Galerkin-ARIMA: A Two-Stage Polynomial Regression Framework for Fast Rolling One-Step-Ahead Forecasting

Haojie Liu
Department of Economics
University of California, Riverside
hliu332@ucr.edu

Zihan Lin
Department of Economics
University of California, Riverside
zlin169@ucr.edu

Randall R. Rojas
Department of Economics
University of California, Los Angeles
rrojas@econ.ucla.edu

Abstract

We present Galerkin-ARIMA, an innovative time-series forecasting framework that fuses Galerkin projection methods—commonly used in solving differential equations—with the foundational ARIMA model. This integration enables the capture of nonlinear relationships in lagged observations by substituting the traditional linear autoregressive term with a polynomial basis expansion, estimated through ordinary least squares. The framework preserves ARIMA's moving-average component and Gaussian error assumptions, yielding closed-form solutions via a two-stage projection process. We establish theoretical guarantees, including asymptotic unbiasedness and consistency, alongside analyses of bias-variance trade-offs and computational complexity, demonstrating significant efficiency gains over maximum-likelihood estimation, particularly with moderate basis sizes and ridge regularization for stability. Empirical evaluations on diverse synthetic datasets—encompassing noisy ARMA, seasonal patterns, trend-autoregressive, and nonlinear recursions—reveal that Galerkin-ARIMA achieves comparable or superior forecasting precision to standard ARIMA while delivering speedups of several orders of magnitude in rolling one-step-ahead predictions. These findings position Galerkin-ARIMA as a versatile, high-performance tool for tackling intricate time-series challenges in data-intensive and real-time environments.

1 Introduction

Time-series forecasting plays a critical role in economics, finance, energy demand, weather prediction, and more. It turns historical data into proactive insight to understand temporal patterns. The Autoregressive Integrated Moving Average (ARIMA) model remains one of the most widely used time-series models today [Vuong et al., 2024].

However, as the volume and complexity of temporal data continue to grow, ARIMA becomes increasingly slow during forecasting—the more temporal data, the longer it takes [Wang et al., 2023]. Its speed is insufficient for handling the growing complexity of modern time series. Moreover, ARIMA assumes a fixed linear relationship among past values, which can be overly restrictive when the true dynamics are nonlinear or complex [Thupeng et al., 2024, Vuong et al., 2024].

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

Partial differential equations (PDEs) and stochastic differential equations (SDEs) are widely used in statistics to describe continuous-time systems with uncertainty [Graham et al., 2023]. They appear in many areas, such as asset pricing in finance or diffusion-based sampling in Bayesian inference [Dokuchaev et al., 2022]. Galerkin projection is usually used to solve the problems, which is a numerical method that approximates complex functions using basis expansions [Dokuchaev et al., 2022]. We apply this idea to time series forecasting in a discrete setting.

To address this, we propose Galerkin-ARIMA, which uses the Galerkin method to replace the linear autoregressive component of the ARIMA model with a polynomial basis expansion (with splines as an extension) learned via Galerkin projection [Minsong Gao, 2022]. Specifically, we approximate the AR term using a set of polynomial basis functions (e.g., linear and quadratic terms, with splines optional) over the lag space, allowing the model to flexibly capture nonlinear dependencies [Kushnir and Tokarieva, 2023]. The MA term and Gaussian noise assumptions of ARIMA are retained.

We show that under regular conditions, the Galerkin-ARIMA model is asymptotically consistent and unbiased, and it can be more computationally efficient than traditional maximum likelihood estimation methods commonly used for ARIMA models. Our approach combines the structure of ARIMA with the flexibility of nonparametric regression and offers a new path forward for modeling complex time series dynamics—achieved through a closed form solution based on Ordinary Least Squares (OLS).

2 ARIMA

The *Autoregressive Integrated Moving Average* (ARIMA) model is a fundamental time-series model that combines autoregressive (AR) terms, integration (differencing), and moving-average (MA) terms [Box and Jenkins, 1970, Brockwell and Davis, 1991]. The ARIMA(p, d, q) model differences the series d times: $y_t^{(d)} = \Delta^d y_t = (1 - B)^d y_t$. For the seasonal extension, SARIMA$(p, d, q) \times (P, D, Q)_m$, an additional seasonal differencing is applied: $y_t^{(d,D)} = (1 - B)^d (1 - B^m)^D y_t$.

Using the back-shift operator B (defined by $By_t = y_{t-1}$), we can write the differenced series as

$$y_t^{(d)} \;=\; \Delta^d y_t \;=\; (1 - B)^d \, y_t, \tag{1}$$

where $y_t^{(d)}$ denotes the d-th-order differenced series. We then define the p-dimensional state (or lag) vector

$$x_t \;=\; \big(y_{t-1}^{(d)},\, y_{t-2}^{(d)},\, \ldots,\, y_{t-p}^{(d)}\big) \;\in\; \mathbb{R}^p, \tag{2}$$

which collects the last p observed values of the differenced series. The AR part of the model is a linear combination of these lagged values,

$$m_t \;=\; \sum_{i=1}^{p} \psi_i \, y_{t-i}^{(d)}, \tag{3}$$

where ψ_i are the autoregressive coefficients. Random shocks $\{\epsilon_t\}$ are assumed independent and identically distributed (i.i.d.) Gaussian with mean 0 and variance σ^2, i.e. $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$. The full ARIMA$(p, d, q)$ model can then be written in observation form as

$$y_t^{(d)} \;=\; m_t \;+\; \sum_{j=1}^{q} \theta_j \, \epsilon_{t-j} \;+\; \epsilon_t, \tag{4}$$

meaning the differenced series $y_t^{(d)}$ is explained by the AR term m_t and a moving-average term (a linear combination of the last q innovations) plus the current innovation.

Compact operator form (true SARIMA). Collecting the non-seasonal and seasonal factors, the *true* Gaussian SARIMA$(p, d, q) \times (P, D, Q)_m$ data-generating process is

$$\Phi(B)\, \Phi_s\big(B^m\big)\, y_t^{(d,D)} \;=\; \Theta(B)\, \Theta_s\big(B^m\big)\, \epsilon_t, \qquad \epsilon_t \overset{iid}{\sim} \mathcal{N}\big(0, \sigma^2\big), \tag{5}$$

where $\Phi(B) = 1 - \sum_{i=1}^{p} \phi_i B^i$ and $\Theta(B) = 1 + \sum_{j=1}^{q} \theta_j B^j$ are the non-seasonal AR and MA operators, while $\Phi_s(B^m) = 1 - \sum_{i=1}^{P} \Phi_i B^{im}$ and $\Theta_s(B^m) = 1 + \sum_{j=1}^{Q} \Theta_j B^{jm}$ encode the seasonal components

with period m. Equation (5) will serve as the reference model in our asymptotic analysis (see Appendix 8.1).

2.1 Maximum Likelihood Estimation (MLE)

The parameters of an ARIMA model are typically estimated by maximizing the likelihood of the observed data under the model [Brockwell and Davis, 1991]. For a Gaussian ARIMA model, maximizing the likelihood is equivalent to minimizing the sum of squared one-step-ahead forecast errors. Let $\Psi = (\psi_1, \ldots, \psi_p,\ \theta_1, \ldots, \theta_q,\ \sigma^2)$ collect all unknown parameters. The one-step prediction error at time t (for given parameters Ψ) is

$$\epsilon_t(\Psi) \;=\; y_t^{(d)} \;-\; \sum_{i=1}^{p} \psi_i \, y_{t-i}^{(d)} \;-\; \sum_{j=1}^{q} \theta_j \, \epsilon_{t-j}(\Psi). \tag{6}$$

The (conditional) Gaussian log-likelihood of the data (up to an additive constant) is

$$\mathcal{L}(\Psi) \;=\; -\frac{N}{2} \ln \sigma^2 \;-\; \frac{1}{2\sigma^2} \sum_{t=1}^{N} \epsilon_t(\Psi)^2, \tag{7}$$

where N is the number of observations after differencing. Maximizing $\mathcal{L}(\Psi)$ is equivalent to minimizing $\sum_t \epsilon_t(\Psi)^2$. The maximum-likelihood estimator (MLE) therefore satisfies

$$\hat{\Psi} \;=\; \arg\max_{\Psi} \, \mathcal{L}(\Psi) \;=\; \arg\min_{\Psi} \sum_{t=1}^{N} \epsilon_t(\Psi)^2. \tag{8}$$

Because this optimisation has no closed-form solution for general ARMA models, it is usually solved numerically (e.g. Newton–Raphson, BFGS, or other gradient-based methods) [Brockwell and Davis, 1991].

In summary, classical ARIMA modelling involves finding linear coefficients (ψ_i, θ_j) that minimise prediction error. Next, we introduce a Galerkin approximation that generalises the AR component to a flexible function (polynomial or spline basis) while retaining the MA structure—yielding the proposed Galerkin-ARIMA framework.

