	0.828	0.694	1.488	19.57	16.21	11.53	277.7	229.8	185.3	8553	7074
	1.8	0.196	0.402	0.328	0.693	4.325	3.270	4.234	38.92	29.27	43.93	426.0	320.2
Mixture Density Network	1.0	**0.689***	**5.908***	**5.619***	**68.94**	**659.3***	**626.9***	–	–	–	–	–	–
	1.2	**0.261**	**2.209***	**2.035***	**1.493**	**23.46***	**21.59***	**42.15**	**1805***	**1661***	–	–	–
	1.4	**0.173**	**0.925***	**0.819***	**1.186**	**4.344***	**3.868***	**13.60**	**180.8***	**159.4***	–	–	–
	1.6	**0.086**	**0.327***	**0.274***	**0.633**	**0.972***	**0.879***	**3.186**	**26.41***	**21.91***	**41.99**	**180.3***	**151.0***
	1.8	**0.089**	**0.132***	**0.115***	**0.576**	**0.902***	**0.776***	**2.915**	**6.460***	**5.208***	**22.14**	**52.50***	**41.95**

Notes: For details on variable definitions and methodology, refer to Table 2.

[14]We also test the sensitivity of our findings to the choice of the forecast horizon ($h = 2$ and $h = 5$) both in terms of RMSE and MCS test. The results, available in Section 3 of the Online Appendix, are qualitatively similar.

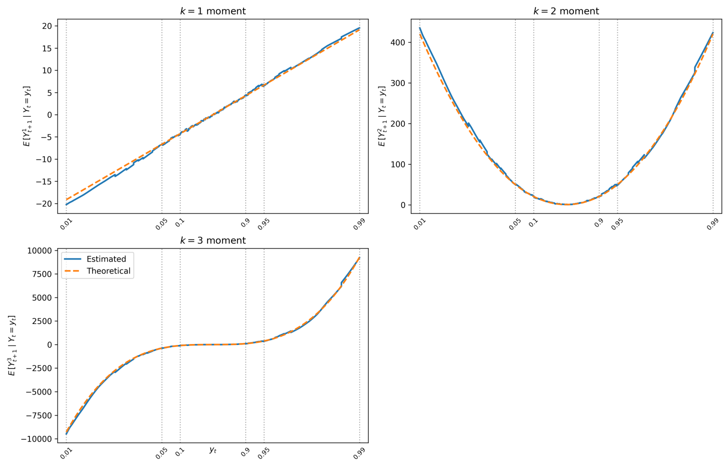
Figure 3 provides a visual comparison of the one-step-ahead predictive conditional moments for a MAR(0,1) process with tail-index $\alpha = 1.4$ across the five competing approaches. The MDN produces smooth predictions that closely track the theoretical moments throughout the distribution. Gourieroux & Jasiak (2025)'s forecasts are also smooth but increasingly drift from the truth in the tails. FlexZBoost exhibits a staircase-like pattern with systematic deviations. The moment forecasts from Nadaraya-Watson and Lanne et al. (2012) become increasingly noisy as we move away from the center. These patterns are qualitatively similar, though noisier, at longer horizons $h = 2$ and $h = 5$ (see the Online Appendix).

3.4.3 Case 3: Comparison with the Realized Outcomes

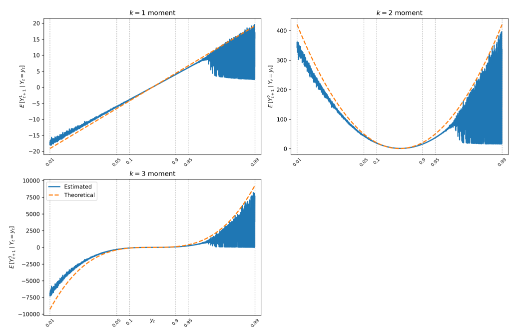
Rather than evaluating predictive densities at pre-specified conditioning values, we now train each method on 5,000 observations from a MAR(0,1) process and use the trained models to forecast the subsequent 500 realizations. This approach allows us to assess density forecast quality using proper scoring rules that directly compare the predictive density, $\hat{p}_h(X_{t+h} \mid X_t = x_i)$, against the realized outcome X_{t+h}.

Table 6 presents detailed results for forecast horizons $h \in \{1, 2, 5\}$ using four complementary evaluation metrics: the Conditional Density Estimation (CDE) loss, the Continuous Ranked Probability Score (CRPS), the logarithmic probability score, and the quantile score at the 10% level (see the Online Appendix for formal definitions).[15] As in our previous evaluation strategy, we assess the performance across three distributional regions (Center, Tails, and Total) to capture method-specific strengths. The results corroborate our grid-based findings. At the one-step-ahead horizon, the MDN achieves the best or second-best performance across nearly all metrics and tail index values, and it is particularly excelling

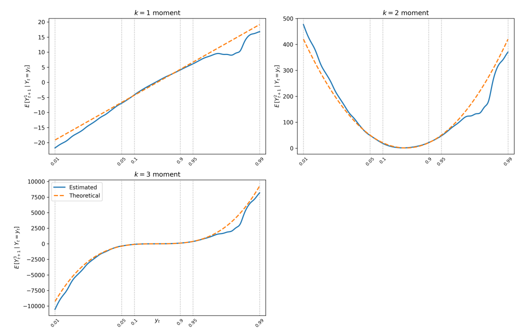
[15] For CDE loss, CRPS, and quantile scores, lower values indicate superior performance, while for the log probability score, higher values are preferred.



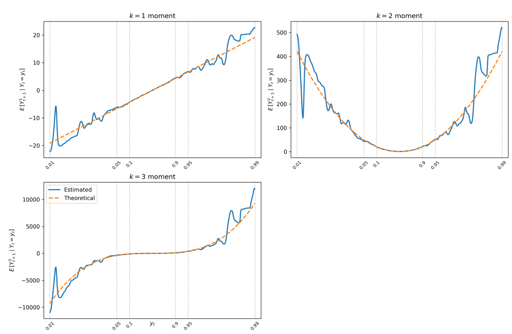
(a) Mixture Density Network



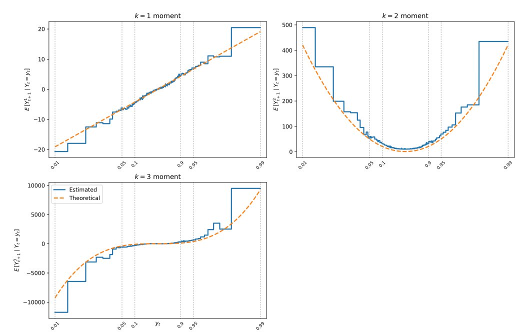
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



(d) Nadaraya-Watson



(e) FlexZBoost

Figure 3: Conditional Predictive Moment Accuracy: MAR(0,1) Process, 1-Step-Ahead Forecasts

Notes: This figure displays the estimated predictive moments $\mathbb{E}[X_{t+h}^k \mid X_t]$ for $k \in \{1, 2, 3\}$ as a function of the conditioning variable X_t for a purely noncausal MAR(0,1) process with α-stable innovations. Each panel from (a) to (e) shows the results for a specific density forecasting method. Blue curves represent estimated moments, while orange curves show the corresponding theoretical ones. The tail-index is fixed at $\alpha = 1.4$.

in the tail region. As the forecast horizon extends to $h = \{2, 5\}$, the relative performance rankings remain remarkably stable. The MDN continues to dominate in the tail region and overall distribution for most α values, though Nadaraya-Watson exhibits competitive performance for $\alpha = 1.8$, in particular at longer horizons. Note also that contrary to the

Table 6: Density Forecast Performance Metrics: MAR(0, 1) Process, 1-Step-Ahead Forecasts

Model	α	CDE Loss			CRPS			Log Prob			QS 10%		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	-0.235	-0.076	-0.221	1.217	3.716	2.041	-2.354	-4.824	-2.822	0.509	1.964	0.889
Lanne et al. (2012)	1.0	0.161	0.080	0.135	3.499	3.811	3.794	-3.926	-3.882	-4.207	0.702	0.756	0.841
Gourieroux and Jasiak (2025)	1.0	0.144	0.192	0.172	3.709	4.312	4.402	-4.046	-4.976	-4.450	1.522	1.916	1.779
FlexZBoost	1.0	-0.034	-0.014	-0.011	7.363	12.13	10.71	-3.452	-4.511	-4.785	1.167	5.739	5.476
Mixture Density Network	1.0	-0.098	-0.087	-0.088	1.584	2.817	2.281	-2.568	-2.887	-2.773	0.693	0.965	0.942
Nadaraya-Watson	1.2	-0.308	-0.342	-0.299	0.690	1.126	0.993	-1.750	-2.158	-1.927	0.264	0.296	0.299
Lanne et al. (2012)	1.2	-0.129	-0.130	-0.114	0.834	1.435	1.128	-2.029	-3.273	-2.403	0.272	0.432	0.342
Gourieroux and Jasiak (2025)	1.2	-0.029	-0.085	-0.015	0.988	1.325	1.296	-2.236	-2.307	-2.429	0.297	0.309	0.335
FlexZBoost	1.2	-0.081	-0.076	-0.051	1.685	2.896	2.440	-2.585	-3.014	-3.228	0.552	1.132	1.141
Mixture Density Network	1.2	-0.302	-0.348	-0.290	0.682	1.108	0.996	-1.580	-1.600	-1.710	0.258	0.288	0.334
Nadaraya-Watson	1.4	-0.350	-0.486	-0.327	0.511	0.585	0.703	-1.397	-1.151	-1.582	0.161	0.167	0.201
Lanne et al. (2012)	1.4	-0.288	-0.459	-0.258	0.529	0.752	0.721	-1.418	-1.663	-1.765	0.166	0.266	0.223
Gourieroux and Jasiak (2025)	1.4	-0.237	-0.408	-0.207	0.583	0.587	0.770	-1.503	-1.329	-1.750	0.175	0.173	0.210
FlexZBoost	1.4	-0.184	-0.237	-0.149	0.795	0.927	0.965	-1.982	-1.846	-2.266	0.279	0.325	0.390
Mixture Density Network	1.4	-0.355	-0.542	-0.332	0.511	0.585	0.684	-1.266	-1.037	-1.450	0.160	0.166	0.202
Nadaraya-Watson	1.6	-0.398	-0.608	-0.360	0.432	0.350	0.514	-1.149	-0.796	-1.317	0.127	0.104	0.152
Lanne et al. (2012)	1.6	-0.388	-0.444	-0.301	0.427	0.420	0.540	-1.156	-1.506	-1.522	0.128	0.121	0.162
Gourieroux and Jasiak (2025)	1.6	-0.306	-0.513	-0.280	0.467	0.386	0.569	-1.244	-0.940	-1.470	0.138	0.105	0.163
FlexZBoost	1.6	-0.354	-0.492	-0.298	0.452	0.393	0.558	-1.542	-0.944	-1.673	0.145	0.125	0.182
Mixture Density Network	1.6	-0.410	-0.578	-0.356	0.420	0.359	0.529	-1.083	-0.834	-1.288	0.125	0.105	0.159
Nadaraya-Watson	1.8	-0.455	-0.716	-0.386	0.371	0.262	0.437	-0.971	-0.535	-1.164	0.109	0.078	0.130
Lanne et al. (2012)	1.8	-0.416	-0.411	-0.327	0.370	0.295	0.440	-1.005	-1.044	-1.357	0.110	0.080	0.131
Gourieroux and Jasiak (2025)	1.8	-0.415	-0.706	-0.357	0.384	0.264	0.450	-1.037	-0.539	-1.240	0.111	0.077	0.136
FlexZBoost	1.8	-0.434	-0.657	-0.378	0.381	0.281	0.447	-1.588	-1.336	-1.787	0.109	0.089	0.131
Mixture Density Network	1.8	-0.460	-0.687	-0.382	0.368	0.263	0.434	-0.943	-0.571	-1.152	0.107	0.075	0.128

Notes: This table reports density forecast performance metrics across different tail index values (α). CDE Loss (Conditional Density Estimation loss), CRPS (Continuous Ranked Probability Score), and QS 10% (Quantile Score at 10% level) are loss functions where lower values indicate better performance. Log Prob (Log Probability Score) is a scoring rule where higher values indicate better performance. Metrics are evaluated over three spatial regions: Center $[q_{0.1}, q_{0.9}]$, Tails $[q_{0.01}, q_{0.1}] \cup [q_{0.9}, q_{0.99}]$, and Total $[q_{0.01}, q_{0.99}]$, where q_p represents the p-th quantile. Best method in red, second best in **bold black**.

previous two setups, this evaluation framework has the advantage of allowing one to compare all the alternative forecasting approaches for all MAR models and at all forecast horizons.

As a final remark, comparing Tables 1 and 6 for the $\alpha = 1.0$ case reveals a notable discrepancy. When evaluated against the true predictive density, the MDN exhibits clear dominance, achieving KL divergence and ISE values substantially lower than all competitors both in the tail and over the total regions. However, when assessed using proxy metrics that do not require the true density, although the MDN remains among the best-performing methods, its superiority is less apparent. This suggests that standard scoring rules may lack sufficient discriminatory power to identify the best-performing forecasting method when predictive distributions exhibit heavy tails and bimodality.

3.5 Runtime Analysis

Finally, Table 7 reports the average computational time (over all tail indices α) required by each method to generate the forecasts for a MAR(0,1) process. For each forecast horizon, the reported runtime corresponds to the average time needed to compute the 5,000 conditional predictive densities over the grid of conditioning values $\{x_1, \ldots, x_{5000}\}$. FlexZBoost achieves the shortest runtime (under 5 seconds), though at the cost of inferior forecast accuracy, as demonstrated above. The MDN requires only 2-3 minutes. In stark contrast, the simulation-based methods of Lanne et al. (2012) and Gourieroux & Jasiak (2025) require substantially longer computation times, with the latter exhibiting a dramatic increase from about 1 hour at $h = 1$ to over 5 hours at $h = 5$.

Table 7: Average running time (in minutes)

Model	Horizon 1	Horizon 2	Horizon 5
Nadaraya-Watson	9.38	9.34	10.72
Gourieroux & Jasiak (2025)	59.03	117.85	300.54
Lanne et al. (2012)	106.33	107.37	107.69
FlexZBoost	0.04	0.05	0.05
Mixture Density Network	2.64	2.20	1.92

4 Empirical Applications

4.1 Forecasting Natural Gas Prices in Real Time

Having established the MDN's superior performance in controlled settings where the true DGP is known, we now evaluate its forecasting ability on real-world data where model specification uncertainty and data complexities are unavoidable. Natural gas prices are notoriously difficult to forecast due to periodic episodes of locally explosive behavior. These characteristics make the natural gas market a compelling testbed for our MDN approach.

We adopt the real-time forecasting framework of Baumeister et al. (2025), who provide an extensive evaluation of point forecasting methods for the real Henry Hub spot price, the benchmark for North American natural gas markets. Their setup accounts for publication lags and data revisions through a database of monthly vintages, ensuring that forecasts rely only on information available at each forecast origin. The estimation period runs from January 1976 to January 1997, and the out-of-sample evaluation period spans February 1997 to February 2024. We use forecast horizons h ranging from 1 to 24 months. We defer the reader to Baumeister et al. (2025) for a detailed description of the data construction and real-time constraints.

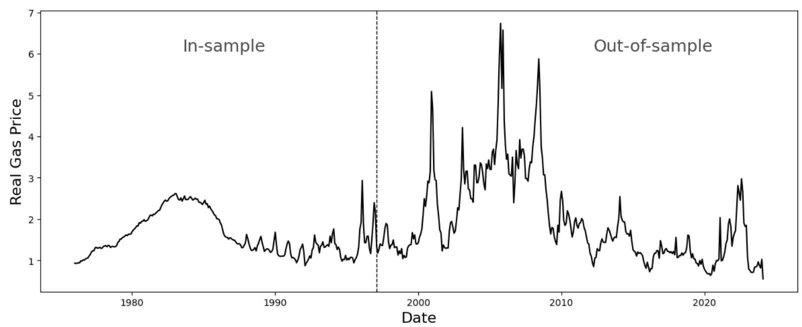


Figure 4: Real Henry Hub spot price of natural gas.

