Merkatas, Christos; Särkkä, Simo

System identification using autoregressive Bayesian neural networks with nonparametric noise models

Published in:
Journal of Time Series Analysis

DOI:
10.1111/jtsa.12669

Published: 01/05/2023

Please cite the original version:
System identification is of special interest in science and engineering. This article is concerned with a system identification problem arising in stochastic dynamic systems, where the aim is to estimate the parameters of a system along with its unknown noise processes. In particular, we propose a Bayesian nonparametric approach for system identification in discrete time nonlinear random dynamical systems assuming only the order of the Markov process is known. The proposed method replaces the assumption of Gaussian distributed error components with a flexible family of probability density functions based on Bayesian nonparametric priors. Additionally, the functional form of the system is estimated by leveraging Bayesian neural networks, which leads to flexible uncertainty quantification. Hamiltonian Monte Carlo sampler within a Gibbs sampler for posterior inference is proposed and its effectiveness is illustrated in real time series.

Received 25 January 2022; Revised 23 September 2022; Accepted 12 October 2022

Keywords: Bayesian nonparametrics; geometric stick breaking; infinite mixture models; nonlinear time series; system identification.

JEL. 62F15; 62M10.

1. INTRODUCTION

Random dynamical systems, that is, dynamical systems that evolve through the presence of noise, play an important role in science and engineering as they can be used to model and explain dependencies in physical phenomena changing in time. This noise may be observational, attributed to measurement errors from the measuring device, or dynamical which compensates for modeling error or both. Common areas of application include but are not limited to neuroscience, finance (Franses et al., 2000; Small, 2005; Ozaki, 2012), and signal processing (Lau and Tse, 2003).

In this article, we consider autoregressive dynamical systems that are observed in the form of a time series and the mathematical model describing the process is given by

\[ y_t = g(y_{t-\rho}; \theta) + z_t, \quad t = 1, 2, \ldots , \]

where \( y_t \) is the observed process and the vector \( y_{t-\rho} := (y_{t-1}, \ldots, y_{t-\rho}) \) consists of a set of lagged versions of the process \( y \), which allows the system to be written in the form of a regression model. Additionally, \( \theta \) is a vector of parameters \( \theta := (\theta_1, \ldots, \theta_m) \) and \( \{z_t\}_{t\geq 1} \) is a collection of i.i.d. zero-mean noise components with finite variance. For simplicity, we assume that we have at our disposal a time series of length \( n \) generated from (1) and that the vector of \( \rho \) initial states is known.

System identification refers to the problem of forming an estimate \( \hat{g} \) of \( g \) or, equivalently, an estimate \( \hat{\theta} \) of its parameters such that it describes the observed process well enough to understand the dynamics and use it for control.
and/or prediction of future unobserved values of the system (Sjöberg et al., 1995; Ljung, 1999; Nelles, 2001; Keesman, 2011). System identification is typically done with some statistical method for estimating the parameters based on the available data.

For a discrete-time scalar time series \( \{y_t\}_{t \geq 1} \), the Box–Jenkins (Box et al., 2015) modeling approach is to assume an autoregressive integrated moving average ARIMA\((p, d, q)\) model for \( \{y_t\} \). In this framework, a nonstationary time series is differentiated \( d \) times to become stationary and then a suitable model for representing the process \( y_t \) is \( \Omega_p(B)y_t = \Psi_q(B)z_t \), where

\[
\begin{align*}
\Omega_p(B) &= 1 - \omega_1 B - \cdots - \omega_p B^p, \\
\Psi_q(B) &= 1 - \psi_1 B - \cdots - \psi_q B^q,
\end{align*}
\]

are polynomials of degrees \( p \) and \( q \), respectively, of the backshift operator \( B \) defined as \( B^q y_t = y_{t-q} \). The polynomial degrees and the coefficients \( \theta = (\omega_1, \ldots, \omega_p, \psi_1, \ldots, \psi_q) \) of the polynomials are estimated from the observed data.

These models work extremely well for linear Gaussian time series, however, they are not sufficient for nonlinear or chaotic time series modeling. To this end, more sophisticated, nonlinear models for the function \( g \) have been developed. Some of the popular choices include, among others, the bilinear model, the threshold autoregressive model along with the extension to self-exciting threshold autoregressive model (Tong and Lim, 1980), and the autoregressive conditional heteroscedasticity (ARCH) model (Engle, 1982). Moreover, black-box nonlinear models using artificial neural networks have been successfully used for nonlinear time series estimation and prediction (Chen and Billings, 1992; Sjöberg et al., 1994; Sjöberg et al., 1995; Zhang, 2003; Aladag et al., 2009).