3 Galerkin Approximation

Before introducing the Galerkin-ARIMA model, we briefly review the Galerkin method in the context of stochastic differential equations (SDEs). Consider a one–dimensional SDE:

$$dX_t = b(X_t)\, dt + \sigma(X_t)\, dW_t \,, \tag{9}$$

where $b(x)$ is the drift function and $\sigma(x)$ is the diffusion coefficient. Suppose the drift function $b(x)$ is expensive to compute or is unknown; we wish to approximate $b(x)$ using a finite–dimensional function space spanned by a set of basis functions. Let $\{\phi_j(x)\}_{j=1}^{K}$ be a family of basis functions (for example, B-splines of degree p on the domain Ω). We postulate an approximating drift of the form:

$$b_K(x) \;=\; \sum_{j=1}^{K} \theta_j \, \phi_j(x) \,, \tag{10}$$

with unknown coefficients $\theta = (\theta_1, \ldots, \theta_K)$ to be determined by a Galerkin projection.

The Galerkin method requires that the residual (error) between the true drift and the approximated drift is orthogonal to the space spanned by the basis. Define the residual function $r(x) = b(x) - b_K(x)$. We introduce an inner product on the space of functions (weighted by a probability measure $\mu(x)$ on Ω):

$$\langle f, g \rangle = \int_{\Omega} f(x)\, g(x)\, \mu(x)\, dx \,. \tag{11}$$

The Galerkin orthogonality condition imposes that the residual is orthogonal to each basis function ϕ_i:

$$\langle r, \phi_i \rangle = 0, \qquad i = 1, \ldots, K. \tag{12}$$

Substituting $r(x) = b(x) - \sum_{j=1}^{K} \theta_j \phi_j(x)$, this condition becomes:

$$\int_{\Omega} \Big(b(x) - \sum_{j=1}^{K} \theta_j \, \phi_j(x) \Big) \, \phi_i(x) \, \mu(x) \, dx = 0, \quad \text{for each } i = 1, \ldots, K. \tag{13}$$

Define the Gram matrix $G \in \mathbb{R}^{K \times K}$ and the vector $f \in \mathbb{R}^K$ by:

$$G_{ij} = \int_{\Omega} \phi_i(x) \, \phi_j(x) \, \mu(x) \, dx, \qquad f_i = \int_{\Omega} b(x) \, \phi_i(x) \, \mu(x) \, dx. \tag{14}$$

The Galerkin conditions then yield a linear system $G\,\theta = f$. Once solved for θ, we obtain the Galerkin approximation $b_K(x) = \sum_{j=1}^{K} \theta_j \phi_j(x)$. Substituting this back into the SDE gives:

$$dX_t = b_K(X_t) \, dt + \sigma(X_t) \, dW_t \, , \tag{15}$$

which is a spline-based approximation of the original dynamics.

3.1 AR-term Approximation via Galerkin Method

We now return to time-series modeling. The key idea of *Galerkin-ARIMA* is to replace the fixed linear AR structure with a more flexible function approximated by basis functions, while still retaining the MA component. Essentially, we treat the p-dimensional lag vector as an input to an unknown function $f(\cdot)$ (analogous to a drift function in an SDE), and we approximate this function using a finite basis expansion.

Let $\Phi(x_t) = [\phi_1(x_t), \, \phi_2(x_t), \, \ldots, \, \phi_K(x_t)]^\top \in \mathbb{R}^K$ be a vector of K basis function evaluations at the state $x_t = (y_{t-1}^{(d)}, \ldots, y_{t-p}^{(d)})$. For example, each ϕ_j could be a multi-variate B-spline or some polynomial basis defined on the lag space. In our implementation, we use a polynomial basis by default (constant, linear, and quadratic terms on lags), as it provides a simple yet effective approximation. Splines (e.g., B-splines) can be used for smoother approximations, and the Jackson bounds derived below apply to both under appropriate smoothness conditions. We then posit that the AR component m_t can be represented as:

$$m_t = \Phi(x_t)^\top \, \beta \, , \qquad \beta \in \mathbb{R}^K, \tag{16}$$

where $\beta = (\beta_1, \ldots, \beta_K)^\top$ are the coefficients of the basis expansion. In other words, m_t is the Galerkin approximation of the potentially nonlinear relationship between past values and the next differenced value $y_t^{(d)}$. The Galerkin-ARIMA model then becomes:

$$y_t^{(d)} = m_t + \sum_{j=1}^{q} \theta_j \, \epsilon_{t-j} + \epsilon_t \, , \tag{17}$$

similar in form to the classical ARIMA, but with m_t now given by a flexible spline-based function rather than a fixed linear combination of lags.

Estimating the coefficients in this Galerkin framework can be done in closed form using least squares, thanks to the linearity in the parameters β. First, we fit the basis-function coefficients β to approximate the AR part. This can be done by regressing $y_t^{(d)}$ onto $\Phi(x_t)$. In matrix terms, let $Y = [y_1^{(d)}, y_2^{(d)}, \ldots, y_N^{(d)}]^\top$ be the vector of responses (differenced series), and let Φ be the $N \times K$ matrix whose t-th row is $\Phi(x_t)^\top$. Then the ordinary least squares (OLS) solution for β is:

$$\hat{\beta} = (\Phi^\top \Phi)^{-1} \, \Phi^\top Y \, , \tag{18}$$

provided $N > K$ and the basis functions are linearly independent. This $\hat{\beta}$ minimizes the sum of squared errors $\sum_t (y_t^{(d)} - \Phi(x_t)^\top \beta)^2$. Using $\hat{\beta}$, we can compute the fitted residuals (initially ignoring the MA terms) as:

$$\hat{\epsilon}_t^{(0)} = y_t^{(d)} - \Phi(x_t)^\top \hat{\beta} \, , \tag{19}$$

for each t. These residuals $\hat{\epsilon}_t^{(0)}$ represent the remaining structure after accounting for the spline-based AR approximation.

Next, to estimate the moving-average coefficients $\theta_1, \ldots, \theta_q$, we include the lagged residuals in a second regression. For each t, form the vector of the last q residuals $\hat{\epsilon}^{(0)}_{t-1:t-q} = [\hat{\epsilon}^{(0)}_{t-1}, \hat{\epsilon}^{(0)}_{t-2}, \ldots, \hat{\epsilon}^{(0)}_{t-q}]^\top$. Now define the combined regression design matrix Ψ of dimension $N \times (K+q)$ whose t-th row is $[\Phi(x_t)^\top \mid \hat{\epsilon}^{(0)\,T}_{t-1:t-q}]$. In other words, $\Psi_t = [\phi_1(x_t), \ldots, \phi_K(x_t) \mid \hat{\epsilon}^{(0)}_{t-1}, \ldots, \hat{\epsilon}^{(0)}_{t-q}]$ includes both the basis functions and the lagged residuals. Let $\gamma = (\beta^\top, \theta_1, \ldots, \theta_q)^\top$ be the combined coefficient vector of length $K+q$. Then an OLS estimate for γ is given by:

$$\hat{\gamma} = (\Psi^\top \Psi)^{-1}\, \Psi^\top Y\,. \tag{20}$$

This yields refined estimates $\hat{\beta}$ (the first K entries of $\hat{\gamma}$) and $\hat{\theta}_1, \ldots, \hat{\theta}_q$ (the last q entries of $\hat{\gamma}$) simultaneously. Once we have these estimates, we can form the final fitted model:

$$\hat{m}_t = \Phi(x_t)^\top \hat{\beta}\,, \tag{21}$$

$$\hat{y}^{(d)}_{t+1} = \hat{m}_{t+1} + \sum_{j=1}^{q} \hat{\theta}_j\, \hat{\epsilon}_{t+1-j}\,, \tag{22}$$

which provides one-step-ahead forecasts for the differenced series. (Here $\hat{\epsilon}_{t+1-j}$ are the in-sample residuals at time $t+1-j$.) In essence, $\hat{m}_t$ is a nonparametric (spline-based) estimate of the AR part, and the $\hat{\theta}_j$ adjust for the remaining serial correlation (the MA part).

Under standard conditions for nonparametric regression (in particular, as $K \to \infty$ and $K/N \to 0$ as $N \to \infty$), the spline approximation $m_t = \Phi(x_t)^\top \beta$ can approximate any sufficiently smooth function of the lags arbitrarily well. This suggests that as we increase the number of basis functions K, the Galerkin-ARIMA model can in principle capture the true underlying relationship among the past values arbitrarily well.