Figure 4 displays the evolution of the real Henry Hub spot price over the sample period. The series exhibits clear episodes of locally explosive behavior, most notably during the 2005-2008 period, characterized by sharp run-ups followed by abrupt collapses, precisely the type of dynamics that noncausal processes are designed to capture. Using the GCov estimation method (Gourieroux & Jasiak 2023) on the in-sample period (1976M1-1997M1), we estimate all possible MAR(p, q) specifications such that $p + q \leq k$, where k is chosen based on the Partial Autocorrelation Function. Then we select the model yielding i.i.d. residuals based on the portmanteau test of Jasiak & Neyazi (2025). The best specification is a purely noncausal MAR(0,1) process. The α-stable distribution parameters and their respective standard deviations are then obtained by fitting the characteristic function-based estimator of Nolan (2020) to the filtered residuals.[16] Besides, the MAR(0,1) dynamics of the noncausal process is found to exhibit temporal robustness, with comparable estimates

[16]The standard deviations of the MAR parameters are obtained by following Gourieroux & Jasiak (2023).

for the full post-revised series (1976M1–2024M2). As reported in Table 8, both samples reveal strong anticipative persistence ($\hat{\psi} \approx 0.95$) and heavy tails ($\hat{\alpha} \approx 1.8$), substantially thicker than in the Gaussian case ($\alpha = 2$), and consistent with the extreme price movements observed in the data.

Table 8: Estimated MAR(0,1) parameters for the real Henry Hub spot price

Parameter	Real-Time In-Sample	Full Period Post-Revised
ψ	0.957*** (0.018)	0.945*** (0.015)
α	1.779*** (0.186)	1.830*** (0.125)
β	0.415** (0.200)	0.628*** (0.149)
σ	0.070*** (0.006)	0.158*** (0.009)

Note: Standard errors in parentheses. *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

To evaluate the density forecasting performance of our MDN approach in this real-time setting, we simulate a trajectory of 5,000 observations from the estimated MAR(0,1) process on which each method is fitted, and generate out-of-sample forecasts over the period 1997M2–2024M2 for horizons $h \in \{1, 3, 6, 9, 12, 15, 18, 21, 24\}$ months. We assess the quality of the predictive densities with the four complementary metrics used in simulations (CDE loss, CRPS, log-probability score, and quantile score).

Table 9 summarizes the results. The MDN approach delivers the best or near-best performance across the majority of horizons and metrics. At the shortest horizon ($h = 1$), the MDN achieves the best CDE loss and CRPS, while FlexZBoost achieves a slightly lower quantile score. The MDN dominates in terms of log-probability score across all horizons. From $h = 3$ onward, the MDN consistently achieves the best performance across all four metrics, with only occasional exceptions at intermediate horizons ($h = 12, 15$) where Nadaraya-Watson performs marginally better in terms of CDE loss and CRPS. Importantly, when the MDN is not the best performer for a given metric, it consistently ranks as the second-best method.

A natural question is whether the MDN's well-calibrated predictive distributions also translate into accurate point predictions. Table 10 addresses this by comparing the RMSE

Table 9: Density forecast comparison

Horizon	Metric	Nadaraya-Watson	Lanne et al. (2012)	Gourieroux and Jasiak (2025)	FlexZBoost	Mixture Density Network
$h = 1$	CDE loss	-0.990	8.105	-0.631	**-1.130**	**-1.167**
	CRPS	0.189	0.212	0.195	**0.177**	**0.175**
	Log probability score	**-0.453**	-8.463	-0.710	-1.165	**-0.214**
	Quantile score (10%)	**0.062**	0.078	0.093	**0.059**	0.063
$h = 3$	CDE loss	**-0.660**	7.813	0.881	-0.637	**-0.701**
	CRPS	0.311	0.395	0.392	**0.297**	**0.291**
	Log probability score	**-1.024**	-8.839	-2.375	-3.573	**-0.713**
	Quantile score (10%)	0.099	0.174	0.235	**0.090**	**0.081**
$h = 6$	CDE loss	**-0.506**	9.300	1.530	-0.469	**-0.542**
	CRPS	0.432	0.613	0.628	**0.410**	**0.393**
	Log probability score	**-1.416**	-9.429	-3.645	-3.904	**-1.029**
	Quantile score (10%)	0.144	0.217	0.426	**0.120**	**0.118**
$h = 9$	CDE loss	**-0.446**	9.128	1.615	-0.383	**-0.459**
	CRPS	0.468	0.790	0.781	**0.453**	**0.435**
	Log probability score	**-1.440**	-10.90	-4.083	-3.906	**-1.102**
	Quantile score (10%)	**0.137**	0.254	0.564	0.145	**0.116**
$h = 12$	CDE loss	**-0.404**	9.744	2.062	-0.350	**-0.402**
	CRPS	**0.474**	0.950	0.960	**0.474**	0.478
	Log probability score	**-1.519**	-11.17	-4.767	-3.625	**-1.232**
	Quantile score (10%)	**0.119**	0.234	0.724	0.128	**0.117**
$h = 15$	CDE loss	**-0.354**	10.14	2.195	-0.283	**-0.348**
	CRPS	**0.489**	1.067	1.181	0.511	**0.510**
	Log probability score	**-1.669**	-11.32	-5.064	-4.285	**-1.327**
	Quantile score (10%)	**0.120**	0.223	0.937	0.134	**0.117**
$h = 18$	CDE loss	**-0.332**	11.06	2.242	-0.244	**-0.349**
	CRPS	**0.514**	1.165	1.318	0.533	**0.515**
	Log probability score	**-1.766**	-12.14	-5.149	-4.773	**-1.350**
	Quantile score (10%)	**0.122**	0.227	1.043	0.136	**0.114**
$h = 21$	CDE loss	**-0.341**	9.754	2.085	-0.273	**-0.356**
	CRPS	**0.524**	1.255	1.359	0.552	**0.516**
	Log probability score	**-1.748**	-11.67	-5.026	-4.682	**-1.339**
	Quantile score (10%)	**0.120**	0.230	1.005	0.134	**0.115**
$h = 24$	CDE loss	**-0.336**	9.564	2.143	-0.245	**-0.336**
	CRPS	**0.532**	1.342	1.408	0.564	**0.524**
	Log probability score	**-1.780**	-12.65	-5.280	-4.307	**-1.359**
	Quantile score (10%)	**0.117**	0.231	0.986	0.133	**0.109**

Note: Best method in red, second best in **bold black**.

of the MDN's point forecasts, computed as the median of the predictive density, against the forecasts of a comprehensive set of models evaluated by Baumeister et al. (2025), expressed as ratios relative to the no-change forecast ($\hat{X}_{t+h} = X_t$). The comparison encompasses both univariate specifications (AR(1), AR(AIC), exponential smoothing) and multivariate Bayesian VAR models that incorporate additional predictors. Focusing first on the univariate benchmarks, the MDN delivers the lowest RMSE at all horizons from $h = 1$ to $h = 9$, outperforming AR and exponential smoothing specifications. At the 9-month-ahead horizon, the MDN achieves an RMSE ratio of 0.864, representing a 14% improvement over the random walk and substantially better than all univariate alternatives. Statistical significance against the no-change forecast is confirmed by Diebold-Mariano tests (Diebold & Mariano 1995), see Online Appendix. At longer horizons ($h \geq 15$), exponential smoothing takes the lead, although the MDN remains competitive, consistently ranking second-best among univariate methods.

Remarkably, the MDN also outperforms the multivariate BVAR models at short- to medium-term horizons. The BVAR specifications in Baumeister et al. (2025) exploit up to six predictor variables, yet the MDN achieves lower RMSE ratios up to horizon $h = 9$. Moreover, unlike Baumeister et al. (2025), our approach does not require recursive re-estimation of the model at each forecast origin, making it computationally more efficient.

Table 10: Average RMSE Ratios Relative to the No-Change Forecast of the Real Natural Gas Spot Price

	Univariate Models				Multivariate Models	
Horizon	AR(1)	AR(AIC)	Exp. Smoothing	MDN	BVAR(AIC)	BVAR(1)
1	**0.993**	1.021	1.456	0.963	1.003	0.984
3	**0.978**	0.989	1.121	0.937	0.986	0.962
6	0.949	**0.944**	0.970	0.894	0.948	0.923
9	0.925	0.921	**0.905**	0.864	0.909	0.897
12	0.915	0.909	**0.893**	0.882	0.900	0.875
15	0.914	0.908	0.896	**0.899**	0.909	0.876
18	0.918	0.911	**0.899**	0.897	0.923	0.886
21	0.924	0.914	0.890	**0.901**	0.925	0.905
24	0.933	0.919	0.884	**0.908**	0.940	0.922

Note: Values below 1 indicate improvements relative to the no-change forecast. Best univariate method in red, second best in **bold black**. Benchmark results from Baumeister et al. (2025), Table 3. We report RMSE ratios by taking the square root of the Mean Squared Prediction Error (MSPE) measures in Baumeister et al. (2025), to ensure consistency throughout the paper.

The excellent point forecast performance of the MDN can be partly attributed to the noncausal specification itself. By explicitly modeling the anticipative dynamics, the MAR(0,1) process captures the locally explosive behavior of natural gas prices, a feature that the purely causal models employed by Baumeister et al. (2025) cannot accommodate. The gains are concentrated at short- to medium-term horizons (1–9 months), where the nonlinear dynamics inherent to noncausal processes exert their strongest influence and prove particularly effective at capturing the explosive episodes observed in natural gas prices. At longer horizons, mean-reverting forces dominate the predictive signal, diminishing the comparative advantage of our approach, although the MDN remains highly competitive.

Furthermore, the MDN outperforms the alternative density forecasting methods in terms of RMSE. Additional results are reported in the Online Appendix.

4.2 Forecasting Inflation in Real Time

To further assess the point forecasting performance of our MDN approach, we apply it to U.S. inflation, a series known to exhibit noncausal dynamics (Lanne & Saikkonen 2011, Lanne et al. 2012), following the real-time framework of Medeiros et al. (2021). Importantly, unlike natural gas prices, inflation exhibits near-Gaussian tail index ($\hat{\alpha} \approx 1.9$), providing a valuable test of whether the MDN, designed to capture heavy-tailed distributions, retains its forecasting advantages when tail behavior is less pronounced. The model is estimated on the January 2001 vintage of FRED-MD dataset, which covers January 1960 to December 2000. Forecasts for the period January 2001 to December 2015 are then evaluated against the ex-post revised series from the January 2016 vintage (see Figure 5). This real-time setup mirrors our natural gas application and reflects the information actually available to forecasters at each point in time.

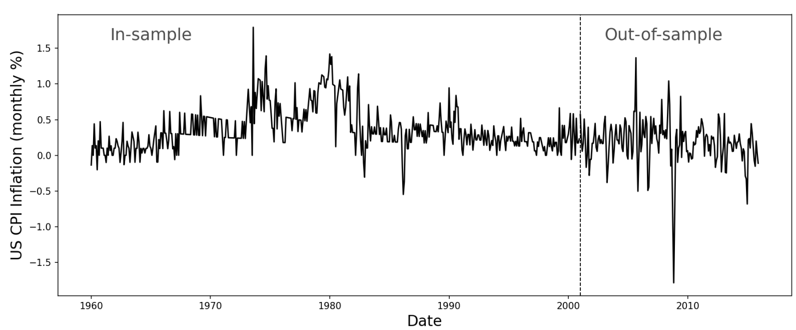


Figure 5: U.S. Inflation.

Using the same procedure as in Section 4, we estimate a purely noncausal MAR(0,2) on the in-sample period and train our MDN on a simulated trajectory of 5,000 observations (see the Online Appendix for estimation results). Table 11 reports the RMSE, mean absolute error (MAE), and median absolute deviation from the median (MAD) for the MDN and the two univariate benchmarks (AR and unobserved components stochastic volatility – UCSV – models) from Medeiros et al. (2021), relative to the no-change forecast.

Table 11: Real-Time Forecast Accuracy Ratios Relative to the No-Change Forecast for US Inflation

	Univariate Models									Multivariate Models					
	AR(BIC)			UCSV			MDN			RF			RR		
Horizon	RMSE	MAE	MAD	RMSE	MAE	MAD	RMSE	MAE	MAD	RMSE	MAE	MAD	RMSE	MAE	MAD
1	**0.91**	**0.87**	**0.80**	0.97	0.91	0.86	**0.86**	**0.85**	**0.77**	0.88	0.83	0.74	0.84	0.81	0.77
2	**0.81**	**0.80**	**0.78**	0.82	0.82	0.82	**0.79**	**0.77**	**0.67**	0.74	0.72	0.72	0.74	0.72	0.70
3	**0.77**	**0.73**	**0.66**	0.79	0.77	0.82	**0.76**	**0.75**	**0.72**	0.70	0.66	0.62	0.72	0.68	0.60
4	**0.78**	**0.75**	**0.80**	0.80	0.77	0.81	**0.77**	**0.74**	**0.63**	0.72	0.72	0.73	0.75	0.72	0.76
5	0.79	0.80	**0.76**	**0.78**	**0.79**	0.80	**0.75**	**0.75**	**0.68**	0.72	0.74	0.75	0.76	0.77	0.74
6	0.80	0.81	**0.76**	**0.79**	**0.80**	**0.76**	**0.76**	**0.76**	**0.66**	0.75	0.76	0.77	0.78	0.80	0.75
7	**0.80**	**0.79**	**0.73**	0.81	0.81	0.90	**0.76**	**0.75**	**0.55**	0.76	0.75	0.75	0.79	0.77	0.75
8	**0.77**	**0.75**	**0.74**	0.80	0.80	0.86	**0.76**	**0.73**	**0.55**	0.74	0.72	0.73	0.76	0.73	0.72
9	**0.79**	**0.77**	**0.81**	**0.79**	0.79	0.84	**0.76**	**0.75**	**0.62**	0.74	0.72	0.76	0.77	0.76	0.77
10	0.82	**0.80**	0.77	**0.81**	**0.80**	**0.75**	**0.80**	**0.78**	**0.63**	0.79	0.75	0.65	0.79	0.76	0.69
11	0.83	**0.84**	**0.83**	**0.81**	**0.83**	0.92	**0.83**	0.84	**0.62**	0.80	0.81	0.81	0.80	0.82	0.75
12	**0.77**	**0.78**	0.73	**0.77**	**0.78**	**0.70**	**0.76**	**0.75**	**0.58**	0.72	0.73	0.68	0.74	0.74	0.67

Note: Values below 1 indicate improvements relative to the no-change forecast. Best univariate method in red, second best in **bold black**. AR, UCSV, RF, and RR results from Medeiros et al. (2021), Table 6.

Focusing on univariate models, our MDN outperforms the benchmarks for all three criteria at most horizons considered (11 out of 12 horizons, with stable gains around 24% relative to the random walk). Again, statistical significance against the no-change forecast is confirmed by Diebold-Mariano tests in Online Appendix. Table 11 also reports results for the Random Forest (RF) and Ridge Regression (RR) models, identified as the best-performing multivariate methods in Medeiros et al. (2021). These machine learning approaches exploit the full FRED-MD database comprising over 120 macroeconomic variables with four lags each, resulting in approximately 500 potential predictors. Despite this vastly richer information set, the MDN remains competitive, with RMSE ratios within 0.01–0.06 of RF and RR at most horizons.

Overall, despite not being tailored to point forecasting, our approach ranks among the leading univariate methods for forecasting U.S. inflation and natural gas prices, and remains competitive with state-of-the-art multivariate methods that rely on considerably richer information sets.

5 Conclusion

Time series forecasting in the presence of locally explosive dynamics, marked by rapid expansions followed by sudden reversals, remains a core challenge in (macro-)financial econometrics. We address this issue by introducing a Mixture Density Network designed to capture the distinctive distributional features of noncausal processes. The resulting framework enables near-instantaneous density forecasting once trained, effectively overcoming the computational constraints associated with existing methods.