The so-called Bayesian approaches (Gelman et al., 2013) incorporate prior information in terms of probability distributions to the estimation process. Bayesian methods have been used to estimate the parameters of maps responsible for the generation of chaotic signals contaminated with Gaussian noise (Meyer and Christensen, 2000; Pantaleón et al., 2000; Luengo et al., 2002; Pantaleón et al., 2003). In all cases above, the general functional form of the system in question is known and the aim is to estimate its parameters. Bayesian methods have also been used to estimate regression models with ARMA\((p, q)\) noise components (Chib and Greenberg, 1994) and piecewise Markov systems (Pantaleón et al., 2003). In the context of Bayesian estimation, but regarding dynamically noise-corrupted signals and taking a black-box approach, Matsumoto et al. (2001); Nakada et al. (2005) developed hierarchical Bayesian methods for the reconstruction and prediction of random dynamical systems. More recently, there has been increasing research interest in the development of Gaussian process models (Rasmussen and Williams, 2006) for black-box time series modeling (Kocijan et al., 2005; Turner et al., 2009; Bijl et al., 2017; Särkkä et al., 2018).

A common feature of most of the aforementioned methods is that they are based on the rather strong assumption of Gaussian noise which is rarely true in real world applications and is typically chosen for computational convenience. Other choices, based on scaled mixtures of Gaussians have also been proposed (Ishwaran and Rao, 2005; Carvalho et al., 2010; Polson and Scott, 2010) however, these choices are still parametric capturing only a bounded amount of information in the data. On the other hand, restricting \( g \) to belong to a particular class of functions might be difficult when there is not enough information for the problem at hand, leading to models with poor generalization capabilities.

In this article, we aim to develop hierarchical Bayesian models for the estimation and prediction of random dynamical systems from observed time series data. In particular, we use a nonparametric Bayesian prior namely, the geometric stick breaking process (GSB) prior introduced by Fuentes-Garcia et al. (2010) which allows for flexible modeling of unknown distributions to drop the common assumption of Gaussian distributed noise components. Bayesian nonparametric modeling of the noise process has been successfully attempted before (Hatjispyros et al., 2009; Merkatas et al., 2017; Hatjispyros and Merkatas, 2019) in the context of dynamical reconstruction from observed time series data.
Our model is an extension of the model proposed at Merkatas et al. (2017) where we additionally aim to adopt a black-box modeling approach that does minimal assumptions for the data at hand. In particular, we drop the assumption of a known functional form of the deterministic part of the system by leveraging Bayesian neural networks (Neal, 1996). We parametrize the data generating mechanism with a autoregressive feed-forward neural network whose weights and biases are assigned a prior distribution. As discussed above, the nonparametric modeling of the noise process as well as the use of autoregressive neural networks to model the deterministic part are not novel as such. However, the contribution of this work is to combine the two approaches to make the model flexible and suitable for modeling real time series data, requiring only the order (lag) $\rho$ of the underlying Markov process to be known or estimated. The choice of lag is a model selection problem typically addressed using some information criterion like the Akaike information criterion (AIC) or Watanabe’s widely applicable information criterion (Akaike, 1974; Watanabe, 2012). A survey of Bayesian predictive methods for model comparison can be found in Vehtari and Ojanen (2012). A more direct procedure to select the value of $\rho$ is by examination of the partial autocorrelation function (PACF) (Chatfield, 2000, Ch. 3.3.1). Partial autocorrelations after some lag $\rho^*$ not significantly different from zero, indicate that the time series is obtained from an AR($\rho^*$) model. Recent advances in embedding dimensions and signature learning (Fermanian, 2021) may provide additional information on how to select the lag in the inferential process but this is beyond the scope of this article. Our model is similar to the model of Chib and Greenberg (2010), however, we use a random distribution with simpler weights for the errors. Additionally, the use of feed forward neural network opens the path to incorporate more sophisticated neural network architectures to model more complex dynamical phenomena in the future.

It should be stressed that a purely Bayesian nonparametric approach to the problem described here would be a density regression approach. In density regression, the functional form of the system and the noise process are assumed unknown and the use of the so called dependent nonparametric priors is required (Fuentes-García et al., 2009; Mena et al., 2011; Lijoi et al., 2014; Hatjispyros et al., 2018). However, the construction of such priors can be hard especially for continuous time series modeling because dependencies among the random probability measures changing in time would require analytical solutions to stochastic differential equations or, at least, efficient computational methods. Our framework can be extended to continuous time modeling using more sophisticated neural network architectures, for example, neural ordinary differential equations (Chen et al., 2018).

The article is organized as follows. Section 2 introduces the proposed model for estimation and prediction of nonlinear, stationary random dynamical systems. Next, in Section 3 we introduce a fast Markov chain Monte Carlo (MCMC) sampling algorithm for posterior inference based on a Gibbs kernel and we show how the samples taken from the posterior can be used for prediction of future values. Experimental evaluation in simulated and real data is given in Section 4. Finally, we conclude and discuss the results in Section 5.