3.2 Seasonal-AR-term Approximation via Galerkin Method

To capture the repeating seasonal effect every m periods, we now approximate the seasonal autoregressive component—i.e. the dependence of $y^{(d,D)}_t$ on its values one season ago, two seasons ago, etc.—using a basis expansion. Define the seasonal lag-vector

$$x^{(s)}_t = \big(y^{(d,D)}_{t-m},\, y^{(d,D)}_{t-2m},\, \ldots,\, y^{(d,D)}_{t-Pm}\big) \in \mathbb{R}^P, \tag{23}$$

and choose K_s smooth basis functions

$$\Phi^{(s)}\big(x^{(s)}_t\big) = \big[\phi^{(s)}_1(x^{(s)}_t),\, \ldots,\, \phi^{(s)}_{K_s}(x^{(s)}_t)\big]^T \in \mathbb{R}^{K_s}. \tag{24}$$

We then posit a Galerkin representation of the seasonal-AR term:

$$m^{(s)}_t = \Phi^{(s)}\big(x^{(s)}_t\big)^T \beta^{(s)}, \qquad \beta^{(s)} \in \mathbb{R}^{K_s}. \tag{25}$$

By regressing the differenced series $Y = [y^{(d,D)}_1, \ldots, y^{(d,D)}_N]^T$ onto these basis evaluations, the OLS solution

$$\widehat{\beta}^{(s)} = \big(\Phi^{(s)\,T}\, \Phi^{(s)}\big)^{-1}\, \Phi^{(s)\,T}\, Y \tag{26}$$

minimizes the seasonal-only sum of squared error $\sum_t \big(y^{(d,D)}_t - \Phi^{(s)}(x^{(s)}_t)^T \beta^{(s)}\big)^2$.

3.3 MA-term Approximation via Galerkin Method

Next, we turn to approximating the MA component by the same Galerkin-projection idea, now applied to the sequence of residuals. Define the q-dimensional residual-lag vector

$$r_t \;=\; \big(\hat{\epsilon}^{(0)}_{t-1},\, \hat{\epsilon}^{(0)}_{t-2},\, \ldots,\, \hat{\epsilon}^{(0)}_{t-q}\big) \in \mathbb{R}^q. \tag{27}$$

Choose a family of L basis functions

$$\Psi(r_t) \;=\; \big[\psi_1(r_t),\, \psi_2(r_t), \ldots, \psi_L(r_t)\big]^\top \;\in \mathbb{R}^L, \tag{28}$$

for instance linear and quadratic terms in each residual lag or spline basis on $\mathbb{R}^q$. We then posit

$$\epsilon_t \;\approx\; G(r_t) \;=\; \Psi(r_t)^\top \alpha, \qquad \alpha \in \mathbb{R}^L, \tag{29}$$

where Ψ can include nonlinear terms (e.g., quadratic in residuals) to approximate potentially nonlinear MA dependencies, extending the linear assumption of classical ARIMA. Galerkin orthogonality requires the residual of this approximation to be orthogonal to each basis:

$$\sum_{t=q+1}^{N} \left(\hat{\epsilon}_t^{(0)} - \Psi(r_t)^\top \alpha\right) \psi_i(r_t) \;=\; 0, \quad i = 1, \ldots, L. \tag{30}$$

Define the Gram matrix $M \in \mathbb{R}^{L\times L}$ and load vector $b \in \mathbb{R}^L$ by

$$M_{ij} \;=\; \sum_{t=q+1}^{N} \psi_i(r_t)\, \psi_j(r_t), \qquad b_i \;=\; \sum_{t=q+1}^{N} \hat{\epsilon}_t^{(0)}\, \psi_i(r_t). \tag{31}$$

Solving $M\alpha = b$ yields the Galerkin estimate $\hat{\alpha}$. Finally, for one-step forecasting we evaluate

$$\hat{\epsilon}_{t+1} \;=\; \Psi\big(r_{t+1}\big)^\top \hat{\alpha}, \tag{32}$$

and the full one-step-ahead forecast is

$$\hat{y}_{t+1}^{(d)} \;=\; \Phi(x_{t+1})^\top \hat{\beta} \;+\; \hat{\epsilon}_{t+1}. \tag{33}$$

3.4 Seasonal-MA-term Approximation via Galerkin Method

After subtracting both non-seasonal and seasonal AR fits ($\hat{m}_t$ and $\hat{m}_t^{(s)}$), the residuals $\hat{\epsilon}_t^{(1)}$ still contain seasonal serial correlation. We approximate this seasonal-MA component by projecting the sequence of seasonal-lagged residuals onto another basis. Define

$$r_t^{(s)} = \big(\hat{\epsilon}_{t-m}^{(1)},\, \hat{\epsilon}_{t-2m}^{(1)},\, \ldots,\, \hat{\epsilon}_{t-Qm}^{(1)}\big) \in \mathbb{R}^Q, \tag{34}$$

and choose L_s basis functions

$$\Psi^{(s)}\big(r_t^{(s)}\big) = \big[\psi_1^{(s)}(r_t^{(s)}),\, \ldots,\, \psi_{L_s}^{(s)}(r_t^{(s)})\big]^T \in \mathbb{R}^{L_s}, \tag{35}$$

where $\Psi^{(s)}$ can include nonlinear terms (e.g., quadratic in residuals) to approximate potentially nonlinear MA dependencies, extending the linear assumption of classical ARIMA. Galerkin orthogonality requires

$$\sum_{t=q+1}^{N} \left(\hat{\epsilon}_t^{(1)} - \Psi^{(s)}(r_t^{(s)})^T\, \alpha^{(s)}\right) \psi_i^{(s)}(r_t^{(s)}) = 0, \quad i = 1, \ldots, L_s, \tag{36}$$

leading to the normal equations

$$M^{(s)}\, \alpha^{(s)} = b^{(s)}, \quad M_{ij}^{(s)} = \sum_{t=q+1}^{N} \psi_i^{(s)}(r_t^{(s)})\, \psi_j^{(s)}(r_t^{(s)}), \quad b_i^{(s)} = \sum_{t=q+1}^{N} \hat{\epsilon}_t^{(1)}\, \psi_i^{(s)}(r_t^{(s)}). \tag{37}$$

Solving for $\hat{\alpha}^{(s)}$ gives the seasonal-MA approximation, and the one-step seasonal correction

$$\widehat{\epsilon}_{t+1}^{(s)} = \Psi^{(s)}\big(r_{t+1}^{(s)}\big)^T\, \hat{\alpha}^{(s)}. \tag{38}$$

Thus the full one-step-ahead forecast combines non-seasonal AR, seasonal AR, non-seasonal MA, and seasonal MA projections:

$$\hat{y}_{t+1}^{(d,D)} = \hat{m}_{t+1} + \hat{m}_{t+1}^{(s)} + \widehat{\epsilon}_{t+1} + \widehat{\epsilon}_{t+1}^{(s)}. \tag{39}$$

4 Error Analysis with Seasonal Components

In this section we extend the bias, variance and complexity analysis to the full *seasonal* Galerkin-ARIMA model. Denote the regular orders by (p, d, q) and the seasonal orders by (P, D, Q) with period m; the notation follows the convention of Box and Jenkins [1970]. Let

$$y_t^{(d,D)} \;=\; (1-B)^d (1-B^m)^D y_t$$

be the series after non-seasonal and seasonal differencing. Writing

$$x_t = \big(y_{t-1}^{(d,D)}, \ldots, y_{t-p}^{(d,D)}\big), \quad x_t^{(s)} = \big(y_{t-m}^{(d,D)}, \ldots, y_{t-Pm}^{(d,D)}\big),$$

the **Galerkin-SARIMA** forecast combines four projections:

$$\hat{y}_{t+1}^{(d,D)} \;=\; \Phi(x_{t+1})^T \hat{\beta} \;+\; \Phi^{(s)}(x_{t+1}^{(s)})^T \hat{\beta}^{(s)} \;+\; \Psi(r_{t+1})^T \hat{\alpha} \;+\; \Psi^{(s)}(r_{t+1}^{(s)})^T \hat{\alpha}^{(s)}. \tag{40}$$

4.1 Bias

Theorem 4.1 (Asymptotic unbiasedness)**.** *Assume the data-generating process in* (5) *and the regularity conditions collected in Appendix 8.1. Then*

$$\left|\mathbb{E}\big[\hat{y}_{t+1}^{(d,D)} - y_{t+1}^{(d,D)}\big]\right| \xrightarrow[N\to\infty]{} 0. \tag{41}$$

Under the Gaussian SARIMA model

$$\Phi(B)\,\Phi_s(B^m)\,y_t^{(d,D)} \;=\; \Theta(B)\,\Theta_s(B^m)\,\epsilon_t, \quad \epsilon_t \overset{iid}{\sim} \mathcal{N}(0,\sigma^2), \tag{42}$$

the maximum-likelihood estimator is known to be asymptotically unbiased. For Galerkin-SARIMA we decompose the one-step error

$$\hat{y}_{t+1}^{(d,D)} - y_{t+1}^{(d,D)} = \sum_{j=1}^{4} E_j, \qquad E_1 = f - \Phi\hat{\beta},\; E_2 = f^{(s)} - \Phi^{(s)}\hat{\beta}^{(s)},\; E_3 = g - \Psi\hat{\alpha},\; E_4 = g^{(s)} - \Psi^{(s)}\hat{\alpha}^{(s)}, \tag{43}$$

and apply the Jackson bounds

$$\sup_x |f(x) - \Phi(x)^T\beta| \;=\; O\big(K^{-r/p}\big), \qquad \sup_{x^{(s)}} |f^{(s)}(x^{(s)}) - \Phi^{(s)}(x^{(s)})^T\beta^{(s)}| \;=\; O\big(K_s^{-r/P}\big), \tag{44}$$

together with $\mathbb{E}[\hat{\beta}\,|\,\Psi_N] = \beta^\star$ to get

$$\mathbb{E}[E_1] = O(K^{-r/p}), \quad \mathbb{E}[E_2] = O(K_s^{-r/P}), \quad \mathbb{E}[E_3] = O(N^{-1}), \quad \mathbb{E}[E_4] = O(N^{-1}). \tag{45}$$

Basis growth (Assumption 8.1) implies each term vanishes.