We evaluate our approach through extensive Monte Carlo simulations that cover a range of MARMA specifications. The results show that the MDN method consistently achieves the best performance in the tail region and over the full distribution at all forecast horizons. These findings are further validated through two empirical applications to real-time forecasting of natural gas prices and inflation.

References

Athey, S. & Imbens, G. W. (2019), 'Machine learning methods that economists should know about', Annual Review of Economics **11**(Volume 11, 2019), 685–725.

Avelino, J. G., Cavalcanti, G. D. C. & Cruz, R. M. O. (2024), 'Resampling strategies for imbalanced regression: A survey and empirical analysis', Artificial Intelligence Review **57**(4), 82.

Azzalini, A. & Dalla Valle, A. (1996), 'The multivariate skew-normal distribution', Biometrika **83**(4), 715–726.

Baumeister, C., Huber, F., Lee, T. K. & Ravazzolo, F. (2025), 'Forecasting natural gas prices in real time', Journal of Applied Econometrics .

Belkin, M., Hsu, D., Ma, S. & Mandal, S. (2019), 'Reconciling modern machine-learning practice and the classical bias–variance trade-off', Proceedings of the National Academy of Sciences **116**(32), 15849–15854.

Bishop, C. M. (1994), 'Mixture density networks'.

Blasques, F., Koopman, S. J., Mingoli, G. & Telg, S. (2025), 'A novel test for the presence of local explosive dynamics', Journal of Time Series Analysis .

Bruffaerts, C., Verardi, V. & Vermandele, C. (2014), 'A generalized boxplot for skewed and heavy-tailed distributions', Statistics & Probability Letters **95**, 110–117.

Cavaliere, G., Nielsen, H. B. & Rahbek, A. (2020), 'Bootstrapping noncausal autoregressions: With applications to explosive bubble modeling', Journal of Business & Economic Statistics **38**(1), 55–67.

Chen, T. & Guestrin, C. (2016), XGBoost: A scalable tree boosting system, in 'Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 785–794.

Dalmasso, N., Pospisil, T., Lee, A. B., Izbicki, R., Freeman, P. E. & Malz, A. I. (2020), 'Conditional density estimation tools in python and R with applications to photometric redshifts and likelihood-free cosmological inference', Astronomy and Computing **30**, 100362.

Davis, R. A. & Song, L. (2020), 'Noncausal vector AR processes with application to economic time series', Journal of Econometrics **216**(1), 246–267.

de Truchis, G., Fries, S. & Thomas, A. (2025), 'Forecasting extreme trajectories using seminorm representations'. Working paper.

de Truchis, G. & Thomas, A. (2025), Laurent series expansion for MA(∞) representation of mixed causal-noncausal autoregressive processes, LEDa Working Paper WP 2025-06, LEDa, Université Paris Dauphine, PSL.

Dey, B., Newman, J. A., Andrews, B. H., Izbicki, R., Lee, A. B., Zhao, D., Rau, M. M. & Malz, A. I. (2022), 'Re-calibrating photometric redshift probability distributions using feature-space regression', Fourth Workshop on Machine Learning and Physical Sciences (NeurIPS 2021) .

Diebold, F. X. & Mariano, R. S. (1995), 'Comparing predictive accuracy', Journal of Business & Economic Statistics **13**(3), 253–263.

Fries, S. (2022), 'Conditional moments of noncausal alpha-stable processes and the prediction of bubble crash odds', Journal of Business & Economic Statistics **40**(4), 1596–1616.

Fries, S. & Zakoian, J.-M. (2019), 'Mixed causal-noncausal AR processes and the modelling of explosive bubbles', Econometric Theory **35**(6), 1234–1270.

Giancaterini, F., Hecq, A., Jasiak, J. & Manafi Neyazi, A. (2025), 'Bubble detection with application to green bubbles: A noncausal approach'.

Goodfellow, I., Bengio, Y. & Courville, A. (2016), Deep Learning, MIT Press.

Gourieroux, C. & Jasiak, J. (2016), 'Filtering, prediction and simulation methods for noncausal processes', Journal of Time Series Analysis **37**(3), 405–430.

Gourieroux, C. & Jasiak, J. (2018), 'Misspecification of noncausal order in autoregressive processes', Journal of Econometrics **205**(1), 226–248.

Gourieroux, C. & Jasiak, J. (2023), 'Generalized covariance estimator', Journal of Business & Economic Statistics **41**(4), 1315–1327.

Gourieroux, C. & Jasiak, J. (2025), 'Nonlinear fore(back)casting and innovation filtering for causal-noncausal VAR models'.

Gourieroux, C., Jasiak, J. & Monfort, A. (2020), 'Stationary bubble equilibria in rational expectation models', Journal of Econometrics **218**(2), 714–735.

Gourieroux, C., Lu, Y. & Robert, C.-Y. (2025), 'The causal-noncausal tail processes'.

Gouriéroux, C. & Monfort, A. (2025), Affine feedforward stochastic (afs) neural network, TSE Working Paper 25-1666, Toulouse School of Economics (TSE).

Gourieroux, C. & Zakoian, J.-M. (2017), 'Local explosion modelling by non-causal process', Journal of the Royal Statistical Society Series B **79**(3), 737–756.

Guillaumin, A. P. & Efremova, N. (2024), 'Tukey g-and-h neural network regression for non-gaussian data'.

Hansen, P. R., Lunde, A. & Nason, J. M. (2011), 'The model confidence set', Econometrica **79**(2), 453–497.

He, K., Zhang, X., Ren, S. & Sun, J. (2015), Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in '2015 IEEE International Conference on Computer Vision (ICCV)', pp. 1026–1034.

Hecq, A. & Velasquez-Gaviria, D. (2025), 'Non-causal and non-invertible ARMA models: Identification, estimation and application in equity portfolios', Journal of Time Series Analysis **46**(2), 325–352.

Hecq, A. & Voisin, E. (2021), 'Forecasting bubbles with mixed causal-noncausal autoregressive models', Econometrics and Statistics **20**, 29–45.

Izbicki, R. (2017), 'Converting high-dimensional regression to high-dimensional conditional density estimation', Electronic Journal of Statistics **11**(2), 2800–2831.

Jasiak, J. & Neyazi, A. M. (2025), 'Gcov-based portmanteau test'.

Kingma, D. P. & Ba, J. (2015), Adam: A method for stochastic optimization, in 'International Conference on Learning Representations (ICLR)'.

Lanne, M., Luoto, J. & Saikkonen, P. (2012), 'Optimal forecasting of noncausal autoregressive time series', International Journal of Forecasting **28**(3), 623–631.

Lanne, M. & Saikkonen, P. (2011), 'Noncausal autoregressions for economic time series', Journal of Time Series Econometrics **3**(3).

Medeiros, M. C., Vasconcelos, G. F. R., Veiga, Á. & Zilberman, E. (2021), 'Forecasting inflation in a data-rich environment: The benefits of machine learning methods', Journal of Business & Economic Statistics **39**(1), 98–119.

Nolan, J. P. (2020), Univariate Stable Distributions: Models for Heavy Tailed Data, Springer Series in Operations Research and Financial Engineering, Springer, Cham.

Nyberg, H. & Saikkonen, P. (2014), 'Forecasting with a noncausal VAR model', Computational Statistics & Data Analysis **76**, 536–555.

Rosenblatt, M. (1969), Conditional probability density and regression estimators, in P. R. Krishnaiah, ed., 'Multivariate Analysis II', Academic Press, pp. 25–31.

Rosenblatt, M. (2000), Gaussian and Non-Gaussian Linear Time Series and Random Fields, Springer.

Rothfuss, J., Ferreira, F., Boehm, S., Walther, S., Ulrich, M., Asfour, T. & Krause, A. (2020), 'Noise regularization for conditional density estimation'.

Rothfuss, J., Ferreira, F., Walther, S. & Ulrich, M. (2019), 'Conditional density estimation with neural networks: Best practices and benchmarks'.

Saïdi, S. (2023), Noncausal Models: Unit Root Tests and Forecasting, PhD thesis, CY Cergy Paris University.

Steininger, M., Kobs, K., Davidson, P., Krause, A. & Hotho, A. (2021), 'Density-based weighting for imbalanced regression', Machine Learning **110**(9-10), 2187–2211.

Tail-Aware Density Forecasting of Locally Explosive Time Series: A Neural Network Approach

Online Appendix

Elena Dumitrescu
University Paris-Panthéon-Assas, CRED, 75006 Paris, France
and
Julien Peignon
CEREMADE, CNRS, UMR 7534, Université Paris-Dauphine
PSL University, Paris, France
and
Arthur Thomas
Université Paris-Dauphine, Université PSL
LEDa, CNRS, IRD, 75016 Paris, France

February 11, 2026

This Online Appendix contains supplementary material for the main paper. Section 1 describes the alternative forecasting methods used in our comparative analysis. Section 2 gathers the auxiliary theoretical results underlying the benchmark computations. Section 3 provides additional tables and figures for the Monte Carlo simulations, in particular the results for longer forecast horizons ($h = 2$ and $h = 5$), while Section 4 provides additional tables for the empirical applications.

1 Alternative Conditional Density Forecasting Methods

To assess the performance of our proposed MDN approach, we compare it against a diverse set of established conditional density forecasting methods. These benchmarks span different methodological paradigms: the nonparametric Nadaraya-Watson kernel density estimator (Nadaraya 1964, Watson 1964), the simulation-based approach of Lanne et al. (2012) and the closed-form predictive densities of Gourieroux & Jasiak (2025), both designed for mixed causal-noncausal AR processes, and the learning-based FlexZBoost method of Dalmasso et al. (2020).

1.1 Nadaraya-Watson

The Nadaraya-Watson kernel density estimator (Rosenblatt 1969) serves as our main benchmark method. This classical nonparametric approach employs kernel smoothing to construct continuous density estimates without imposing parametric assumptions on the underlying distribution. It provides a baseline for conditional density estimation through direct estimation of joint and marginal probability densities.

From a time series $(X_t)_{t=1}^T$ and a forecast horizon h, the conditional density is obtained through the Bayes' formula:

$$\hat{p}_h(X_{t+h}|X_t) = \frac{\hat{p}_h(X_t, X_{t+h})}{\hat{p}(X_t)},$$

where the joint and marginal densities are:

$$\hat{p}_h(X_t, X_{t+h}) = \frac{1}{T-h} \sum_{i=1}^{T-h} K_{b_{\text{joint}}} \left((X_t, X_{t+h}) - (X_i, X_{i+h}) \right),$$
$$\hat{p}(X_t) = \frac{1}{T-1} \sum_{i=1}^{T-1} K_{b_{\text{marginal}}} (X_t - X_i).$$

We employ a gaussian kernel, with bandwidths $(b_{\text{joint}}, b_{\text{marginal}})$ selected according to Silverman's rule.

1.2 Lanne et al. (2012)

The method proposed by Lanne et al. (2012) relies on a simulation-based approach for computing point and density forecasts in mixed causal-noncausal autoregressive processes. Given an observed series $(X_t)_{t=1}^T$, it predicts X_{T+h} by approximating the noncausal component through Monte Carlo simulations.

The forecast of X_{T+h} conditional on the observed data is computed recursively:

$$E_T(X_{T+h}) = \phi_1 E_T(X_{T+h-1}) + \cdots + \phi_r E_T(X_{T+h-r}) + E_T(v_{T+h}),$$

where $E_T(\cdot) = E[\cdot \mid X_1, \ldots, X_T]$ denotes the conditional expectation given all observations up to time T, and r is the number of lag coefficients in the causal part of the model. The noncausal component $v_{T+h} = \varphi(B^{-1})^{-1}\epsilon_{T+h}$ is approximated by:

$$v_{T+h} \approx \sum_{j=0}^{M-h} \beta_j \epsilon_{T+h+j},$$

with β_j derived from the power series expansion of $\varphi(B^{-1})^{-1}$ and M chosen sufficiently large. This approach entails model risk due to the necessity of specifying the innovation density g, as well as estimation risk since the causal and noncausal parameters $\phi_1, \dots, \phi_r$ and $\psi_1, \dots, \psi_s$ (from which the coefficients β_j are derived), together with the parameters of g, must all be estimated from the observed data.

The point forecast is computed via Monte Carlo simulations:

$$E_T\left(\sum_{j=0}^{M-h} \beta_j \epsilon_{T+h+j}\right) \approx \frac{\sum_{i=1}^{N}\left(\sum_{j=0}^{M-h} \beta_j \epsilon_{T+h+j}^{(i)} \prod_{k=1}^{s} g(e_{T-s+k}^{(i)};\lambda)\right)}{\sum_{i=1}^{N}\left(\prod_{k=1}^{s} g(e_{T-s+k}^{(i)};\lambda)\right)},$$

where N sequences of simulated innovations, $\{\epsilon_{T+1}^{(i)}, \dots, \epsilon_{T+M}^{(i)}\}_{i=1}^{N}$, are drawn from the error distribution. To derive the predictive density, they first obtain the empirical conditional cumulative distribution function (CDF) of X_{T+h}. For this, they replace the sum $\sum_{j=0}^{M-h} \beta_j \epsilon_{T+h+j}$ with the indicator function $\mathbf{1}(\sum_{j=0}^{M-h} \beta_j \epsilon_{T+h+j} \leq x)$ in the numerator of the expectation (in Eq. 1.2) for a predefined equispaced grid of values $x_1, \dots, x_K$. This grid spans the relevant range of the predicted variable, typically constructed from appropriate lower and upper quantiles of the empirical distribution, and serves as the discrete support on which the CDF is numerically evaluated. The predictive probability density function (PDF), $\hat{p}(X_{T+h}|X_1, \dots, X_T)$, is then immediately obtained by numerical differentiation of the empirical CDF.

While Lanne et al. (2012) originally developed their forecasting algorithm using Student's t-distributed innovations, the methodology extends to α-stable distributions. The framework requires the innovation density g to be absolutely continuous with respect to the Lebesgue measure, with a density that is positive for all $x \in \mathbb{R}$ and twice continuously differentiable. The α-stable distributions satisfy these regularity conditions: for any $\alpha \in (0, 2]$, they admit a smooth Lebesgue density $g_\alpha(x) > 0$ for all $x \in \mathbb{R}$ (see Samorodnitsky & Taqqu 1996, Nolan 2020). Moreover, efficient algorithms exist both for evaluating the α-stable density numerically and for simulating α-stable random variables, making the Monte Carlo procedure in Eq. (1.2) directly applicable. In our simulation study, we implement this α-stable variant.

1.3 Gourieroux & Jasiak (2025)

Gourieroux & Jasiak (2025) develop closed-form predictive densities for mixed causal-noncausal Vector Autoregressive (VAR) processes. Their method relies on the spectral decomposition of the autoregressive companion matrix to identify latent causal and noncausal state components. While the original framework is designed for multivariate processes, we apply it here in a univariate setting.

For a purely noncausal AR(1) process, the one-step-ahead predictive density given X_t admits the closed-form expression:

$$p(X_{t+1}|X_t) = \frac{1}{|\psi_1|} \frac{l(X_{t+1})}{l(X_t)} \, g\left(X_{t+1} - \frac{1}{\psi_1} X_t\right),$$

where l denotes the stationary marginal density of the process and g is the density of $\eta_t = -\frac{1}{\psi_1}\varepsilon_t$. This methodology involves both model risk, as it requires specifying the density g, and estimation risk, as ψ_1 and the parameters of g must be estimated from the data. In practice, the stationary density l is approximated via kernel density estimation. In our implementation, we employ a Gaussian kernel with bandwidth selected according to Silverman's rule. For longer horizon forecasts, the method proceeds iteratively: first compute the predictive density $p(X_{t+1}|X_t)$ and extract its mode, then forecast X_{t+2} conditional on X_{t+1} set to this mode, and repeat until the desired horizon h is reached.