2. NONLINEAR NONPARAMETRIC AUTOREGRESSIVE PROCESS FOR TIME SERIES MODELING

We consider a discrete time dynamical system

$$y_t = g(y^\rho_t, \theta) + z_t, \quad 1 \leq t \leq n,$$

where $g : \mathbb{R}^{D \rho} \times \Theta \rightarrow \mathbb{R}^D$ is a nonlinear function of $y^\rho_t := (y_{t-1}, \ldots, y_{t-\rho})$ parametrized by a vector of parameters $\theta \in \Theta \subset \mathbb{R}^m$. The random variables $z_t$ represent additive dynamical noise which for now we assume are i.i.d. samples from an unknown density $f(\cdot)$ taking values in $\mathbb{R}^D$.

2.1. Nonparametric Noise Model

Here we aim to construct the nonparametric model for the density of the noise $z_t$ with generic function $g$. 

In Merkatas et al. (2017), the following hierarchical model was introduced. For a time series \( \{ y_t, 1 \leq t \leq n \} \) we have

\[
y_t | y^o_t, \theta, \Lambda^\text{ind} \sim N(y_t | g(y^o_t, \theta), \Lambda_t^{-1}), \\
\Lambda_t | \mathbb{P} \sim \mathbb{P}(\cdot), \quad \theta \sim f(\theta), \\
\mathbb{P} \sim \Pi(\phi, P_0),
\]

where \( \phi \) parametrizes \( \Pi \) and \( P_0(d\Lambda) \) is a parametric base measure. In this model, the random density of the noise components \( f(z) \) is a mixture density of the random probability measure \( \mathbb{P} = \sum_{j=1}^{\infty} \pi_k \delta_{\Lambda_j} \) via \( f(z) = \int_{\mathbb{M}^+} N(z | 0, \Lambda^{-1}) \mathbb{P}(d\Lambda) \), where \( \mathbb{M}^+ \) denotes the set of \( D \times D \) positive semidefinite matrices defined on \( \mathbb{R} \) and \( 0 \) is the \( D \)-dimensional vector of zeros.

Following Fuentes-Garcia et al. (2010), for each observation we introduce the almost surely finite discrete random variables \( R_t \) for \( 1 \leq t \leq n \) with probability mass function \( f(\cdot | \phi) \) where \( \phi \) is a hyperparameter. Additionally, we introduce a clustering variable \( d_t \) for \( 1 \leq t \leq n \) that indicates the component of the mixture that each observation came from. Conditionally on \( R_t \)'s the \( d_t \)'s have a discrete uniform distribution on the set \( \{1, \ldots, R_t\} \). Then, the \((d_t, R_t)\)-augmented transition density for an observation reads

\[
f(y_t, R_t = r, d_t = k) = \frac{1}{r} f(R_t = r | \phi) \mathbb{I}(k \leq r) N(y_t | g(y^o_t, \theta), \Lambda_k^{-1}).
\]  

Here and from now on, we drop the conditioning on the rest of the variables to avoid notational clutter. Marginalizing equation (4) w.r.t. \((d_t, R_t)\) and selecting \( f(R_t = r | \phi) = \text{Nbin}(R_t | 2, \phi) \) where \( \phi \) is a probability parameter, the random density becomes \( f(y_t) \) a geometric stick breaking process mixture model with density given by

\[
f(y_t) = \sum_{k=1}^{\infty} \pi_k N(y_t | g(y^o_t, \theta), \Lambda_k^{-1}), \quad \text{where} \quad \pi_k = \phi(1 - \phi)^{k-1}.
\]

We note here that other discrete distributions can be used to construct decreasing weights. It can be shown (De Blasi et al., 2020) that negative-binomial distributions with parameter 3 lead to weights which have the form \( \pi_k = \phi(1 - \phi)^{k-1}(1 + k\phi)/2 \). The choice of Poisson distribution with parameter \( \phi > 0 \), shifted to start at 1, would give \( \pi_k = \Gamma(k) - \Gamma(k, \phi))/\phi \Gamma(k) \) where \( \Gamma(\cdot, \cdot) \) are the Gamma and upper incomplete Gamma functions, respectively.