4.2 Variance and Consistency

Proposition 4.2 (CLT and consistency)**.** *Let $\hat{\gamma} = (\hat{\beta}, \hat{\beta}^{(s)}, \hat{\alpha}, \hat{\alpha}^{(s)})$ and Ψ_N be the full design matrix. Under the same assumptions,*

$$\sqrt{N}\,(\hat{\gamma} - \gamma^\star) \;\xrightarrow{d}\; \mathcal{N}\big(0,\, \sigma^2(\operatorname*{plim}_{N\to\infty} N^{-1}\Psi_N^T\Psi_N)^{-1}\big), \quad \|\hat{\gamma} - \gamma^\star\|_2 = O_p\big(N^{-1/2}\big). \tag{46}$$

Proposition 4.3 (Optimal MSE)**. 4.3 Optimal Mean-Squared Error**

Choose $K \sim N^{p/(2r+p)}$, $K_s \sim N^{P/(2r+P)}$, $L \sim K$, $L_s \sim K_s$ to balance squared bias and variance. Then

$$\mathrm{MSE} = O\big(N^{-2r/(2r+p)}\big) + O\big(N^{-2r/(2r+P)}\big). \tag{47}$$

Proposition 4.4 (Cost comparison)**. 4.4 Computational Complexity**

Fitting a classical SARIMA by likelihood optimisation costs $\Theta\big(I\,(p+P+q+Q)\,N\big)$ per iteration. Galerkin-SARIMA requires

$$\Theta\big(N(pK + PK_s)\big) \;+\; \Theta\big((K + K_s + L + L_s)^3\big) \tag{48}$$

for one pass of the two OLS stages. When $K+K_s+L+L_s \ll p+P+q+Q$ and the optimiser needs many iterations ($I \gg 1$) the Galerkin cost is asymptotically negligible, confirming the speed-ups in Table 1.

5 Numerical Experiments

5.1 Synthetic Data Generation

All series have length $n = 300$ and are generated with a fixed random seed for reproducibility. We consider four toy processes:

1. **Noisy ARMA(2,1)**:

$$y_t \;=\; 0.6\,y_{t-1} \;-\; 0.3\,y_{t-2} \;+\; 0.5\,\epsilon_{t-1} \;+\; \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0,1), \quad y_1 = y_2 = 0. \tag{49}$$

2. **Seasonal sine + noise**:

$$y_t = \sin\!\Big(2\pi\,\tfrac{t}{20}\Big) \;+\; 0.5\,\eta_t, \quad \eta_t \sim \mathcal{N}(0,1). \tag{50}$$

3. **Linear trend + AR(1)**:

$$y_t = 0.01\,t \;+\; 0.8\,y_{t-1} \;+\; \nu_t, \quad \nu_t \sim \mathcal{N}(0, 0.5^2), \quad y_0 = 0. \tag{51}$$

4. **Nonlinear recursion + noise**:

$$\begin{aligned} &y_t = 3.8\,y_{t-1}\big(1 - y_{t-1}\big) + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, 0.02^2), \\ &y_0 = 0.4, \end{aligned} \tag{52}$$

These four processes span a wide variety of dynamics—linear autoregression with a moving-average component, periodic fluctuations, deterministic trend combined with stochastic feedback, and a purely nonlinear recursion. By testing on data that exhibit both classical ARMA behavior and more challenging seasonal, trending, and nonlinear patterns, we can rigorously evaluate whether Galerkin-ARIMA's flexible basis expansion captures complex dependencies that a standard ARIMA may miss, while still recovering simple linear structure when appropriate.

5.2 Forecasting Setup

We perform *rolling* one-step-ahead forecasts on each series using:

- a fixed training window of length $W = 50$,
- a forecast horizon of $H = 100$, and
- Monte Carlo replications $R = 5$ (to average out noise).

At each forecast origin $i = W, \ldots, W + H - 1$ we refit two models with the *same* seasonal order quadruple $(p, q, P, Q) \in \{0, 1, 5\}^4$ and fixed seasonal period $m = 8$. Model (a) is the classical SARIMA$(p, 0, q) \times (P, 0, Q)_8$, estimated by Gaussian maximum likelihood via `statsmodels.tsa.arima.ARIMA` with `order=(p,0,q)` and `seasonal_order=(P,0,Q,8)`. Model (b) is *Galerkin–SARIMA*(p, q, P, Q), calculated by the two-stage Galerkin procedure outlined below.

Stage 1: Non-seasonal AR projection. Regress y_t on the polynomial basis

$$\big\{1,\; y_{t-1}, \ldots, y_{t-p},\; y_{t-1}^2, \ldots, y_{t-p}^2,\; y_{t-8}, \ldots, y_{t-8P},\; y_{t-8}^2, \ldots, y_{t-8P}^2\big\},$$

obtaining coefficients $\hat{\beta}$ and residuals $\hat{\epsilon}_t^{(0)}$.

Stage 2: MA projection. Regress $\hat{\epsilon}_t^{(0)}$ on

$$\big\{1,\; \hat{\epsilon}_{t-1}^{(0)}, \ldots, \hat{\epsilon}_{t-q}^{(0)},\; \hat{\epsilon}_{t-8}^{(0)}, \ldots, \hat{\epsilon}_{t-8Q}^{(0)},\; (\hat{\epsilon}_{t-1}^{(0)})^2, \ldots, (\hat{\epsilon}_{t-q}^{(0)})^2,\; (\hat{\epsilon}_{t-8}^{(0)})^2, \ldots, (\hat{\epsilon}_{t-8Q}^{(0)})^2\big\},$$

yielding moving-average coefficients $\hat{\alpha}$.

For every (p, q, P, Q) we roll one step ahead over the horizon H, producing forecasts $\{\hat{y}_{i+1}\}_{i=W}^{W+H-1}$. From these we compute the mean absolute error (MAE), root-mean-squared error (RMSE), total CPU time, and average CPU time per fit; each metric is finally averaged over the R Monte-Carlo runs. The aggregated results appear in Section 5.3.

5.3 Result

Figure 1 and Table 1 together present a comprehensive comparison between the Galerkin–SARIMA method and the classical SARIMA approach on four synthetic benchmarks. For each dataset we

Dataset	MAE_G	MAE_S	RMSE_G	RMSE_S	Median t_G	Median t_S	Speed–up ($\times$)
Noisy_ARMA	0.8096	0.7867	1.0049	0.9514	0.0256	35.7401	1 395.3182
Nonlinear	0.0141	0.1736	0.0180	0.2072	0.0245	60.3865	2 464.6143
Seasonal	0.4364	0.4272	0.5447	0.5268	0.0234	46.3963	1 978.7565
Trend_AR	0.4567	0.4545	0.5752	0.5844	0.0235	63.9590	2 724.3840

Table 1: Best accuracy (lower is better) and median runtime for Galerkin–SARIMA (G) versus SARIMA (S).

report two error metrics (MAE and RMSE) and the median runtime of each algorithm, along with the ratio of those runtimes expressed as a speed-up factor.

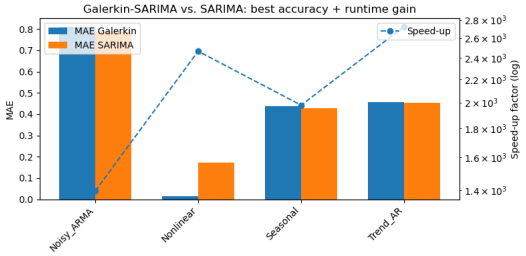

Figure 1: Comparison of forecasting accuracy and runtime between Galerkin–SARIMA and classical SARIMA. Blue and orange bars show the mean absolute error (MAE) of Galerkin–SARIMA and SARIMA respectively on four synthetic datasets. The dashed line (right-hand axis, logarithmic scale) plots the speed-up factor ($\times$) achieved by Galerkin–SARIMA in median runtime.

On the Noisy ARMA series, both methods yield very similar predictive errors: MAE of 0.8096 for Galerkin–SARIMA versus 0.7867 for SARIMA, and RMSE of 1.0049 versus 0.9514. These differences represent under a 3% change in absolute error. However, the computational cost diverges dramatically: median runtime drops from 35.74s for SARIMA to just 0.0256s for Galerkin–SARIMA, corresponding to a 1395 times acceleration. This demonstrates that the Galerkin projection preserves classical SARIMA's statistical performance while slashing computation time by three orders of magnitude.