1.4 FlexZBoost

FlexZBoost (Dalmasso et al. 2020) is a learning-based approach to conditional density forecasting, which makes it a direct competitor for our MDN approach. The method uses the FlexCode framework (Izbicki 2017) by employing gradient boosting as the underlying regression mechanism.[1] Its theoretical foundation rests on the representation of the conditional density through an orthonormal basis expansion:

$$p_h(X_{t+h}|\mathbf{X}_t) = \sum_{j=1}^{B} \beta_j(\mathbf{X}_t)\phi_j(X_{t+h}),$$

where $\{\phi_j(X_{t+h})\}_{j=1}^B$ constitutes an orthonormal basis system, and $\{\beta_j(\mathbf{x}_t)\}_{j=1}^B$ represent feature-dependent expansion coefficients. In practice, we follow Dalmasso et al. (2020) and employ a cosine basis with 31 basis functions. This formulation transforms the density estimation problem into B independent regression tasks, wherein the coefficients are obtained by regressing the orthogonal projections $\phi_j(X_{t+h})$ onto the predictor space $\mathbf{X}_t$.

2 Auxiliary Theoretical Results

This section gathers the theoretical foundations underlying the benchmark computations used throughout our Monte Carlo analysis. First, we present the methodology of Fries (2022) for computing predictive conditional moments of MARMA processes driven by α-stable innovations, which serves as the theoretical ground truth for evaluating forecasting performance. Second, we recall the closed-form expression for the predictive density of the

[1]More specifically, the XGBoost algorithm (Chen & Guestrin 2016) is used.

noncausal Cauchy AR(1) process derived by Gourieroux & Zakoian (2017), which provides a unique setting where the full conditional distribution, not just its moments, can be compared against estimated densities.

2.1 Computing Predictive Conditional Moments

This subsection summarizes the methodology of Fries (2022) for computing predictive conditional moments of MARMA processes with α-stable errors. Consider the two-sided moving average representation $X_t = \sum_{k\in\mathbb{Z}} a_k \varepsilon_{t+k}$, where the coefficients satisfy $\sum_{k\in\mathbb{Z}} |a_k|^s < \infty$ for some $s \in (0,\alpha) \cap [0,1]$, ensuring strict stationarity (Rosenblatt 2000). A key insight from Fries (2022) is that noncausal processes can admit finite conditional moments up to order $\lfloor 2\alpha + 1 \rfloor$ despite having infinite marginal variance, provided that the process is sufficiently anticipative. For $\alpha \in (0,2) \setminus \{1\}$ and $p \in \{1,2,3,4\}$ with $p < 2\alpha + 1$, the conditional moments $\mathbb{E}[X_{t+h}^p | X_t = x]$ exist and take the form:

$$\mathbb{E}[X_{t+h}|X_t = x] = \kappa_1 x + \frac{a(\lambda_1 - \beta_1\kappa_1)}{1 + a^2\beta_1^2} g(x), \tag{1}$$

$$\mathbb{E}[X_{t+h}^p|X_t = x] = \kappa_p x^p + \frac{a x^{p-1}(\lambda_p - \beta_1\kappa_p)}{1 + a^2\beta_1^2} g(x) - R_p(x), \quad p \in \{2,3,4\}, \tag{2}$$

where $a = \tan(\pi\alpha/2)$, the function $g(x)$ involves the marginal density $f_{X_t}(x)$ and integrals of the form

$$\mathcal{H}(n, \boldsymbol{\theta}; x) = \int_0^{+\infty} e^{-\sigma_1^\alpha u^\alpha} u^{n(\alpha-1)} \Big[\theta_1 \cos\Big(ux - a\beta_1\sigma_1^\alpha u^\alpha\Big) + \theta_2 \sin\Big(ux - a\beta_1\sigma_1^\alpha u^\alpha\Big)\Big]\, du, \tag{3}$$

and $R_p(x)$ denotes remainder terms involving $\mathcal{H}$ evaluated at orders $n = 2, \ldots, p$ with specific coefficient vectors $\boldsymbol{\theta}$ depending on $\alpha, \beta_1, \kappa_1, \ldots, \kappa_p, \lambda_1, \ldots, \lambda_p$.

The parameters appearing in (1)-(3) are defined as follows:

$$\sigma_1^\alpha = \sigma^\alpha \sum_{k\in\mathbb{Z}} |a_k|^\alpha, \qquad \beta_1 = \beta \frac{\sum_{k\in\mathbb{Z}} a_k^{\langle\alpha\rangle}}{\sum_{k\in\mathbb{Z}} |a_k|^\alpha},$$

$$\kappa_p = \frac{\sum_{k\in\mathbb{Z}} |a_k|^\alpha \left(\frac{a_{k-h}}{a_k}\right)^p}{\sum_{k\in\mathbb{Z}} |a_k|^\alpha}, \qquad \lambda_p = \beta \frac{\sum_{k\in\mathbb{Z}} a_k^{\langle\alpha\rangle} \left(\frac{a_{k-h}}{a_k}\right)^p}{\sum_{k\in\mathbb{Z}} |a_k|^\alpha},$$

where $y^{\langle\alpha\rangle} = \text{sign}(y)|y|^\alpha$ denotes the signed power function. The coefficients $(a_k)_{k\in\mathbb{Z}}$ of the MA(∞) representation can be recovered from the MARMA polynomial structure via partial fraction decomposition (see Fries 2022, Equation 3.7). For practical computation, these infinite sums are truncated, which yields accurate approximations since (a_k) decays geometrically for MARMA processes.

2.2 Closed-Form Predictive Density for the Noncausal Cauchy AR(1)

For the special case of a purely noncausal AR(1) process with Cauchy errors ($\alpha = 1$, $\beta = 0$), Gourieroux & Zakoian (2017) derive a closed-form expression for the causal predictive density. Consider the model

$$(1 - \psi_1 F)X_t = \varepsilon_t, \qquad \varepsilon_t \overset{\text{i.i.d.}}{\sim} \text{Cauchy}(0, \sigma),$$

where $|\psi_1| < 1$ ensures stationarity. The predictive density at horizon h is given by:

$$p(X_{t+h} \mid X_t) = \frac{1}{\pi\sigma_h} \cdot \frac{1}{1 + \left(\dfrac{X_t - \psi_1^h X_{t+h}}{\sigma_h}\right)^2} \cdot \frac{\sigma^2 + (1 - |\psi_1|)^2 X_t^2}{\sigma^2 + (1 - |\psi_1|)^2 X_{t+h}^2},$$

where the horizon-dependent scale parameter is

$$\sigma_h = \sigma \frac{1 - |\psi_1|^h}{1 - |\psi_1|}.$$

This closed-form expression enables direct comparison of estimated predictive densities against the true conditional distribution, providing a rigorous benchmark for density forecast evaluation that complements the moment-based comparisons available for general α-stable MARMA processes.

More generally, for any stable mixed-causal process (MAR) with a single anticipative root, the conditional distribution during extreme events follows a binomial law: the future trajectory can either continue its explosive path or collapse abruptly, with the probability of crashing at horizon h determined by the persistence parameter (see Gourieroux & Zakoian 2017, Fries 2022, de Truchis et al. 2025, Gourieroux et al. 2025). In the same vein, de Truchis et al. (2025) and Gourieroux et al. (2025) showed that the conditional distribution of the MAR model with at least two anticipative roots is 0,1-valued, *i.e.*, the conditional predictive density converges to a Dirac measure. This implies that, unlike the noncausal AR(1) where only crash probabilities can be predicted, the AR(2) specification allows for exact prediction of the bubble peak, the crash date, and the post-crash value. Properly accounting for these specificities of non-causal models is critical for ensuring accurate forecasts and strong predictive performance enspecially in the tails.

3 Additional Simulation results

In this Section, we include detailed results on the Model Confidence Set analysis summarized by an asterisk ($*$) in the tables of Section 3.4.2 in the main paper and then provide evidence of the robustness of our findings in Sections 3.3, 3.4.2 and 3.4.3 of the article to the choice of the forecasting horizon by setting $h = \{2, 5\}$.

3.1 Model Confidence Set

Table 1: Model Confidence Set Test Results: MAR(0, 1) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0002	0.0004	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	1.0000	0.0002	0.0002	–	–	–
Nadaraya-Watson	1.4	1.0000	1.0000	1.0000	1.0000	0.1312	0.1698	1.0000	0.0208	0.0228	–	–	–
	1.6	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	1.0000	0.0000	0.0000	1.0000	0.0000	0.0004
	1.8	1.0000	0.0792	0.4850	1.0000	0.0044	0.8030	1.0000	0.1902	1.0000	1.0000	0.2558	1.0000
	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0000	0.0000	–	–	–
Lanne et al. (2012)	1.4	0.0000	0.0990	0.1102	0.0000	0.0000	0.0012	0.0000	0.0000	0.0002	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002
	1.0	0.0000	0.0028	0.0018	0.0000	0.0330	0.0350	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0002	0.0002	–	–	–
Gourieroux and Jasiak (2025)	1.4	0.0000	0.6404	0.4834	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008	0.0000	0.0002	0.0018
	1.0	0.0000	0.0000	0.0002	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.6404	0.4834	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.0	0.0002	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0000	1.0000	1.0000	0.0004	1.0000	1.0000	0.0004	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.0000	0.2064	0.1670	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0698	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0006	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	0.9898	0.0000	1.0000	0.3868

Notes: This Table reports MCS p-values from the Model Confidence Set procedure of Hansen et al. (2011) applied at the 90% confidence level ($\alpha = 0.10$). Values represent the probability threshold at which each model enters the set of superior forecasting methods. A model belongs to the $\text{MCS}_{90\%}$ if its p-value ≥ 0.10. Values in red indicate models included in the MCS (i.e., models with statistically indistinguishable superior performance). The loss function is the squared error of predictive moments relative to theoretical ones.

Table 2: Model Confidence Set Test Results: MAR(0, 2) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0234	0.0296	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0006	0.0008	–	–	–
	1.6	1.0000	0.0000	0.0008	1.0000	0.0062	0.0502	1.0000	0.1858	0.3776	1.0000	1.0000	1.0000
	1.8	1.0000	0.1758	0.5138	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1.0	0.0000	0.0006	0.0018	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0008	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0020	0.0012	0.0000	0.0032	0.0020
	1.8	0.0000	0.0030	0.0000	0.0000	0.0962	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.0	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.1454	1.0000	1.0000	0.0000	1.0000	1.0000	0.0030	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.4572	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	0.2448	0.1272
	1.8	0.0016	1.0000	1.0000	0.0000	0.4470	0.0092	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000

Notes: For details on variable definitions and methodology, refer to Table 1.

Table 3: Model Confidence Set Test Results: MAR(1, 1) Process, 1-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.4	1.0000	0.0004	0.0006	1.0000	0.0000	0.0000	1.0000	0.0002	0.0000	–	–	–
	1.6	1.0000	0.0000	0.0006	1.0000	0.0000	0.0002	1.0000	0.0000	0.0000	1.0000	0.0000	0.0006
	1.8	1.0000	0.7840	1.0000	1.0000	0.5364	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
FlexZBoost	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.4	0.0000	0.0004	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mixture Density Network	1.0	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.4	0.3582	1.0000	1.0000	0.0002	1.0000	1.0000	0.0446	1.0000	1.0000	–	–	–
	1.6	0.1802	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	1.0000	0.6182	0.0000	1.0000	0.5702	0.0000	0.6822	0.3766	0.0000	0.0734	0.0278

Notes: For details on variable definitions and methodology, refer to Table 1.

Table 4: Model Confidence Set Test Results: MARMA(1, 1, 1, 1) Process, 1-Step-Ahead Forecasts

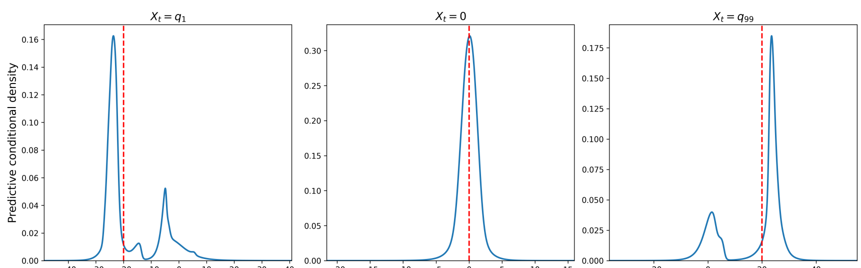
Model	α	$\mathbb{E}[y_{t+1} \mid y_t]$			$\mathbb{E}[y_{t+1}^2 \mid y_t]$			$\mathbb{E}[y_{t+1}^3 \mid y_t]$			$\mathbb{E}[y_{t+1}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.4	1.0000	0.0002	0.0004	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	–	–	–
	1.6	1.0000	0.1940	0.3650	1.0000	0.0000	0.0004	1.0000	0.0980	0.1874	1.0000	0.0006	0.0024
	1.8	1.0000	0.0092	0.1660	1.0000	0.0162	0.8638	1.0000	0.2392	1.0000	1.0000	1.0000	1.0000
FlexZBoost	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.4	0.0000	0.0002	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0000	0.0002	0.0002	0.0000	0.0006	0.0024
	1.8	0.0000	0.0000	0.0000	0.0000	0.0050	0.0158	0.0000	0.0256	0.0280	0.0000	0.0388	0.0340
Mixture Density Network	1.0	0.7244	1.0000	1.0000	0.0284	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0010	1.0000	1.0000	0.0350	1.0000	1.0000	0.0064	1.0000	1.0000	–	–	–
	1.4	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0036	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	0.8070	0.0000	0.3872	0.0340

Notes: For details on variable definitions and methodology, refer to Table 1.

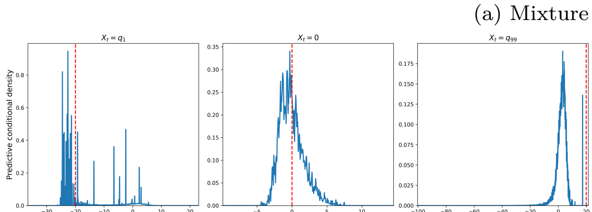
3.2 Bimodality Analysis and Sampled Trajectories (additional results to Section 3.3)

Figure 1: Animated GIF, 1-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process Computed with Mixture Density Network

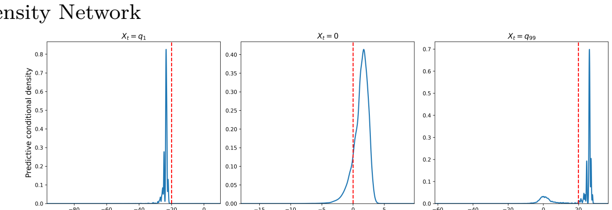
Notes: This animated figure requires a PDF reader with animation support and may not display correctly in all PDF viewers. The figure displays the estimated conditional predictive density $\hat{p}(X_{t+h}|X_t)$ for a purely noncausal MAR(0,1) process with α-stable innovations ($\alpha = 1.4$) across the full range of conditioning values X_t in the grid. The red dashed line indicates the current conditioning value X_t



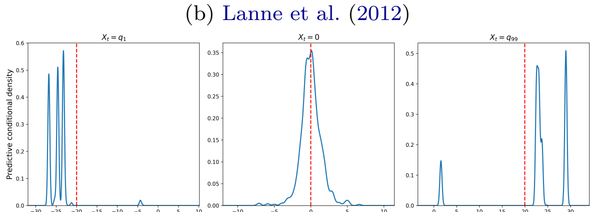
(a) Mixture Density Network



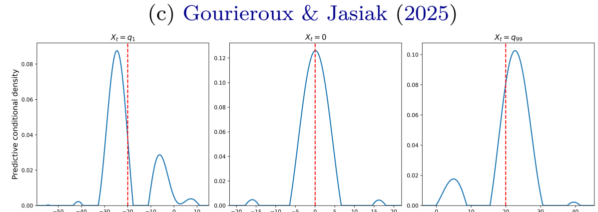
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



(d) Nadaraya-Watson



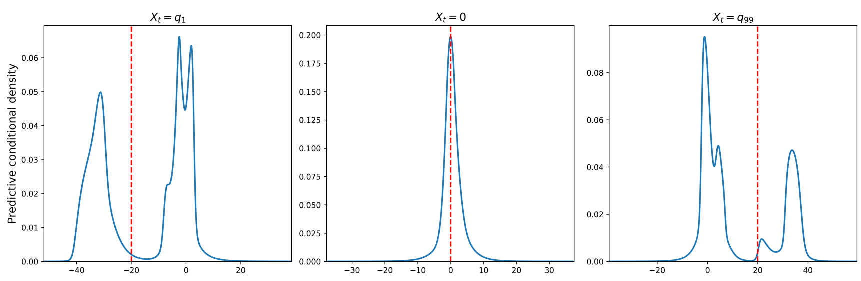
(e) FlexZBoost

Figure 2: 2-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process

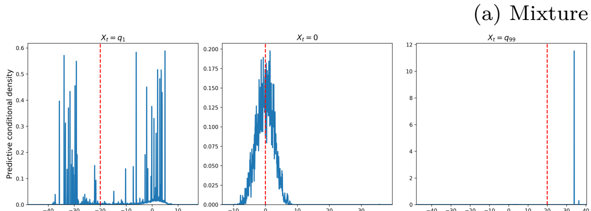
Notes: This figure displays the estimated conditional predictive density $\hat{p}(X_{t+h}|X_t)$ for a purely noncausal MAR(0,1) process with α-stable innovations at three conditioning values: $X_t = q_1$ (left tail), $X_t = 0$ (center), and $X_t = q_{99}$ (right tail), where $q_.$ denotes the percentiles. The red dashed line indicates the conditioning value X_t. The tail-index is fixed at $\alpha = 1.4$.

Figure 3: Animated GIF, 2-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process Computed with Mixture Density Network

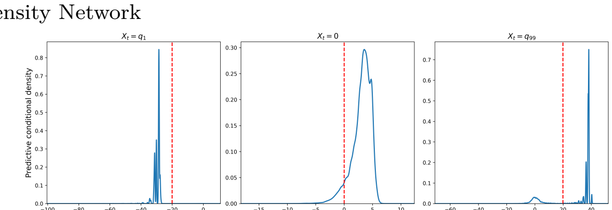
Notes: For details on variable definitions and methodology, refer to Figure 1.



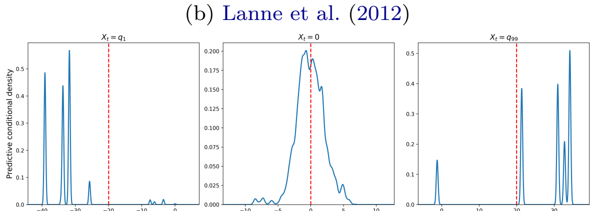
(a) Mixture Density Network



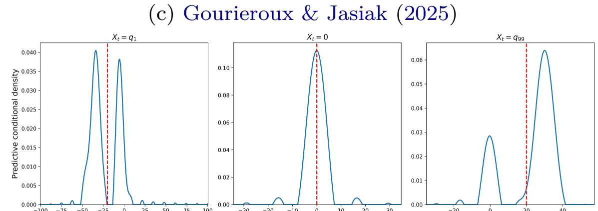
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



(d) Nadaraya-Watson



(e) FlexZBoost

Figure 4: 5-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process

Notes: For details on variable definitions and methodology, refer to Figure 2.

Figure 5: Animated GIF, 5-Step-Ahead Conditional Predictive Density of a MAR(0,1) Process Computed with Mixture Density Network

Notes: For details on variable definitions and methodology, refer to Figure 1.

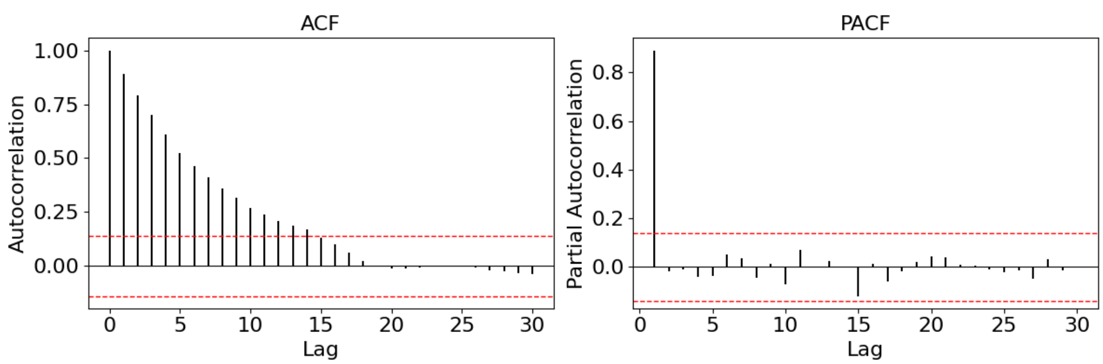


Figure 6: ACF - PACF of the MDN-sampled MAR(0,1) trajectory

Table 5: Estimated MAR(0,1) parameters from the MDN-sampled trajectory

Parameter	Estimate
ψ	0.887^{***} (0.033)
α	1.424^{***} (0.188)
β	0.297 (0.213)
σ	0.495^{***} (0.049)

Note: Standard errors in parentheses. *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

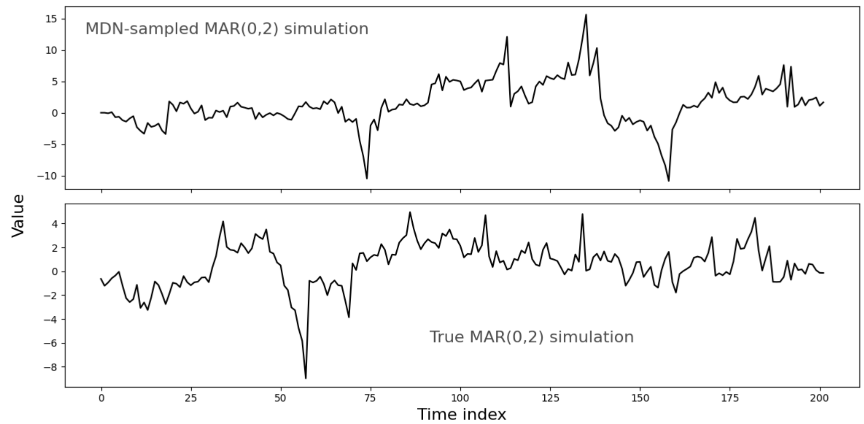


Figure 7: MDN-Sampled vs. True MAR(0,2) Trajectories
Notes: Top panel: trajectory generated by iteratively sampling from the MDN's one-step-ahead predictive density. Bottom panel: true MAR(0,2) simulation with $\alpha = 1.4$. The underlying MDN has been trained on a simulated sample of 5,000 realizations from the true MAR(0,2) specification with $\alpha = 1.4$.

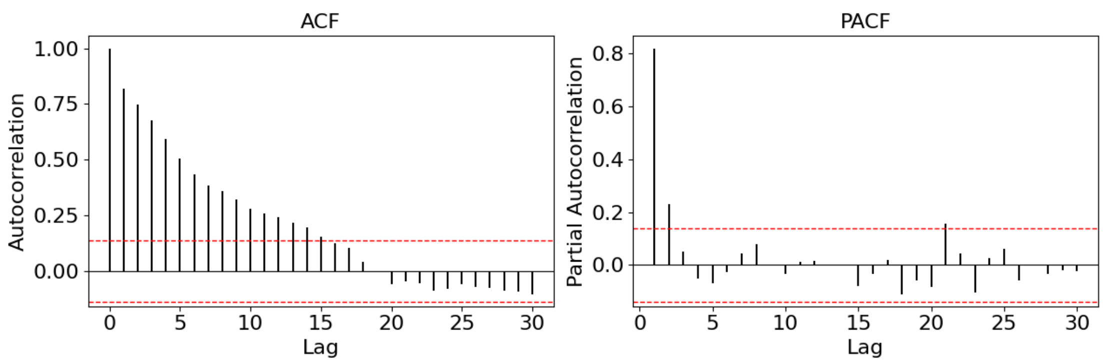


Figure 8: ACF - PACF of the MDN-sampled MAR(0,2) trajectory

Table 6: Estimated MAR(0,2) parameters from the MDN-sampled trajectory

Parameter	Estimate
ψ_1	0.665^{***} (0.067)
ψ_2	0.116^{*} (0.070)
α	1.475^{***} (0.191)
β	0.128 (0.202)
σ	0.651^{***} (0.065)

Note: Standard errors in parentheses. *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

3.3 Predictive Moments Approach (additional results to Section 3.4.2)

Table 7: Root Mean Squared Error of Predictive Moments: MAR(0, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.649**	47.73	45.37	**20.67**	8081	7680	–	–	–	–	–	–
	1.2	**0.206**	11.33	10.42	**2.559**	594.3	546.7	**24.12**	3.255e+04	2.994e+04	–	–	–
	1.4	**0.115**	3.587	3.160	**0.703**	87.25	76.86	**3.973**	2256	1988	–	–	–
	1.6	**0.074**	1.238	1.025	**0.299**	18.27	15.11	**0.998**	246.7	204.0	**13.62**	3179	2629
	1.8	**0.059**	0.249	0.191	**0.114**	**1.824**	**1.368**	**0.334**	**13.49**	**10.10**	**2.438**	**98.94**	**74.08**
Lanne et al. (2012)	1.0	2.219	42.41	40.31	308.1	8418	8001	–	–	–	–	–	–
	1.2	0.423	13.03	11.99	11.13	720.1	662.6	**77.19**	3.853e+04	3.545e+04	–	–	–
	1.4	0.226	4.655	4.102	2.379	111.4	98.18	14.24	2546	2242	–	–	–
	1.6	**0.102**	1.153	0.956	0.979	17.65	14.61	**4.259**	235.9	195.1	**28.73**	3159	2612
	1.8	**0.051**	**0.152**	**0.119**	**0.441**	2.660	2.012	**1.652**	25.25	18.93	**7.370**	211.8	158.6
Gourieroux and Jasiak (2025)	1.0	9.917	26.59	25.46	248.3	**4203**	**3995**	–	–	–	–	–	–
	1.2	2.157	**5.156**	**4.818**	22.64	**223.9**	**206.2**	241.6	**1.175e+04**	**1.081e+04**	–	–	–
	1.4	0.917	**2.199**	**1.985**	5.341	**40.36**	**35.65**	31.38	**865.1**	**762.2**	–	–	–
	1.6	0.617	**1.046**	**0.932**	2.387	**10.94**	**9.151**	9.222	**130.8**	**108.3**	42.00	**1546**	**1278**
	1.8	0.375	0.357	0.365	1.169	2.461	1.999	3.577	18.69	14.19	12.21	**147.8**	**110.9**
FlexZBoost	1.0	2.206	**25.81**	**24.54**	2475	9855	9398	–	–	–	–	–	–
	1.2	0.792	6.277	5.783	96.96	597.5	551.0	347.7	5.265e+04	4.844e+04	–	–	–
	1.4	0.366	3.232	2.853	9.567	95.95	84.64	83.94	3049	2686	–	–	–
	1.6	0.271	1.376	1.148	2.220	24.27	20.11	29.45	407.5	337.5	779.3	7719	6399
	1.8	0.260	0.358	0.318	1.464	3.312	2.662	11.07	24.64	19.85	150.8	281.3	233.1
Mixture Density Network	1.0	**1.437**	**11.20**	**10.66**	**49.95**	**1199**	**1140**	–	–	–	–	–	–
	1.2	**0.358**	**3.424**	**3.153**	**5.574**	**155.1**	**142.7**	196.1	**1.007e+04**	**9263**	–	–	–
	1.4	**0.177**	**0.462**	**0.415**	**1.585**	**12.47**	**11.01**	**13.50**	**374.5**	**329.9**	–	–	–
	1.6	0.124	**0.634**	**0.529**	**0.973**	**6.715**	**5.581**	5.234	**83.05**	**68.75**	89.26	**1014**	**840.3**
	1.8	0.135	**0.217**	**0.186**	0.944	**1.707**	**1.423**	4.459	**14.88**	**11.52**	48.13	149.2	116.2

Notes: This Table reports the root mean squared error (RMSE) of estimated predictive moments relative to theoretical values. α denotes the tail index of the stable distribution. Predictive moments are evaluated over three spatial regions: Center $[q_{0.1}, q_{0.9}]$, Tails $[q_{0.01}, q_{0.1}] \cup [q_{0.9}, q_{0.99}]$, and Total $[q_{0.01}, q_{0.99}]$, where q_p represents the p-th quantile. Best method in red, second best in **bold black**.

Table 8: Model Confidence Set Test Results: MAR(0, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0002	0.0002	–	–	–
	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.6	1.0000	0.0000	0.0064	1.0000	0.0000	0.0000	1.0000	0.0004	0.0002	1.0000	0.0014	0.0050
	1.8	0.2548	0.0032	0.0116	1.0000	0.3694	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Lanne et al. (2012)	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0040	0.0000	0.0000	–	–	–
	1.6	0.0028	0.0884	0.1180	0.0000	0.0000	0.0000	0.0000	0.0004	0.0002	0.0714	0.0000	0.0004
	1.8	1.0000	1.0000	1.0000	0.0000	0.0238	0.0090	0.0002	0.0004	0.0068	0.0004	0.0000	0.0002
Gourieroux and Jasiak (2025)	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0028	0.0030	0.0000	1.0000	1.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0002	0.0000	0.0028	0.0092
	1.8	0.0000	0.0000	0.0000	0.0000	0.0030	0.0000	0.0000	0.0004	0.0000	0.0000	0.0000	0.0002
FlexZBoost	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0018	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mixture Density Network	1.0	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0016	1.0000	1.0000	0.0068	1.0000	1.0000	0.0000	0.5836	0.6088	–	–	–
	1.4	0.0042	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0210	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	0.7304	0.0226	0.0000	1.0000	0.1066	0.0000	0.4720	0.0358	0.0000	0.0000	0.0000

Notes: This Table reports MCS p-values from the Model Confidence Set procedure of Hansen et al. (2011) applied at the 90% confidence level ($\alpha = 0.10$). Values represent the probability threshold at which each model enters the set of superior forecasting methods. A model belongs to the $\text{MCS}_{90\%}$ if its p-value ≥ 0.10. Values in red indicate models included in the MCS (i.e., models with statistically indistinguishable superior performance). The loss function is the squared error of predictive moments relative to theoretical ones. See Table 7 for corresponding RMSE values.