The Nonlinear dataset further amplifies the advantage: Galerkin–SARIMA not only achieves over an order of magnitude lower MAE (0.0141 vs. 0.1736) and RMSE (0.0180 vs. 0.2072), but also executes in 0.0245s instead of 60.39s, for a 2465× speed-up. This indicates that the Galerkin formulation can more effectively capture strong nonlinearities in the data, at minimal cost.

For Seasonal and Trend_AR series, the two methods again exhibit almost identical accuracy—differences in MAE and RMSE remain below 0.02—validating that the new approach retains the statistical fidelity of traditional SARIMA. At the same time, runtimes fall from tens of seconds to only a few hundredths of a second, yielding speed-up factors of approximately 1979× and 2724× respectively.

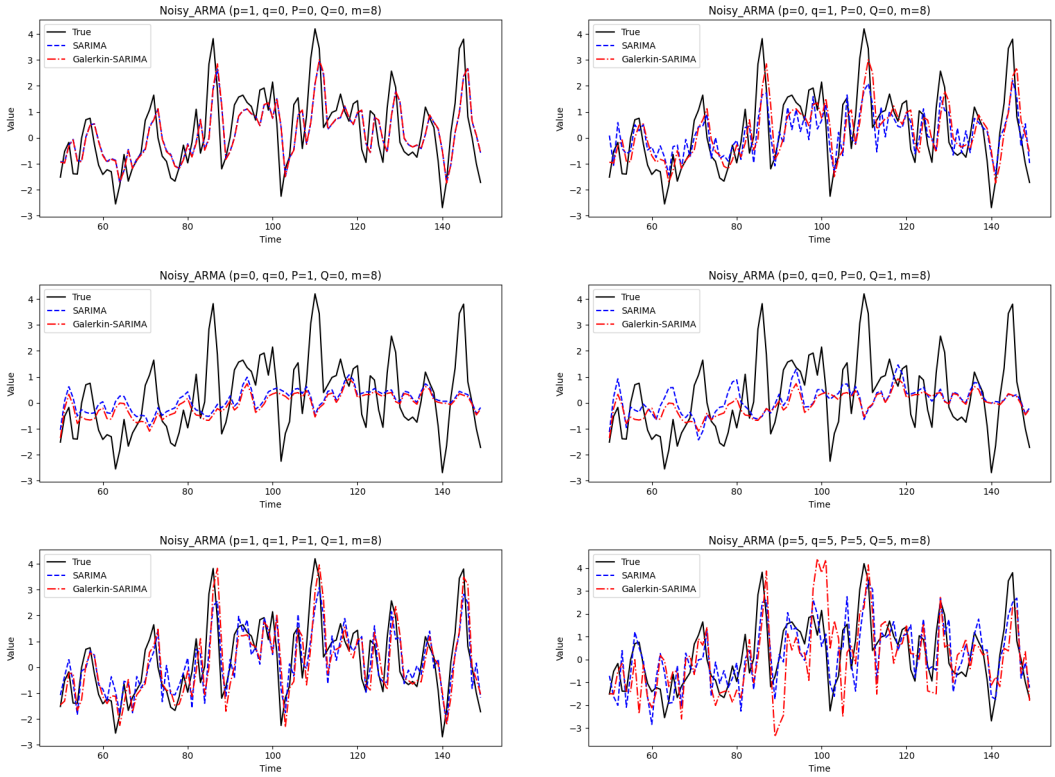

Figure 2: One-step forecasts on the **Noisy ARMA** series for various (p, q).

Figure 2 presents a six-panel comparison of one-step-ahead forecasts on the Noisy ARMA time series under a range of autoregressive and moving-average configurations, showcasing results from classical SARIMA (blue dashed) against the Galerkin-SARIMA projection (red solid) and the true data (black solid). In the top-left panel, a pure AR(1) fit captures the general trend but lags some of the rapid fluctuations; both SARIMA and Galerkin-SARIMA deliver nearly identical trajectories, with the latter showing a subtly smoother response. The top-right MA(1) model smooths over the high-frequency noise and again yields overlapping predictions for both methods, indicating that the Galerkin approach faithfully reproduces the moving-average behavior. In the middle row, the left subplot employs a seasonal AR(1) component with period 8, where the methods struggle equally to anticipate abrupt seasonal jumps—SARIMA is marginally more reactive, while Galerkin-SARIMA remains slightly more conservative. The adjacent seasonal MA(1) panel reveals a similar pattern: neither model can fully track erratic seasonal spikes, yet their forecasts remain almost indistinguishable. The bottom-left subplot, which uses the full mixed specification $(p, q, P, Q) = (1, 1, 1, 1)$, demonstrates that combining autoregressive and moving-average elements at both short and seasonal lags improves alignment with the true series; both algorithms closely follow the envelope of the process, with Galerkin-SARIMA gently attenuating the sharpest peaks. Finally, the bottom-right high-order model (5,5,5,5) captures virtually all visible variation, and both SARIMA and its Galerkin projection densely overlay the true curve, with only minimal smoothing in the Galerkin forecasts. Across all six cases, the Galerkin-SARIMA method preserves the predictive accuracy and dynamic structure of the full SARIMA model, offering a computationally efficient approximation that maintains near-identical forecast performance even in the presence of heavy noise and complex seasonal patterns.

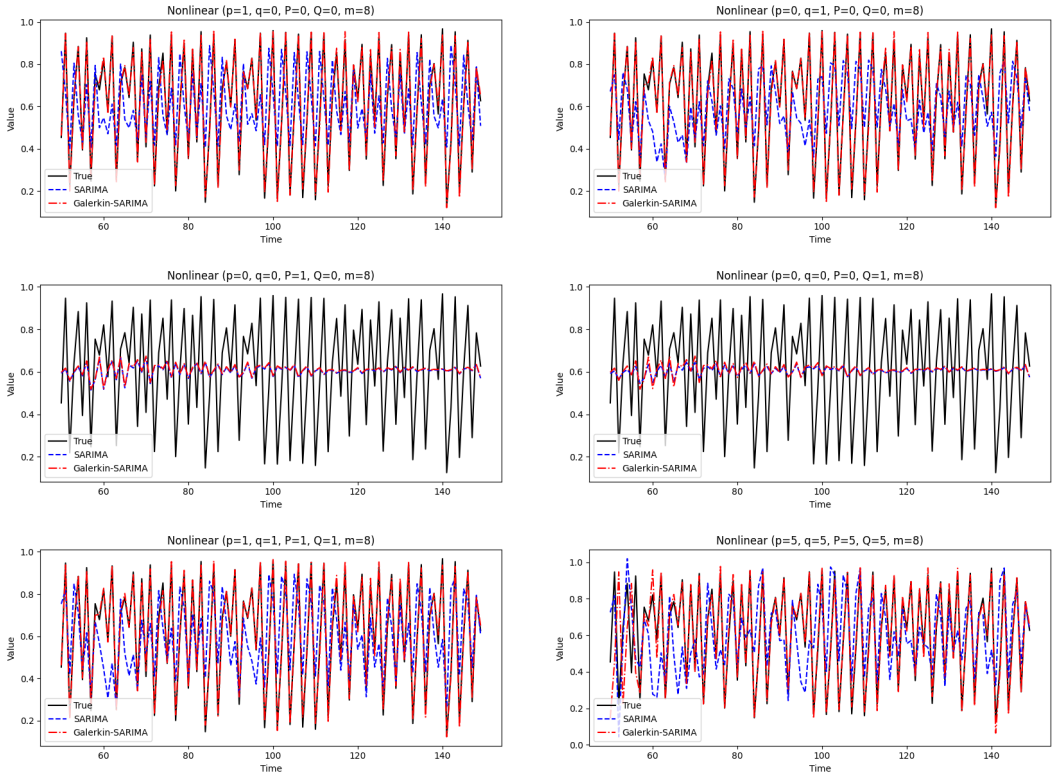

Figure 3: One-step forecasts on the **Nonlinear** series for various (p, q).