Table 9: Root Mean Squared Error of Predictive Moments: MAR(0, 2) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.327**	29.60	28.13	**3.139**	3044	2893	–	–	–	–	–	–
	1.2	**0.186**	7.053	6.489	**1.178**	**287.5**	**264.5**	**14.32**	**1.371e+04**	**1.261e+04**	–	–	–
	1.4	**0.091**	1.944	1.713	**0.347**	**36.00**	**31.71**	**1.623**	**742.3**	**653.9**	–	–	–
	1.6	**0.053**	**0.489**	**0.405**	**0.095**	4.481	**3.706**	**0.429**	**44.21**	**36.57**	**4.731**	**561.2**	**464.2**
	1.8	**0.040**	**0.107**	**0.084**	**0.102**	**0.467**	**0.356**	**0.168**	**2.635**	**1.976**	**0.703**	**18.60**	**13.93**
FlexZBoost	1.0	1.277	**14.24**	**13.54**	614.5	**2716**	**2589**	–	–	–	–	–	–
	1.2	0.573	**5.459**	**5.027**	29.36	334.3	307.8	73.93	2.430e+04	2.235e+04	–	–	–
	1.4	0.363	**1.916**	**1.697**	3.402	40.25	35.49	23.45	1057	931.1	–	–	–
	1.6	0.258	**0.465**	**0.411**	1.878	**4.446**	3.826	20.30	72.63	61.14	396.6	1506	1265
	1.8	0.205	0.368	0.307	0.874	2.314	1.826	4.440	17.90	13.72	42.63	153.7	118.4
Mixture Density Network	1.0	**0.737**	**13.00**	**12.35**	**35.31**	**1151**	**1094**	–	–	–	–	–	–
	1.2	**0.177**	**3.082**	**2.836**	**3.083**	**101.4**	**93.33**	**56.72**	**3916**	**3602**	–	–	–
	1.4	**0.114**	**1.016**	**0.897**	**1.224**	**17.09**	**15.06**	**9.612**	**288.7**	**254.4**	–	–	–
	1.6	**0.097**	0.591	0.492	**0.827**	**3.300**	**2.768**	**5.085**	**63.18**	**52.33**	**56.81**	**411.5**	**341.8**
	1.8	**0.076**	**0.197**	**0.156**	**0.649**	**0.642**	**0.644**	**3.233**	**5.399**	**4.570**	**25.78**	**48.24**	**39.92**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 10: Model Confidence Set Test Results: MAR(0, 2) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.7282	0.0000	0.0000	1.0000	0.0002	0.0002	1.0000	0.0016	0.0016	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0026	0.0030	1.0000	0.0004	0.0028	1.0000	0.0004	0.0008	–	–	–
	1.6	1.0000	0.9786	1.0000	1.0000	0.0062	0.0800	1.0000	1.0000	1.0000	1.0000	0.0748	0.1892
	1.8	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1.0	0.0000	1.0000	1.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	0.0002	–	–	–
	1.6	0.0000	1.0000	0.3232	0.0000	0.0062	0.0010	0.0000	0.0482	0.0162	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.0	0.0000	0.5762	0.6270	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	1.0000	1.0000	1.0000	0.0028	1.0000	1.0000	0.0312	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.1258	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0004	0.7866	0.4288	0.0000	1.0000	1.0000	0.0000	0.1162	0.1022	0.0000	1.0000	1.0000
	1.8	0.0680	0.0000	0.0000	0.0000	0.0132	0.0000	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 9 for corresponding RMSE values.

Table 11: Root Mean Squared Error of Predictive Moments: MAR(1, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	**0.842**	54.89	52.16	**29.23**	**1.046e+04**	**9942**	–	–	–	–	–	–
	1.2	**0.232**	11.76	10.82	**3.018**	**705.6**	**649.2**	**35.65**	**4.241e+04**	**3.902e+04**	–	–	–
Nadaraya-Watson	1.4	**0.127**	3.873	3.413	**0.818**	**106.5**	**93.83**	**5.471**	**3018**	**2658**	–	–	–
	1.6	**0.080**	**1.285**	**1.063**	**0.369**	**20.73**	**17.15**	**1.495**	**316.3**	**261.6**	**17.96**	**4631**	**3830**
	1.8	**0.064**	**0.279**	**0.213**	**0.130**	**2.515**	**1.885**	**0.393**	**21.57**	**16.15**	**2.794**	**179.3**	**134.3**
	1.0	2.424	**28.40**	**27.00**	3058	1.234e+04	1.177e+04	–	–	–	–	–	–
	1.2	0.866	**6.879**	**6.338**	119.5	757.2	698.2	516.2	7.308e+04	6.724e+04	–	–	–
FlexZBoost	1.4	0.465	**3.189**	**2.818**	12.07	109.0	96.21	120.0	3802	3349	–	–	–
	1.6	0.289	1.461	1.219	2.279	27.65	22.90	21.76	495.4	409.9	526.3	9910	8201
	1.8	0.253	0.533	0.433	1.740	5.350	4.168	15.45	52.22	40.42	239.6	584.5	465.5
	1.0	**1.583**	**7.698**	**7.333**	**62.87**	**878.5**	**835.1**	–	–	–	–	–	–
	1.2	**0.268**	**2.554**	**2.352**	**4.874**	**153.0**	**140.7**	**240.3**	**1.099e+04**	**1.011e+04**	–	–	–
Mixture Density Network	1.4	**0.195**	**0.737**	**0.656**	**1.525**	**18.40**	**16.23**	**15.13**	**572.0**	**503.9**	–	–	–
	1.6	**0.135**	**0.444**	**0.375**	**1.108**	**4.044**	**3.402**	**7.931**	**46.85**	**39.00**	**115.5**	**533.1**	**445.6**
	1.8	**0.098**	**0.294**	**0.230**	**0.837**	**2.939**	**2.269**	**4.058**	**30.49**	**22.98**	**48.13**	**343.2**	**258.9**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 12: Model Confidence Set Test Results: MAR(1, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.6	1.0000	0.0004	0.0028	1.0000	0.0000	0.0008	1.0000	0.0002	0.0010	1.0000	0.0006	0.0024
	1.8	1.0000	0.9888	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.0	0.0002	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.1592	1.0000	1.0000	0.0366	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.0038	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0234	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0044	1.0000	0.5752	0.0000	0.3582	0.0210	0.0000	0.0100	0.0044	0.0000	0.0000	0.0002

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 11 for corresponding RMSE values.

Table 13: Root Mean Squared Error of Predictive Moments: MARMA(1, 1, 1, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	**0.590**	32.00	30.41	**18.81**	**4448**	**4227**	–	–	–	–	–	–
	1.2	**0.171**	8.262	7.601	**1.489**	**260.5**	**239.7**	**12.81**	**1.340e+04**	**1.233e+04**	–	–	–
Nadaraya-Watson	1.4	**0.117**	3.848	3.390	**0.513**	68.64	60.47	**4.802**	**1735**	**1528**	–	–	–
	1.6	**0.067**	**1.248**	**1.033**	**0.187**	**8.975**	**7.423**	**1.119**	**98.39**	**81.37**	**10.84**	**1125**	**930.2**
	1.8	**0.051**	**0.526**	**0.395**	**0.128**	**2.708**	**2.029**	**0.371**	**14.91**	**11.17**	**1.526**	**89.14**	**66.74**
	1.0	1.721	**21.61**	**20.55**	1767	5873	5609	–	–	–	–	–	–
	1.2	0.639	**4.934**	**4.546**	66.19	469.0	432.3	198.3	5.234e+04	4.816e+04	–	–	–
FlexZBoost	1.4	0.362	**2.576**	**2.276**	6.652	**61.88**	**54.60**	64.18	2657	2341	–	–	–
	1.6	0.238	1.445	1.203	1.695	19.99	16.56	18.94	283.2	234.5	380.4	8777	7262
	1.8	0.219	0.662	0.516	0.864	4.351	3.307	5.816	44.27	33.37	63.45	439.7	331.9
	1.0	**0.877**	**5.987**	**5.697**	**40.34**	**632.3**	**601.1**	–	–	–	–	–	–
	1.2	**0.337**	**1.162**	**1.078**	**1.961**	**38.89**	**35.79**	**47.69**	**2831**	**2604**	–	–	–
Mixture Density Network	1.4	**0.244**	**0.570**	**0.515**	**1.817**	**7.700**	**6.837**	**19.49**	**229.6**	**202.5**	–	–	–
	1.6	**0.139**	**0.363**	**0.310**	**1.119**	**3.612**	**3.053**	**8.019**	**54.81**	**45.56**	**108.5**	**753.8**	**626.4**
	1.8	**0.106**	**0.187**	**0.156**	**0.801**	**1.761**	**1.422**	**3.911**	**15.69**	**12.03**	**35.38**	**146.6**	**112.2**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 14: Model Confidence Set Test Results: MARMA(1, 1, 1, 1) Process, 2-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+2} \mid y_t]$			$\mathbb{E}[y_{t+2}^2 \mid y_t]$			$\mathbb{E}[y_{t+2}^3 \mid y_t]$			$\mathbb{E}[y_{t+2}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	1.0000	0.0000	0.0002	–	–	–
	1.6	1.0000	0.0000	0.0000	1.0000	0.0006	0.0094	1.0000	0.0212	0.0716	1.0000	0.5472	0.8434
	1.8	1.0000	0.0000	0.0086	1.0000	0.1394	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008	0.0000	0.0036	0.0058	0.0000	0.0106	0.0110
	1.8	0.0000	0.0000	0.0000	0.0000	0.1394	0.1298	0.0000	0.0522	0.0200	0.0000	0.0000	0.0000
	1.0	0.0036	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0002	1.0000	1.0000	0.0280	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	1.0000	1.0000	0.0000	1.0000	0.9328	0.0000	0.2890	0.0422	0.0000	0.0000	0.0000

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 13 for corresponding RMSE values.

Table 15: Root Mean Squared Error of Predictive Moments: MAR(0, 1) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	**1.141**	71.94	68.37	**49.85**	1.563e+04	1.485e+04	–	–	–	–	–	–
	1.2	**0.352**	19.06	17.53	**4.398**	1293	1190	**61.79**	9.265e+04	8.524e+04	–	–	–
Nadaraya-Watson	1.4	**0.233**	6.043	5.324	**1.406**	183.7	161.9	**14.86**	5771	5084	–	–	–
	1.6	**0.159**	2.317	1.919	**0.675**	37.50	31.02	**4.083**	609.7	504.3	**69.22**	9728	8045
	1.8	**0.079**	**0.427**	**0.324**	**0.264**	**4.348**	**3.259**	**1.148**	57.23	42.85	**9.333**	738.6	553.0
	1.0	4.204	51.83	49.28	926.6	1.329e+04	1.263e+04	–	–	–	–	–	–
	1.2	**0.506**	13.91	12.80	32.11	**1009**	**928.6**	**357.4**	**7.355e+04**	**6.767e+04**	–	–	–
Lanne et al. (2012)	1.4	**0.248**	4.965	4.376	5.521	**152.4**	**134.3**	**42.34**	**4715**	**4154**	–	–	–
	1.6	**0.067**	**1.330**	**1.100**	**2.390**	**25.62**	**21.24**	**13.23**	**437.9**	**362.2**	**111.3**	**8117**	**6713**
	1.8	**0.033**	**0.120**	**0.092**	**0.932**	4.643	3.530	**4.163**	**49.35**	**37.05**	**26.60**	**501.6**	**375.9**
	1.0	30.28	36.06	35.54	1419	**5616**	**5356**	–	–	–	–	–	–
	1.2	7.755	15.94	14.97	135.9	1101	1014	2743	8.688e+04	7.993e+04	–	–	–
Gourieroux and Jasiak (2025)	1.4	3.058	8.019	7.211	22.23	243.2	214.5	215.0	8017	7063	–	–	–
	1.6	1.919	3.839	3.353	7.658	66.57	55.22	42.70	1204	995.8	237.5	2.156e+04	1.783e+04
	1.8	1.065	1.114	1.093	2.649	10.32	7.919	8.815	103.1	77.40	31.62	1035	775.0
	1.0	2.427	**35.47**	**33.72**	2454	1.371e+04	1.305e+04	–	–	–	–	–	–
	1.2	0.969	**11.41**	**10.50**	93.06	1064	979.7	475.9	1.191e+05	1.096e+05	–	–	–
FlexZBoost	1.4	0.642	**4.802**	**4.241**	8.841	177.3	156.2	124.5	6302	5552	–	–	–
	1.6	0.481	1.997	1.674	4.565	43.30	35.90	84.33	811.4	672.8	2642	1.985e+04	1.648e+04
	1.8	0.392	0.594	0.515	2.384	4.904	3.997	17.01	52.43	40.84	255.8	605.7	484.1
	1.0	**2.091**	**23.33**	**22.18**	**92.59**	**3325**	**3160**	–	–	–	–	–	–
	1.2	0.512	**5.074**	**4.672**	**10.97**	**277.5**	**255.3**	627.1	**2.122e+04**	**1.953e+04**	–	–	–
Mixture Density Network	1.4	0.287	**1.990**	**1.759**	**3.817**	**52.67**	**46.43**	47.33	**1586**	**1397**	–	–	–
	1.6	0.217	**0.995**	**0.832**	2.394	**14.65**	**12.19**	19.03	**256.2**	**212.1**	376.6	**4489**	**3719**
	1.8	0.213	0.427	0.350	1.793	**3.723**	**3.030**	11.61	**40.76**	**31.47**	145.7	**489.0**	**378.6**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 16: Model Confidence Set Test Results: MAR(0, 1) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	–	–	–
	1.6	0.0000	0.0050	0.0084	1.0000	0.0006	0.0076	1.0000	0.0048	0.0162	1.0000	0.0084	0.0264
	1.8	0.0000	0.0000	0.0000	1.0000	0.5260	1.0000	1.0000	0.2638	1.0000	1.0000	0.2300	1.0000
Lanne et al. (2012)	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	–	–	–
	1.4	0.4140	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	–	–	–
	1.6	1.0000	1.0000	1.0000	0.0000	0.0042	0.0118	0.0000	0.0332	0.0686	0.0000	0.0084	0.0264
	1.8	1.0000	1.0000	1.0000	0.0000	0.0006	0.0008	0.0000	0.0314	0.2966	0.0000	0.2396	0.9590
Gourieroux and Jasiak (2025)	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.0006	0.0008	0.0000	0.0314	0.0936	0.0000	0.1122	0.1786
FlexZBoost	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0000	0.0000	0.0000	0.1074	0.0008	0.0000	0.0666	0.0936	0.0000	0.1122	0.0086
Mixture Density Network	1.0	0.0000	1.0000	1.0000	0.0076	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0090	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.4	0.4140	1.0000	1.0000	0.0000	1.0000	1.0000	0.0002	1.0000	1.0000	–	–	–
	1.6	0.0000	0.3778	0.2262	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	0.0000	0.0000	0.0000	1.0000	0.0198	0.0000	1.0000	0.6834	0.0000	1.0000	0.9590

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 15 for corresponding RMSE values.

Table 17: Root Mean Squared Error of Predictive Moments: MAR(0, 2) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**0.646**	60.19	57.21	**20.01**	1.135e+04	1.079e+04	–	–	–	–	–	–
	1.2	**0.311**	12.69	11.67	**3.962**	980.5	902.1	**83.75**	8.826e+04	8.120e+04	–	–	–
	1.4	**0.133**	**2.975**	**2.621**	**0.486**	**86.79**	**76.46**	**4.042**	**3268**	**2879**	–	–	–
	1.6	**0.073**	1.020	0.845	**0.290**	12.51	10.35	**1.451**	192.7	159.3	**45.88**	**3476**	**2875**
	1.8	**0.053**	**0.207**	**0.159**	**0.217**	**1.714**	**1.291**	**1.738**	**15.79**	**11.88**	**14.16**	**199.5**	**149.6**
FlexZBoost	1.0	1.672	**19.84**	**18.87**	598.1	**4347**	**4136**	–	–	–	–	–	–
	1.2	0.863	**6.588**	**6.070**	28.40	**561.1**	**516.4**	**297.8**	**5.433e+04**	**4.998e+04**	–	–	–
	1.4	0.497	3.192	2.822	6.064	112.8	99.44	123.2	4601	4053	–	–	–
	1.6	0.382	**0.757**	**0.662**	3.038	**9.996**	**8.442**	34.63	**165.7**	**138.4**	694.7	3904	3253
	1.8	0.280	0.439	0.378	1.341	3.241	2.584	8.245	29.56	22.80	84.25	297.1	229.3
Mixture Density Network	1.0	**1.464**	**17.64**	**16.77**	**115.0**	**2946**	**2800**	–	–	–	–	–	–
	1.2	**0.398**	**2.263**	**2.088**	**7.633**	**154.3**	**142.0**	329.2	**9495**	**8736**	–	–	–
	1.4	**0.244**	**1.306**	**1.156**	**3.448**	**43.70**	**38.53**	**40.85**	**1351**	**1191**	–	–	–
	1.6	**0.119**	**0.330**	**0.281**	**1.751**	**8.035**	**6.718**	**10.45**	**133.2**	**110.3**	**194.6**	**3032**	**2510**
	1.8	**0.099**	**0.262**	**0.207**	**0.829**	**1.549**	**1.283**	**3.112**	**16.29**	**12.37**	**32.38**	**194.6**	**147.3**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 18: Model Confidence Set Test Results: MAR(0, 2) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0002	0.0002	1.0000	0.0000	0.0004	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0000	0.0004	1.0000	0.0102	0.0168	1.0000	0.0094	0.0118	–	–	–
	1.6	1.0000	0.0000	0.0000	1.0000	0.0064	0.1176	1.0000	0.0164	0.0694	1.0000	0.9702	1.0000
	1.8	1.0000	1.0000	1.0000	1.0000	0.3926	1.0000	1.0000	1.0000	1.0000	1.0000	0.9716	1.0000
	1.0	0.0000	0.0944	0.1002	0.0000	0.0000	0.0002	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0002	0.0002	0.0002	0.0000	0.0004	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0002	0.0000	0.0102	0.0168	0.0000	0.0016	0.0044	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.2070	0.1456	0.0000	0.4300	0.2496	0.0000	0.9702	0.6976
	1.8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008	0.0002
	1.0	0.0000	1.0000	1.0000	0.0006	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0146	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.0004	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0072	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	0.9982
	1.8	0.0092	0.0000	0.0000	0.0000	1.0000	0.0194	0.0000	0.8694	0.4522	0.0000	1.0000	0.4698

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 17 for corresponding RMSE values.