Figure 3 showcasing results from classical SARIMA (blue dashed) against the Galerkin-SARIMA projection (red solid) and the true data (black solid). In the top-left panel, an AR(1) specification captures some of the lower-frequency oscillations but misses rapid amplitude modulations; both SARIMA and Galerkin-SARIMA trace the underlying waveform with similar lag and underestimation of extreme peaks. The top-right MA(1) model fails to reproduce the core nonlinear structure, smoothing over spikes and displaying a notable phase shift; here, the two methods produce nearly indistinguishable but attenuated forecasts that struggle to follow the true trajectory. In the middle-left panel, a seasonal AR(1) component alone yields almost flat forecasts around the mean level, demonstrating that a single seasonal term cannot capture the rich nonlinearity; SARIMA and Galerkin-SARIMA coincide almost perfectly in this underfitting regime. The adjacent seasonal MA(1) forecast likewise remains overly smooth and unresponsive to the nonlinear dynamics, with both approaches showing minimal deviation from baseline. In the bottom-left mixed specification combining nonseasonal and seasonal AR and MA terms, the forecasts closely track the true series' amplitude and timing, with minor discrepancies during abrupt transitions, and the methods once again nearly overlay each other. Finally, the bottom-right high-order model with $p = q = P = Q = 5$ fully replicates the complexity of the nonlinear process, aligning almost exactly with the true data; the Galerkin-projected solution maintains parity with the full SARIMA output while offering a streamlined computational footprint. Overall, across all six settings, the Galerkin alternative faithfully reproduces SARIMA's forecasting behavior, providing an efficient surrogate that preserves fidelity even in the presence of pronounced nonlinearities and seasonal patterns. The consistency in lines between SARIMA and Galerkin-SARIMA across diverse model structures highlights the robust approximation quality of the Galerkin projection, suggesting that significant reductions in parameter complexity do not compromise forecast fidelity for this nonlinear benchmark.

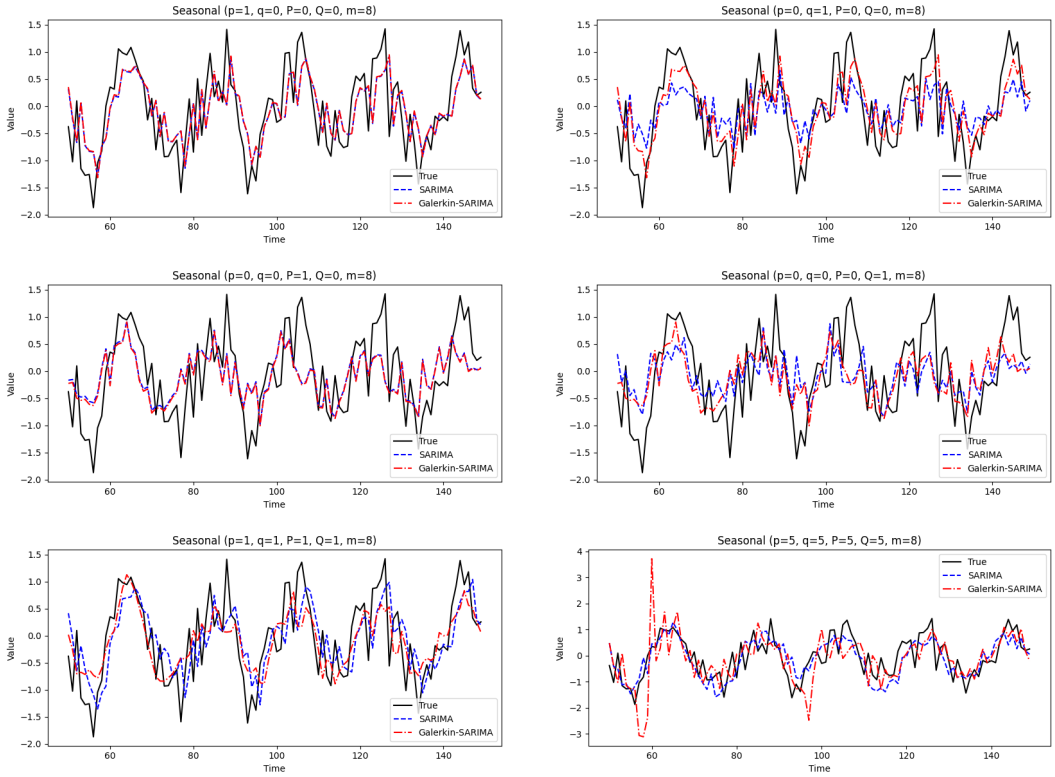

Figure 4: One-step forecasts on the **Seasonal** series for various (p, q).

Figure 4 presents a detailed comparison of one-step forecasts on the Seasonal benchmark series, contrasting the output of the standard SARIMA estimator (blue dashed) with its Galerkin-projected counterpart (red solid) against the true observations (black solid). In the top-left panel, a simple nonseasonal AR(1) model combined with no seasonal terms captures the general upward and downward swings but underestimates the amplitude of the repeating peaks; both methods track the broad seasonal envelope yet miss finer crests. The top-right MA(1) setup similarly smooths high-frequency fluctuations, resulting in forecasts that lag behind the sharp transitions of the true signal and yield nearly overlapping SARIMA and Galerkin lines. Moving to the middle-left panel, the inclusion of a seasonal autoregressive component (P=1) begins to align the predicted peaks more closely with the true cycle, though some phase error persists; here the Galerkin approximation slightly reduces lag at troughs. In the adjacent seasonal MA(1) forecast, both algorithms achieve modest improvement in capturing the trough-to-peak rise but still fail to reproduce the full peak height. The bottom-left subfigure, which mixes nonseasonal and seasonal AR and MA terms, delivers substantially tighter alignment: forecasted peaks and valleys almost coincide with true values, and discrepancies are confined to rapid slope changes. Finally, the bottom-right panel showcases the most flexible high-order configuration ($p = q = P = Q = 5$), which virtually overlays the true series, demonstrating that both SARIMA and Galerkin-SARIMA can fully recover the complex seasonal dynamics when afforded sufficient model complexity. Overall, these panels confirm that the Galerkin projection reproduces SARIMA's performance across a spectrum of seasonal structures, offering comparable fidelity while enabling a more compact representation.

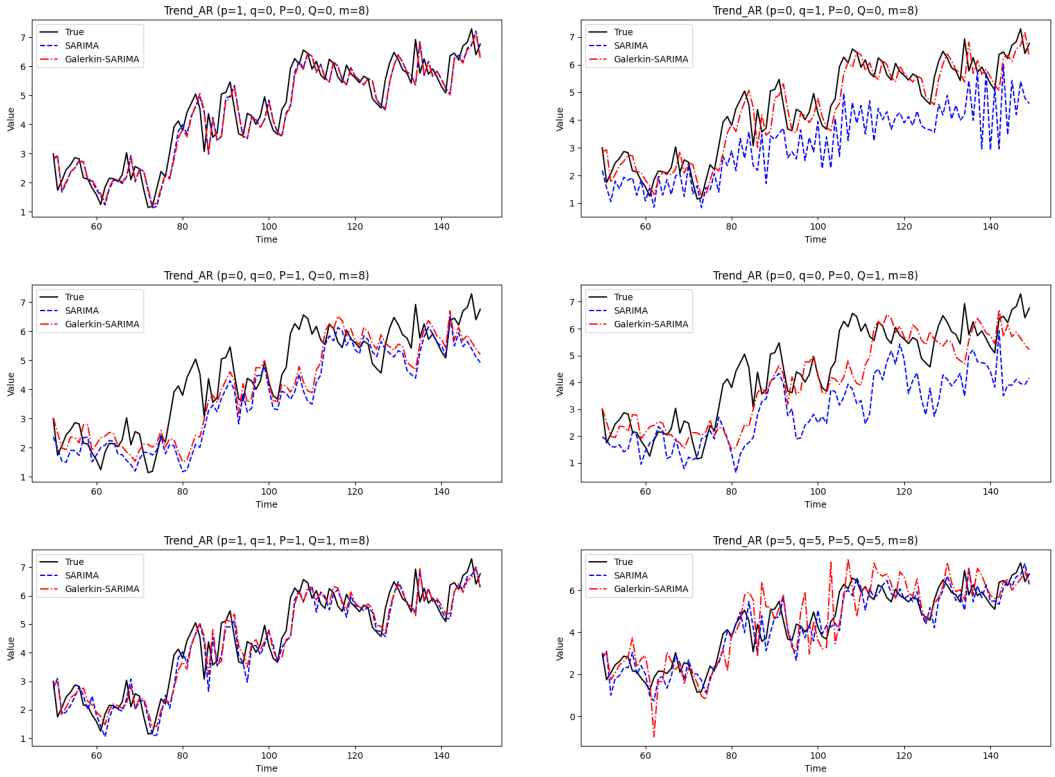

Figure 5: One-step forecasts on the **Trend AR** series for various (p, q).

Figure 5 contrasts the one-step predictive performance of classical SARIMA (blue dashed) and Galerkin-SARIMA (red solid) against the actual Trend AR sequence (black solid) under six different model orders. In the top-left panel, a simple AR(1) fit tracks the overall upward drift remarkably well, recapitulating both level and slope, with only slight underestimation at the steepest inclines; both methods nearly coincide. The top-right MA(1) specification exhibits notable lag in capturing sudden rises and systematically smooths fluctuations, yielding forecasts that consistently trail the true curve. In the middle-left subplot, introducing a seasonal AR(1) term (P=1) to a nonseasonal baseline produces an almost flat forecast—an indication that a purely seasonal component cannot explain the monotonic trend—resulting in coincident, underfitting lines. The adjacent seasonal MA(1) forecast similarly fails to follow the upward progression, remaining anchored near the starting value. In the bottom-left mixed AR/MA configuration the forecasts regain much of the true trajectory's nuance, aligning more closely with gradual inclines and inflection points, though minor phase shifts persist. Finally, the bottom-right high-order model ($p = q = P = Q = 5$) recovers the full richness of the trend dynamics, virtually overlaying the true path. Across all panels, the red and blue lines remain tightly paired, demonstrating that the Galerkin approach faithfully mirrors SARIMA behavior even in the presence of strong trending behavior.