Table 19: Root Mean Squared Error of Predictive Moments: MAR(1, 1) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	**1.406**	78.39	74.50	**65.66**	1.956e+04	1.859e+04	–	–	–	–	–	–
	1.2	**0.429**	21.34	19.63	**5.324**	1619	1489	**90.96**	**1.280e+05**	**1.178e+05**	–	–	–
Nadaraya-Watson	1.4	**0.262**	7.069	6.228	**1.539**	239.7	211.2	**18.37**	8331	7339	–	–	–
	1.6	**0.170**	2.572	2.129	**0.754**	**46.84**	**38.74**	**4.959**	**840.1**	**694.8**	**93.05**	**1.468e+04**	**1.214e+04**
	1.8	**0.085**	**0.479**	**0.363**	**0.302**	**5.606**	**4.202**	**1.581**	**79.92**	**59.83**	**13.37**	**1127**	**843.5**
	1.0	2.579	**39.30**	**37.36**	3037	**1.660e+04**	**1.581e+04**	–	–	–	–	–	–
	1.2	1.242	**11.74**	**10.81**	114.1	**1336**	**1230**	739.6	1.562e+05	1.437e+05	–	–	–
FlexZBoost	1.4	0.739	**5.023**	**4.438**	11.33	**205.4**	**181.0**	192.2	**8268**	**7284**	–	–	–
	1.6	0.523	**2.267**	**1.898**	5.377	55.86	46.30	103.7	1170	969.2	3413	3.149e+04	2.611e+04
	1.8	0.441	0.723	0.615	2.644	8.440	6.557	20.66	113.4	85.99	343.1	1568	1196
	1.0	**2.248**	**22.66**	**21.55**	**88.25**	**3912**	**3718**	–	–	–	–	–	–
	1.2	**0.614**	**5.118**	**4.715**	**13.53**	**341.2**	**314.0**	**637.2**	**2.851e+04**	**2.623e+04**	–	–	–
Mixture Density Network	1.4	**0.283**	**1.750**	**1.547**	**3.791**	**53.54**	**47.19**	**50.63**	**1767**	**1556**	–	–	–
	1.6	**0.237**	**1.022**	**0.856**	**2.750**	**18.30**	**15.21**	**23.14**	**380.9**	**315.3**	**494.6**	**8415**	**6965**
	1.8	**0.192**	**0.536**	**0.421**	**1.989**	**5.180**	**4.096**	**12.78**	**60.98**	**46.43**	**188.6**	**822.4**	**628.2**

Notes: For details on variable definitions and methodology, refer to Table 7.

Table 20: Model Confidence Set Test Results: MAR(1, 1) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
	1.0	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
Nadaraya-Watson	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0004	1.0000	0.0000	0.0000	–	–	–
	1.6	1.0000	0.0096	0.0216	1.0000	0.0006	0.0056	1.0000	0.0034	0.0092	1.0000	0.0108	0.0294
	1.8	1.0000	1.0000	1.0000	1.0000	0.7494	1.0000	1.0000	0.1408	1.0000	1.0000	0.1438	1.0000
	1.0	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0002	0.0000	0.0002	0.0000	0.0000	0.0004	0.0000	–	–	–
FlexZBoost	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	1.8	0.0000	0.0006	0.0000	0.0000	0.0042	0.0002	0.0000	0.0032	0.0028	0.0000	0.0044	0.0010
	1.0	0.0036	1.0000	1.0000	0.0990	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0078	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
Mixture Density Network	1.4	0.7260	1.0000	1.0000	0.0000	1.0000	1.0000	0.0012	1.0000	1.0000	–	–	–
	1.6	0.0220	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0000	0.4446	0.0412	0.0000	1.0000	0.0052	0.0000	1.0000	0.7536	0.0000	1.0000	0.4730

Notes: For details on variable definitions and methodology, refer to Table 8. See Table 19 for corresponding RMSE values.

Table 21: Root Mean Squared Error of Predictive Moments: MARMA(1, 1, 1, 1) Process, 5-Step-Ahead Forecasts

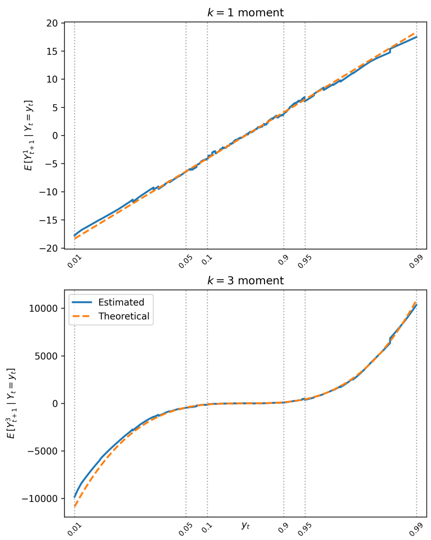
Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	**1.196**	62.23	59.14	**26.36**	**1.014e+04**	**9640**	–	–	–	–	–	–
	1.2	**0.328**	13.40	12.33	**3.038**	788.4	725.3	**62.72**	5.543e+04	5.099e+04	–	–	–
	1.4	**0.199**	4.897	4.315	**0.937**	132.4	116.6	**7.448**	**4080**	**3594**	–	–	–
	1.6	**0.087**	**1.815**	**1.502**	**0.224**	**22.89**	**18.93**	**1.756**	**310.9**	**257.2**	**24.82**	**4377**	**3620**
	1.8	**0.076**	**0.484**	**0.366**	**0.177**	**2.947**	**2.209**	**0.823**	**21.71**	**16.26**	**4.807**	**174.3**	**130.5**
FlexZBoost	1.0	2.129	**29.04**	**27.61**	1747	1.121e+04	1.067e+04	–	–	–	–	–	–
	1.2	0.946	**8.269**	**7.616**	63.64	**626.2**	**576.7**	287.9	**5.139e+04**	**4.728e+04**	–	–	–
	1.4	0.525	**4.014**	**3.545**	6.846	**109.7**	**96.73**	147.7	4537	3997	–	–	–
	1.6	0.355	2.008	1.673	3.500	31.19	25.87	55.75	440.6	365.7	1320	1.021e+04	8477
	1.8	0.284	0.906	0.704	**1.074**	7.723	5.825	7.401	74.10	55.69	84.15	740.3	557.0
Mixture Density Network	1.0	**1.023**	**13.55**	**12.88**	**121.2**	**3338**	**3173**	–	–	–	–	–	–
	1.2	**0.467**	**2.135**	**1.973**	**7.168**	**143.7**	**132.3**	**224.4**	**8459**	**7783**	–	–	–
	1.4	**0.233**	**1.032**	**0.915**	**4.314**	**31.35**	**27.69**	**47.73**	**1209**	**1065**	–	–	–
	1.6	**0.191**	**0.458**	**0.394**	**2.131**	**8.030**	**6.748**	**17.18**	**163.7**	**135.7**	**283.0**	**3511**	**2908**
	1.8	**0.128**	**0.156**	**0.144**	1.241	**2.412**	**1.984**	**5.718**	**19.27**	**14.91**	**75.77**	**230.1**	**179.4**

Notes: For details on variable definitions and methodology, refer to Table 7.

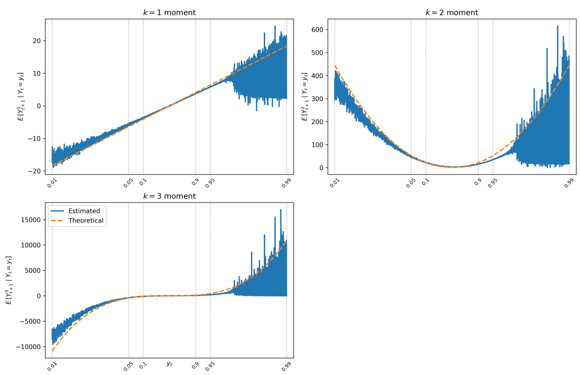
Table 22: Model Confidence Set Test Results: MARMA(1, 1, 1, 1) Process, 5-Step-Ahead Forecasts

Model	α	$\mathbb{E}[y_{t+5} \mid y_t]$			$\mathbb{E}[y_{t+5}^2 \mid y_t]$			$\mathbb{E}[y_{t+5}^3 \mid y_t]$			$\mathbb{E}[y_{t+5}^4 \mid y_t]$		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	0.8642	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	–	–	–
	1.4	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	–	–	–
	1.6	1.0000	0.0000	0.0000	1.0000	0.0000	0.0002	1.0000	0.0010	0.0070	1.0000	0.1186	0.2564
	1.8	1.0000	0.0000	0.0004	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
FlexZBoost	1.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–	–	–	–
	1.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	–	–	–
	1.6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000	0.0004	0.0002
	1.8	0.0000	0.0000	0.0000	0.0000	0.1674	0.0234	0.0000	0.0664	0.0306	0.0000	0.0012	0.0000
Mixture Density Network	1.0	1.0000	1.0000	1.0000	0.0120	1.0000	1.0000	–	–	–	–	–	–
	1.2	0.0960	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.4	0.2104	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	–	–	–
	1.6	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000	0.0000	1.0000	1.0000
	1.8	0.0002	1.0000	1.0000	0.0000	0.8296	0.0234	0.0000	0.2250	0.0306	0.0000	0.0012	0.0000

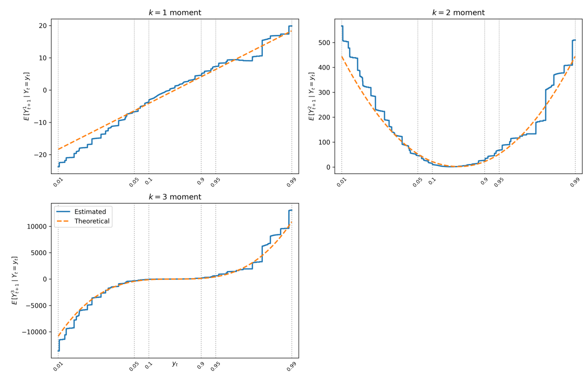
Notes: For details on variable definitions and methodology, refer to Table 8. See Table 21 for corresponding RMSE values.



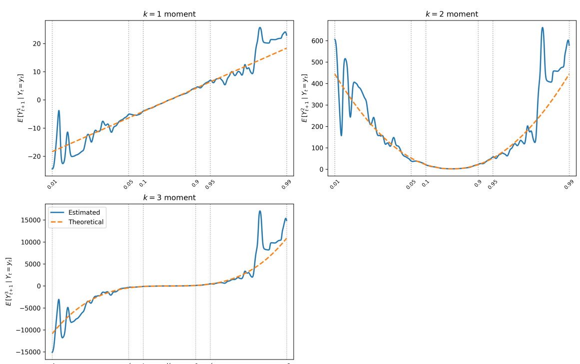
(a) Mixture Density Network



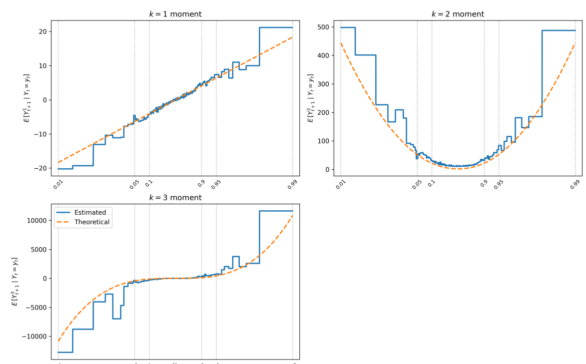
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



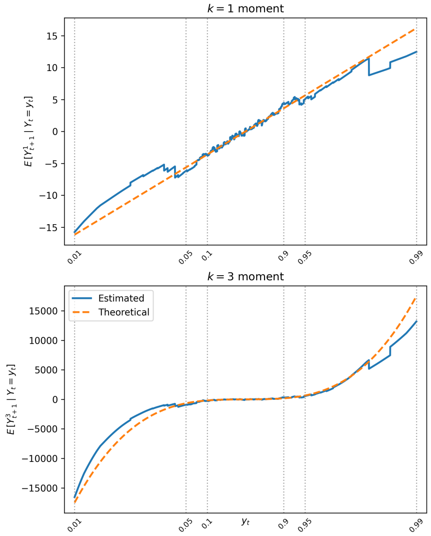
(d) Nadaraya-Watson



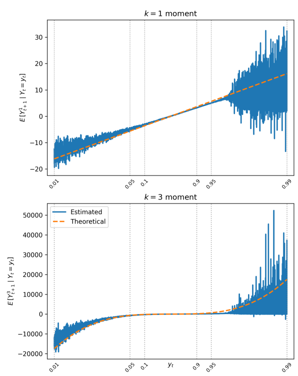
(e) FlexZBoost

Figure 9: Conditional Predictive Moment Accuracy: MAR(0,1) Process, 2-Step-Ahead Forecasts, with tail-index $\alpha = 1.4$

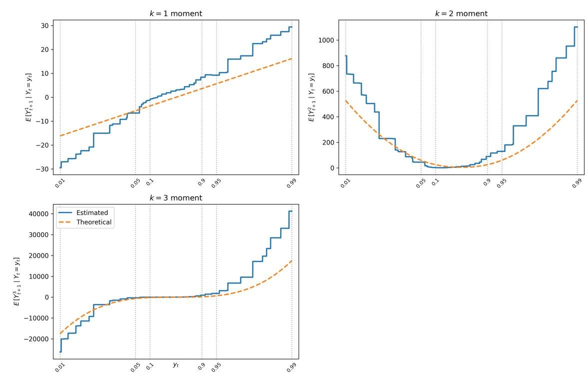
Notes: This figure displays the estimated predictive moments $\mathbb{E}[X_{t+h}^k \mid X_t]$ for $k \in \{1, 2, 3\}$ as a function of the conditioning variable X_t for a purely noncausal MAR(0,1) process with α-stable innovations. Each panel from (a) to (e) shows the results for a specific density forecasting method. Blue curves represent estimated moments, while orange curves show the corresponding theoretical ones.



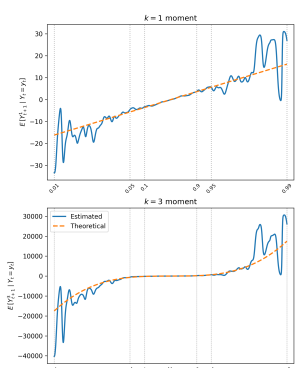
(a) Mixture Density Network



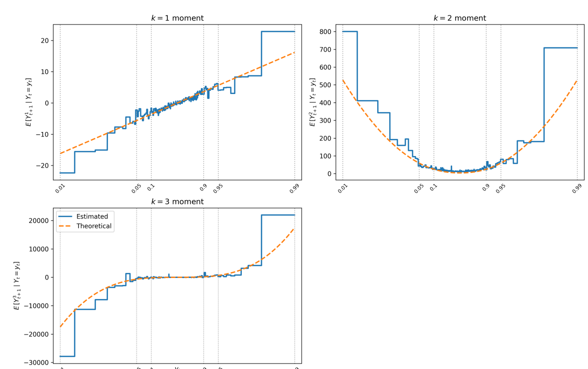
(b) Lanne et al. (2012)



(c) Gourieroux & Jasiak (2025)



(d) Nadaraya-Watson



(e) FlexZBoost

Figure 10: Conditional Predictive Moment Accuracy: MAR(0,1) Process, 5-Step-Ahead Forecasts, with tail-index $\alpha = 1.4$

Notes: For details on variable definitions and methodology, refer to Figure 9.