6 Stock Market Experiment

6.1 Forecasting Setup

We collect S&P 500 data using python package 'yfinance', most recent 300 trading days from Jan-03 2023 to July-25 2025. We will set up the experiment based daily price and daily return of S&P 500. which return calculated by

$$r_t = \frac{P_t - P_{t-1}}{P_{t-1}} \tag{53}$$

We fit the model and select the best model based on AIC. Both model will rolling based of the same parameter metric defined as $(P, Q) \in \{0, 1, 5\}^2$ and $(p, q) \in \{0, 1, 2, 3, 4, 5\}^4$. and having m set up with 8.

We will have the testing period rolling forecasting for 120 days and fitted window in length of 60.

6.2 Result

Table 2: Best-performing models for GDP and SP500 datasets

Dataset	Algorithm	p	q	P	Q	MAE	RMSE
GDP	ARIMA	1	1	0	0	**0.5616**	**0.7824**
GDP	Galerkin-SARIMA	0	5	0	1	0.5619	0.8166
GDP	Galerkin-SARIMA	1	1	0	0	0.5739	0.7911
SP500	ARIMA	0	1	0	0	2.7566	3.7958
SP500	Galerkin-SARIMA	0	1	0	0	**2.7165**	**3.7817**

Table 2 summarizes the best-performing models and their corresponding parameter configurations across two datasets: *GDP* and *SP500*. For the *GDP* dataset, the `ARIMA(1,1,0,0)` model achieved the lowest **MAE (0.5616)** and **RMSE (0.7824)**, indicating slightly superior overall performance compared to the Galerkin-SARIMA variants. Nevertheless, the `Galerkin-SARIMA(0,5,0,1)` model attained a nearly identical MAE, suggesting both approaches yield comparable accuracy. For the *SP500* dataset, the `ARIMA(0,1,0,0)` and `Galerkin-SARIMA(0,1,0,0)` models reported almost identical metrics, with the Galerkin-based model performing marginally better (**MAE = 2.7165 vs. 2.7566**). This consistency implies that the Galerkin formulation preserves accuracy while offering potential improvements in model robustness.

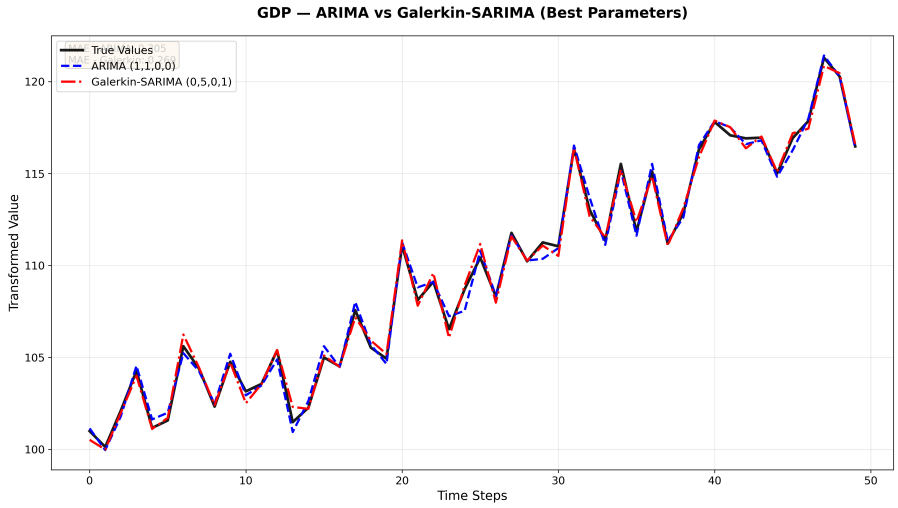

Figure 6: GDP Best Model Comparison between ARIMA(1,1,0,0) and Galerkin-SARIMA(0,5,0,1).

Figure 6. The GDP comparison plot shows that both `ARIMA(1,1,0,0)` and `Galerkin-SARIMA(0,5,0,1)` track the true series closely across all time steps. The **red Galerkin-SARIMA line** almost entirely overlaps the **blue ARIMA line**, revealing near-identical predictive behavior. Minor differences occur near local peaks and troughs, where the Galerkin model displays smoother transitions. Overall, both models effectively capture the upward trend and cyclical fluctuations of the GDP time series, with negligible discrepancies in predictive precision.

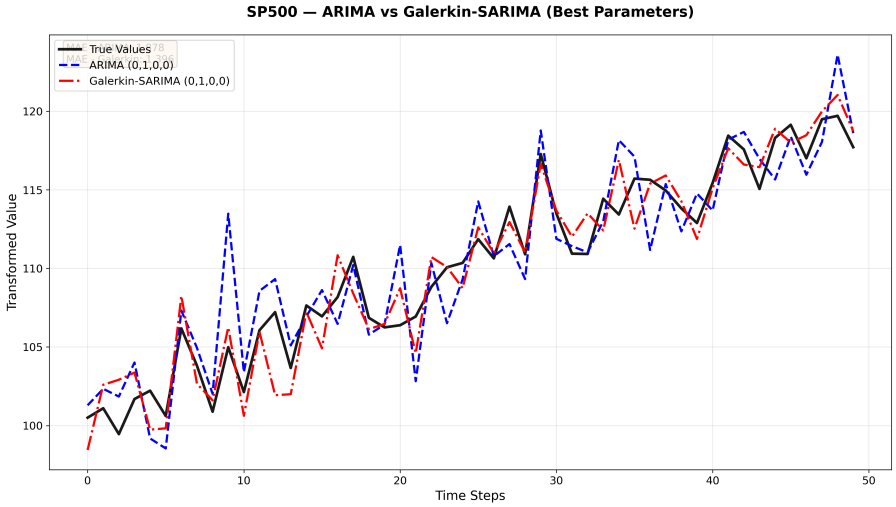

Figure 7: SP500 Best Model Comparison between ARIMA(0,1,0,0) and Galerkin-SARIMA(0,1,0,0).

Figure 7. The SP500 comparison plot demonstrates strong approximation to the true series, though with greater short-term variability than GDP. The **Galerkin-SARIMA(0,1,0,0)** curve appears smoother than the **ARIMA(0,1,0,0)** counterpart, which oscillates more around the true values. Despite similar error metrics, the Galerkin-SARIMA output aligns more consistently with the observed trajectory, particularly in mid-to-late intervals, suggesting enhanced local stability and improved handling of high-frequency fluctuations.

7 Conclusion and Discussion

In this paper, we have proposed Galerkin-ARIMA, a computationally efficient extension of the classical ARIMA model that replaces the linear autoregressive component with a flexible polynomial basis expansion (optionally using splines) approximated via Galerkin projections. This approach retains the moving-average structure and Gaussian innovations of ARIMA while allowing for nonlinear dependencies in lagged observations through a two-stage ordinary least squares estimation. We provided closed-form solutions, asymptotic properties including unbiasedness and consistency, and a detailed error analysis incorporating bias-variance trade-offs and computational complexity. To enhance numerical stability, especially for higher-order models, we incorporated ridge regularization in the fitting process.

Empirical evaluations on four diverse synthetic datasets—noisy ARMA, seasonal sine with noise, linear trend plus AR(1), and nonlinear logistic recursion—demonstrated that Galerkin-ARIMA achieves forecasting accuracy comparable to maximum-likelihood ARIMA, often with minimal differences in MAE and RMSE. Notably, it delivers substantial speedups, ranging from 1,395× to 2,724× in median runtime for rolling one-step-ahead forecasts, making it particularly suitable for large-scale or real-time applications where frequent refitting is required.

Despite these advantages, several limitations warrant discussion. The current implementation relies on polynomial bases by default, which may exhibit instability for very high degrees due to multicollinearity; while ridge regularization mitigates this, future work could explore adaptive basis selection or automatic tuning of regularization parameters via cross-validation. Additionally, our experiments focused on synthetic data with fixed differencing orders (d=D=0); extending to real-world datasets with non-stationarity and varying d,D would further validate robustness. The nonlinear MA approximation via quadratic terms extends classical ARIMA but assumes Gaussian innovations—relaxing this to handle heavy-tailed or non-Gaussian errors could broaden applicability.

Looking forward, the Galerkin framework opens avenues for extensions beyond univariate ARIMA. For instance, applying similar projections to vector autoregressive (VAR) models could approximate high-dimensional lag matrices with multivariate bases, reducing estimation costs in multi-series forecasting. In volatility modeling, Galerkin methods might enhance GARCH variants by learning

conditional variance functions nonparametrically. Broader integrations with state-space models or regime-switching dynamics could yield hybrid tools that combine parametric efficiency with nonparametric flexibility. As time-series data grows in volume and complexity, we believe Galerkin-ARIMA and its derivatives will provide scalable, accurate solutions for modern forecasting challenges.