3.4 Comparison with realized outcomes (additional results to Section 3.4.3)

Let $\hat{p}_h(x \mid X_t)$ denote the estimated predictive density and $\hat{F}_h(x \mid X_t) = \int_{-\infty}^{x} \hat{p}_h(u \mid X_t)du$ the corresponding cumulative distribution function.

Logarithmic score:

$$\text{LogS} = -\log \hat{p}_h(X_{t+h} \mid X_t) \tag{4}$$

Continuous Ranked Probability Score:

$$\text{CRPS} = \int_{-\infty}^{\infty} \left(\hat{F}_h(x \mid X_t) - \mathbf{1}(X_{t+h} \leq x)\right)^2 dx \tag{5}$$

CDE loss:

$$\text{CDE} = \int_{-\infty}^{\infty} \hat{p}_h(x \mid X_t)^2 dx - 2\hat{p}_h(X_{t+h} \mid X_t) \tag{6}$$

Quantile score at level τ:

$$\text{QS}_\tau = (\tau - \mathbf{1}(X_{t+h} < \hat{q}_\tau))\,(X_{t+h} - \hat{q}_\tau) \tag{7}$$

where $\hat{q}_\tau$ satisfies $\hat{F}_h(\hat{q}_\tau \mid X_t) = \tau$.

Table 23: Density Forecast Performance Metrics: MAR(0, 1) Process, 2-Step-Ahead Forecasts

Model	α	CDE Loss			CRPS			Log Prob			QS 10%		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	-0.115	0.078	-0.092	1.712	5.874	2.880	-2.852	-7.174	-3.645	0.593	2.293	1.047
Lanne et al. (2012)		0.080	0.039	0.061	5.750	7.122	6.707	-4.380	-4.413	-4.996	1.154	1.163	1.444
Gourieroux and Jasiak (2025)		0.095	0.181	0.163	6.743	8.384	9.474	-4.532	-5.957	-5.449	2.994	4.939	3.790
FlexZBoost		-0.033	-0.014	-0.011	11.15	12.63	11.06	-3.500	-4.525	-4.826	1.169	5.669	5.489
Mixture Density Network		-0.085	-0.075	-0.076	1.917	4.174	3.055	-2.689	-3.015	-2.913	0.733	1.171	0.999
Nadaraya-Watson	1.2	-0.163	-0.167	-0.156	0.989	1.852	1.656	-2.086	-3.320	-2.527	0.346	0.481	0.500
Lanne et al. (2012)		-0.085	-0.029	-0.070	1.212	2.194	1.717	-2.435	-3.167	-2.817	0.385	0.452	0.489
Gourieroux and Jasiak (2025)		0.092	0.108	0.129	1.959	2.506	2.690	-3.132	-3.527	-3.597	0.790	0.926	0.970
FlexZBoost		-0.080	-0.076	-0.049	1.713	3.041	2.720	-2.593	-2.842	-3.228	0.554	1.187	1.164
Mixture Density Network		-0.184	-0.183	-0.164	0.958	1.927	1.589	-1.935	-2.204	-2.176	0.320	0.428	0.416
Nadaraya-Watson	1.4	-0.204	-0.310	-0.191	0.718	0.830	1.068	-1.815	-1.662	-2.111	0.216	0.224	0.270
Lanne et al. (2012)		-0.203	-0.296	-0.169	0.739	0.912	1.030	-1.732	-2.002	-2.133	0.219	0.294	0.296
Gourieroux and Jasiak (2025)		-0.068	-0.160	-0.010	0.990	1.046	1.406	-2.148	-2.101	-2.630	0.347	0.400	0.464
FlexZBoost		-0.171	-0.244	-0.133	0.880	1.008	1.198	-1.942	-2.052	-2.339	0.291	0.315	0.403
Mixture Density Network		-0.235	-0.360	-0.198	0.705	0.824	1.031	-1.613	-1.461	-1.886	0.213	0.238	0.281
Nadaraya-Watson	1.6	-0.258	-0.452	-0.230	0.581	0.483	0.745	-1.508	-1.051	-1.731	0.166	0.138	0.209
Lanne et al. (2012)		-0.247	-0.381	-0.205	0.593	0.491	0.762	-1.451	-1.197	-1.831	0.166	0.135	0.212
Gourieroux and Jasiak (2025)		-0.078	-0.216	-0.050	0.756	0.630	0.971	-1.859	-1.471	-2.235	0.257	0.237	0.325
FlexZBoost		-0.271	-0.431	-0.219	0.595	0.518	0.784	-1.579	-1.112	-1.823	0.171	0.153	0.223
Mixture Density Network		-0.283	-0.422	-0.229	0.581	0.527	0.764	-1.387	-1.129	-1.678	0.164	0.224	0.215
Nadaraya-Watson	1.8	-0.342	-0.599	-0.265	0.485	0.322	0.606	-1.207	-0.725	-1.487	0.138	0.102	0.174
Lanne et al. (2012)		-0.326	-0.421	-0.246	0.487	0.330	0.601	-1.210	-0.882	-1.506	0.134	0.092	0.167
Gourieroux and Jasiak (2025)		-0.185	-0.501	-0.143	0.565	0.355	0.696	-1.528	-0.826	-1.841	0.173	0.128	0.231
FlexZBoost		-0.313	-0.521	-0.240	0.502	0.372	0.633	-1.771	-2.097	-2.226	0.141	0.140	0.187
Mixture Density Network		-0.332	-0.594	-0.263	0.487	0.335	0.603	-1.214	-0.742	-1.507	0.137	0.122	0.175

Notes: This table reports density forecast performance metrics for different tail index values (α). CDE Loss (Conditional Density Estimation loss), CRPS (Continuous Ranked Probability Score), and QS 10% (Quantile Score at 10% level) are loss functions where lower values indicate better performance. Log Prob (Log Probability Score) is a scoring rule where higher values indicate better performance. Metrics are evaluated over three spatial regions: Center $[q_{0.1}, q_{0.9}]$, Tails $[q_{0.01}, q_{0.1}] \cup [q_{0.9}, q_{0.99}]$, and Total $[q_{0.01}, q_{0.99}]$, where q_p represents the p-th quantile. Best method in red, second best in **bold black**.

Table 24: Density Forecast Performance Metrics: MAR(0, 1) Process, 5-Step-Ahead Forecasts

Model	α	CDE Loss			CRPS			Log Prob			QS 10%		
		Center	Tails	Total	Center	Tails	Total	Center	Tails	Total	Center	Tails	Total
Nadaraya-Watson	1.0	-0.027	0.212	-0.010	2.553	10.71	4.790	-3.672	-11.26	-5.044	0.818	2.788	1.383
Lanne et al. (2012)		0.044	0.032	0.023	4.859	17.90	13.64	-4.105	-6.267	-6.171	1.381	3.430	3.658
Gourieroux and Jasiak (2025)		-0.038	0.200	0.162	4.068	27.80	26.86	-3.421	-7.376	-5.923	1.493	20.86	6.577
FlexZBoost		-0.033	-0.013	-0.010	3.636	14.42	11.84	-3.410	-4.571	-4.782	1.170	5.573	5.599
Mixture Density Network		-0.061	-0.041	-0.050	2.612	8.881	5.124	-2.969	-3.675	-3.338	0.890	1.969	1.461
Nadaraya-Watson	1.2	-0.086	-0.031	-0.071	1.454	3.890	2.741	-2.622	-5.334	-3.451	0.491	1.120	0.725
Lanne et al. (2012)		-0.041	-0.027	-0.039	1.928	4.377	2.940	-2.769	-3.693	-3.408	0.555	0.988	0.956
Gourieroux and Jasiak (2025)		0.023	0.083	0.144	3.089	5.384	7.136	-3.357	-4.305	-4.937	1.338	2.798	2.530
FlexZBoost		-0.077	-0.069	-0.045	1.773	4.070	3.270	-2.614	-3.171	-3.337	0.561	1.235	1.217
Mixture Density Network		-0.117	-0.078	-0.088	1.391	4.003	2.620	-2.320	-2.939	-2.755	0.454	0.897	0.745
Nadaraya-Watson	1.4	-0.150	-0.215	-0.110	0.968	1.602	1.622	-2.066	-2.385	-2.615	0.287	0.422	0.487
Lanne et al. (2012)		-0.119	-0.176	-0.092	1.104	1.668	1.610	-2.088	-3.155	-2.805	0.309	0.455	0.515
Gourieroux and Jasiak (2025)		0.042	-0.045	0.111	1.842	2.326	3.285	-2.931	-2.793	-4.081	0.834	1.234	1.388
FlexZBoost		-0.149	-0.221	-0.105	1.034	1.599	1.698	-2.018	-2.417	-2.569	0.315	0.503	0.521
Mixture Density Network		-0.170	-0.195	-0.118	0.959	1.785	1.642	-1.908	-2.079	-2.397	0.284	0.507	0.508
Nadaraya-Watson	1.6	-0.208	-0.291	-0.150	0.756	0.847	1.144	-1.661	-1.531	-2.135	0.220	0.249	0.369
Lanne et al. (2012)		-0.148	-0.296	-0.113	0.832	0.927	1.147	-1.818	-1.612	-2.323	0.223	0.257	0.365
Gourieroux and Jasiak (2025)		0.060	-0.015	0.122	1.312	1.318	2.103	-2.613	-2.147	-3.568	0.590	0.592	0.944
FlexZBoost		-0.204	-0.300	-0.141	0.784	0.972	1.179	-1.865	-3.097	-2.622	0.220	0.369	0.401
Mixture Density Network		-0.216	-0.315	-0.150	0.760	0.913	1.136	-1.637	-1.472	-2.103	0.217	0.356	0.375
Nadaraya-Watson	1.8	-0.267	-0.436	-0.182	0.604	0.586	0.876	-1.403	-0.998	-1.865	0.168	0.195	0.289
Lanne et al. (2012)		-0.204	-0.223	-0.136	0.634	0.596	0.875	-1.518	-1.661	-2.179	0.165	0.177	0.280
Gourieroux and Jasiak (2025)		-0.042	-0.131	0.024	0.855	0.768	1.246	-2.073	-1.262	-2.814	0.347	0.363	0.597
FlexZBoost		-0.244	-0.261	-0.155	0.638	0.867	0.908	-1.812	-4.552	-3.066	0.177	0.350	0.308
Mixture Density Network		-0.269	-0.440	-0.181	0.611	0.627	0.876	-1.411	-1.029	-1.884	0.169	0.240	0.291

Notes: For details on variable definitions and methodology, refer to Table 23.

4 Empirical Applications

4.1 Forecasting Natural Gas Prices in Real Time

Table 25: MSPE Ratios Relative to the No-Change Forecast

Horizon	Nadaraya-Watson	Gourieroux and Jasiak (2025)	Lanne et al. (2012)	FlexZBoost	Mixture Density Network
1	1.099	1.083	1.990	**1.081**	**0.928**
3	1.047	1.237	1.781	**0.923**	**0.878**
6	1.032	1.340	2.221	**0.959**	**0.799**
9	0.951	1.557	2.798	**0.659**	**0.746**
12	0.836	1.683	3.290	**0.693**	**0.778**
15	**0.717**	2.325	3.615	**0.755**	0.808
18	0.819	2.769	3.427	**0.806**	**0.805**
21	**0.861**	2.983	3.528	0.927	**0.813**
24	**0.897**	2.922	3.599	0.927	**0.825**

Values below 1 indicate improvements relative to the no-change forecast. Best method in red, second best in **bold black**.

Table 26: Diebold-Mariano Tests: MDN vs. No-change Forecast for the Real Henry Hub Spot Price

Horizon	p-value
1	0.3118
3	0.0292**
6	0.0664*
9	0.1496
12	0.2410
15	0.2011
18	0.0613*
21	0.0909*
24	0.1189

Note: *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

4.2 Forecasting Inflation in Real Time

Table 27: Estimated MAR(0,2) parameters for US CPI inflation

Parameter	Real-Time In-Sample	Full Period Post-Revised
ψ_1	0.444*** (0.041)	0.440*** (0.041)
ψ_2	0.226*** (0.041)	0.247*** (0.041)
α	1.910*** (0.139)	1.902*** (0.139)
β	0.0009 (0.128)	0.002 (0.128)
σ	0.141*** (0.009)	0.136*** (0.009)

Note: Standard errors in parentheses. *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

Table 28: Diebold-Mariano Tests: MDN vs. No-change Forecast for U.S. Inflation

Horizon	p-value
1	0.0000***
2	0.0000***
3	0.0000***
4	0.0002***
5	0.0011***
6	0.0022***
7	0.0003***
8	0.0001***
9	0.0003***
10	0.0050***
11	0.0393**
12	0.0010***

Note: *** $p < 0.01$, ** $p < 0.05$, * $p < 0.10$.

References

Chen, T. & Guestrin, C. (2016), XGBoost: A scalable tree boosting system, in 'Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 785–794.

Dalmasso, N., Pospisil, T., Lee, A. B., Izbicki, R., Freeman, P. E. & Malz, A. I. (2020), 'Conditional density estimation tools in python and R with applications to photometric redshifts and likelihood-free cosmological inference', Astronomy and Computing **30**, 100362.

de Truchis, G., Fries, S. & Thomas, A. (2025), 'Forecasting extreme trajectories using seminorm representations'. Working paper.

Fries, S. (2022), 'Conditional moments of noncausal alpha-stable processes and the prediction of bubble crash odds', Journal of Business & Economic Statistics **40**(4), 1596–1616.

Gourieroux, C. & Jasiak, J. (2025), 'Nonlinear fore(back)casting and innovation filtering for causal-noncausal VAR models'.

Gourieroux, C., Lu, Y. & Robert, C.-Y. (2025), 'The causal-noncausal tail processes'.

Gourieroux, C. & Zakoian, J.-M. (2017), 'Local explosion modelling by non-causal process', Journal of the Royal Statistical Society Series B **79**(3), 737–756.

Hansen, P. R., Lunde, A. & Nason, J. M. (2011), 'The model confidence set', Econometrica **79**(2), 453–497.

Izbicki, R. (2017), 'Converting high-dimensional regression to high-dimensional conditional density estimation', Electronic Journal of Statistics **11**(2), 2800–2831.

Lanne, M., Luoto, J. & Saikkonen, P. (2012), 'Optimal forecasting of noncausal autoregressive time series', International Journal of Forecasting **28**(3), 623–631.

Nadaraya, E. A. (1964), 'On estimating regression', Theory of Probability & Its Applications **9**(1), 141–142.

Nolan, J. P. (2020), Univariate Stable Distributions: Models for Heavy Tailed Data, Springer Series in Operations Research and Financial Engineering, Springer, Cham.

Rosenblatt, M. (1969), Conditional probability density and regression estimators, in P. R. Krishnaiah, ed., 'Multivariate Analysis II', Academic Press, pp. 25–31.

Rosenblatt, M. (2000), Gaussian and Non-Gaussian Linear Time Series and Random Fields, Springer.

Samorodnitsky, G. & Taqqu, M. S. (1996), 'Stable non-gaussian random processes: Stochastic models with infinite variance', Bulletin of the London Mathematical Society **28**(5), 554–555.

Watson, G. S. (1964), 'Smooth regression analysis', Sankhyā: The Indian Journal of Statistics, Series A **26**(4), 359–372.