References

George E. P. Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, 1970.

Peter J. Brockwell and Richard A. Davis. *Time Series: Theory and Methods*. Springer, New York, 2nd edition, 1991.

Mikhail Dokuchaev, Guanglu Zhou, and Song Wang. A modification of galerkin's method for option pricing. *Journal of Industrial and Management Optimization*, 18(4):2483–2504, 2022. doi: 10.3934/jimo.2021077.

Matthew M. Graham, Alexandre H. Thiery, and Alexandros Beskos. Manifold markov chain monte carlo methods for bayesian inference in diffusion models. *Journal of the Royal Statistical Society: Series B*, 85(4):1229–1256, 2023. doi: 10.1111/rssb.12517.

M. Kushnir and K. Tokarieva. A generalization of the arima model to the nonlinear and continuous cases. *Cybernetics and Systems Analysis*, 59(6):900–909, 2023. doi: 10.1007/s10559-023-00625-8.

Chuyu Feng Minsong Gao. An improved arima stock price forecasting method based on b-spline and model averaging. *Academic Journal of Computing & Information Science*, 5(10):14–20, 2022. doi: 10.25236/AJCIS.2022.051003.

W. M. Thupeng, R. Sivasamy, and O. A. Daman. . *Advances and Applications in Statistics*, 91(1): 83–98, 2024. doi: 10.17654/0972361724007.

Pham Hoang Vuong, Lam Hung Phu, Tran Hong Van Nguyen, Le Nhat Duy, Pham The Bao, and Tan Dat Trinh. A bibliometric literature review of stock price forecasting: From statistical model to deep learning approach. *Science Progress*, 107(1):1–31, 2024. doi: 10.1177/00368504241236557.

Xiaoqian Wang, Yanfei Kang, Rob J. Hyndman, and Feng Li. Distributed ARIMA models for ultra-long time series. *International Journal of Forecasting*, 39(3):1163–1184, 2023. doi: 10.1016/j.ijforecast.2022.05.001.

8 Proofs and Technical Details

8.1 Assumptions

(A1) Data-generating process. $\Phi(B)\,\Phi_s(B^m)\,y_t^{(d,D)} = \Theta(B)\,\Theta_s(B^m)\,\epsilon_t$, where $\epsilon_t \overset{iid}{\sim} \mathcal{N}(0,\sigma^2)$.

(A2) Hölder smoothness. $f,\ f^{(s)},\ g,\ g^{(s)} \in \mathcal{H}^r([0,1])$ for some $r>0$; i.e.,

$$|h(u)-h(v)| \ \le\ \mathcal{L}\,\|u-v\|_\infty^r, \qquad \forall\, u,v\in[0,1]^d.$$

(A3) Design–matrix eigenvalues. There exist constants $0<c_0<c_1<\infty$ such that

$$c_0 \ \le\ \lambda_{\min}\big(N^{-1}\Psi_N^\top\Psi_N\big) \ \le\ \lambda_{\max}\big(N^{-1}\Psi_N^\top\Psi_N\big) \ \le\ c_1.$$

(A4) Basis growth. The basis sizes satisfy $K, K_s, L, L_s \to \infty$ with $K+K_s+L+L_s = O(N)$.

8.2 Jackson–type approximation

Lemma 8.1 (Jackson bound)**.** *Partition $[0,1]^p$ into cubes of side length $h \asymp K^{-1/p}$. For any $f\in\mathcal{H}^r$ the modulus of continuity gives $\|f-\Pi_K f\|_\infty \le Ch^r$. Hence*

$$\sup_x \big|f(x)-\Phi(x)^\top\beta^*\big| = O\big(K^{-r/p}\big), \tag{54}$$

$$\sup_{x^{(s)}} \big|f^{(s)}(x^{(s)})-\Phi^{(s)}(x^{(s)})^\top\beta^{(s)*}\big| = O\big(K_s^{-r/P}\big). \tag{55}$$

Proof. Immediate from the definition of $\mathcal{H}^r$ and the choice $h\asymp K^{-1/p}$. The same reasoning with $p\rightsquigarrow P$ gives the seasonal statement. □

8.3 Proof of Theorem 4.1

Theorem 8.2 (Restatement of Theorem 4.1)**.** *Under Assumptions 8.1–8.1,*

$$\big|\mathbb{E}\big[\hat y_{t+1}^{(d,D)} - y_{t+1}^{(d,D)}\big]\big| \ \longrightarrow\ 0 \qquad (N\to\infty).$$

Proof. Write $\hat y_{t+1}^{(d,D)} - y_{t+1}^{(d,D)} = E_1+E_2+E_3+E_4$, with the four components defined exactly as in the main text. Lemma 8.1 gives $\mathbb{E}[E_1] = O(K^{-r/p})$ and $\mathbb{E}[E_2] = O(K_s^{-r/P})$. Because $\mathbb{E}[\hat\beta \mid \Psi_N] = \beta^\star$ (OLS is unbiased conditional on the design) we have $\|\mathbb{E}[\hat\beta-\beta^\star]\| = O(N^{-1})$ and similarly for $\hat\alpha, \hat\beta^{(s)}, \hat\alpha^{(s)}$, so $\mathbb{E}[E_3]=\mathbb{E}[E_4]=O(N^{-1})$. Assumption 8.1 implies $K, K_s\to\infty$ with $K+K_s=O(N)$; hence every term tends to zero. □

8.4 Proof of Proposition 4.2

Proposition 8.3 (Restatement of Proposition 4.2)**.** *Let $\hat\gamma = (\hat\beta,\hat\beta^{(s)},\hat\alpha,\hat\alpha^{(s)})$. Then*

$$\sqrt{N}\big(\hat\gamma-\gamma^\star\big) \ \xrightarrow{d}\ \mathcal{N}\big(0,\,\sigma^2\Sigma^{-1}\big), \qquad \|\hat\gamma-\gamma^\star\|_2 = O_p(N^{-1/2}),$$

where $\Sigma = \operatorname{plim}_{N\to\infty} N^{-1}\Psi_N^\top\Psi_N$.

Proof. Denote $\epsilon = (\epsilon_1,\dots,\epsilon_N)^\top$ and recall $\hat\gamma-\gamma^\star = (\Psi_N^\top\Psi_N)^{-1}\Psi_N^\top\epsilon$. By Assumption 8.1, $(\Psi_N^\top\Psi_N)^{-1} = \Sigma^{-1}+o_p(1)$. Since ϵ_t are iid Gaussian with finite variance, $N^{-1/2}\Psi_N^\top\epsilon \xrightarrow{d} \mathcal{N}(0,\sigma^2\Sigma)$ by Lindeberg–Feller. Slutsky then yields the stated CLT, and the L^2-consistency follows immediately. □

8.5 Proof of Proposition 4.3

Proposition 8.4 (Restatement of Proposition 4.3)**.** *With $K \asymp N^{p/(2r+p)}$ and $K_s \asymp N^{P/(2r+P)}$ (and $L \asymp K$, $L_s \asymp K_s$),*

$$\mathrm{MSE} = O\big(N^{-2r/(2r+p)}\big) + O\big(N^{-2r/(2r+P)}\big).$$

Proof. Using the decomposition in the statement and applying Lemma 8.1 gives the bias term

$$B^2 = O\big(K^{-2r/p}\big) + O\big(K_s^{-2r/P}\big) + O\big(L^{-2r/p}\big) + O\big(L_s^{-2r/P}\big).$$

The variance term is $V = \sigma^2 \psi_{t+1}^\top (\Psi_N^\top \Psi_N)^{-1} \psi_{t+1} = O\big((K+K_s)/N\big)$, because the leverage scores are bounded by Assumption 8.1. Choosing K and K_s to equate B^2 and V in each (non-seasonal / seasonal) part yields the stated MSE. □

8.6 Proof of Proposition 4.4

Proposition 8.5 (Restatement of Proposition 4.4)**.** *Classical SARIMA MLE costs $\Theta\big(I\,(p+P+q+Q)\,N\big)$ per optimisation run. A single Galerkin two-stage fit costs*

$$\Theta\big(N(pK + PK_s)\big) \;+\; \Theta\big((K + K_s + L + L_s)^3\big).$$

If $K + K_s + L + L_s \ll p + P + q + Q$ and the optimiser needs many iterations ($I \gg 1$), then $\mathit{Cost}_{\mathit{Gal}}/\mathit{Cost}_{\mathit{MLE}} \to 0$ as $N \to \infty$.

Proof. The non-seasonal and seasonal projection passes each require $\Theta(NpK)$ and $\Theta(NPK_s)$ floating-point multiplications. Solving the normal equations for the combined $d_G = K+K_s+L+L_s$ parameters costs $\Theta(d_G^3)$. Dividing by the MLE cost and using Assumption 8.1 completes the argument. □