### 2.2. Nonlinear Autoregressive Bayesian Neural Network Model

To be able to capture nonlinearities in the time series with our model, a sufficiently expressive function should be chosen for \( g(\cdot, \theta) \). To this end, we model the function responsible for the generation of the time series \( g(\cdot, \theta) \) where \( \theta = (W_l^{(i)}, b_l^{(i)})_{i=1}^{L} \), with a Bayesian neural network (Neal, 1996) of \( L \) layers and tanh activation functions \( h_l^{(i)} \). The prior over the parameters in \( \theta \) is taken to be Gaussian parametrized by precision \( \tau \). In particular, we tie each group of weights and biases in a specific layer to their own precision \( \tau_w^{(l)}, \tau_b^{(l)} \) so that for \( 1 \leq l \leq L \):

\[
W_l^{(i)} \sim^\text{ind} N \left( 0, \left[ \tau_w^{(l)} \right]^{-1} \right), \quad b_l^{(i)} \sim^\text{ind} N \left( 0, \left[ \tau_b^{(l)} \right]^{-1} \right),
\]

where \( v \sim \nu \) denotes independently \( \nu \)-distributed elements \( v_i \) of the vector/matrix \( v \). The precision parameters \( \tau_\cdot^{(l)} \) control how much the weights/biases can vary in group \( l \) and are further assigned Gamma hyperprior distributions \( \text{Ga}(\tau^{(l)} | \alpha_\cdot, \beta_\cdot) \) with mean \( \alpha_\cdot/\beta_\cdot \). This way a hierarchical model for the weights and biases between layers is defined and information can be shared among different layers, a concept known as “borrowing of strength”
(Neal, 1996; Gelman et al., 2013). Having a collection of weights and biases, the $o$th output of a network with a single hidden layer of $N_l$ neurons is computed by

$$g_o(y_t^o, \theta) = \sum_{j=1}^{N_l} W_{oj}(y_t^j) h_l^{(1)} \left( \sum_{i=1}^{d} W_{ji} y_t^i + b_j^{(0)} \right) + b_o^{(1)} , \quad 1 \leq o \leq D. \tag{7}$$

### 2.3. Combined Model and Its Properties

For our purposes, we choose $f_R(R_t) = \text{Nbin}(R_t = r \mid 2, \phi)$ for the auxiliary variables. Under this choice, and this will make the steps of the Gibbs sampler easier to understand, we can write the proposed nonlinear model in hierarchical fashion as

$$y_t \mid y_t^o, \theta, d_i = k, \Lambda \sim \text{N}(y_t \mid g(y_t^o, \theta), \Lambda^{-1}_k),$$

$$W_l^{(i)} \sim \text{N} \left( 0, \left[r_l^{(i)}\right]^{-1} \right), \quad b_l^{(1)} \sim \text{N} \left( 0, \left[r_l^{(1)}\right]^{-1} \right), \quad 1 \leq l \leq L,$$

$$\tau_W^{(i)} \sim \text{Ga}(\alpha_{\tau_W}, \beta_{\tau_W}), \quad \tau_b^{(i)} \sim \text{Ga}(\alpha_{\tau_b}, \beta_{\tau_b}), \quad 1 \leq l \leq L,$$

$$d_i = k \mid R_t = r \sim \text{DiscreteUnif} \{1, \ldots, r\},$$

$$R_t \sim \text{Nbin}(2, \phi), \quad \phi \sim \text{Be}(\alpha_\phi, \beta_\phi),$$

$$\pi_k = \phi(1 - \phi)^{k-1}, \quad \Lambda_k \sim P_0, \quad 1 \leq k \leq R^*,$$

where $R^* := \max_i \{R_t\}$. The hierarchical Bayesian model defined above, is a highly flexible regression model in the sense that a nonparametric prior is defined on the space of all possible noise distributions while the functional form of the system is estimated from the Bayesian neural network. We will call the model with those properties NP-BNN from now on.

### 2.4. Choice of Hyperparameters

We discuss how the different hyperparameters affect the inferential process and discuss an approach for specifying the hyperparameters of the model. First consider the hyperparameters of the nonparametric noise density. The $\text{Be}(\alpha_\phi, \beta_\phi)$ prior on $\phi$ controls a priori the number of distinct components in the mixture. When there is no prior information it is desirable to let the model enable a sufficient number of active components in order to reconstruct the true density of the noise. A common choice is $(\alpha_\phi, \beta_\phi) = (1, 1)$ which corresponds to a uniform prior over $\phi$ meaning that no region of the parameter space $(0, 1)$ is favored over another. For the precisions of the random measure, $(\Lambda_k)$, under the absence of prior information, the prior should not convey subjective information to the posterior thus the prior has to be noninformative or vague.

Then the gamma priors should have small shape and rate parameters such that their variance is large. Defining the Bayesian hierarchical model for the weights and biases $\theta$ of the network as in Section 2.3, the variance of the posterior over $\theta^{(i)}$ is controlled through the precisions $\tau_W^{(i)}$ and $\tau_b^{(i)}$ which are coupled through their gamma prior. When there is no prior information, suitable choices for the gamma hyperpriors then should lead to priors with large variance, allowing the encoded functions vary greatly. In contrast, moderating the variance of the prior will lead to functions that do not vary greatly.

### 3. POSTERIOR INFERENCE VIA MARKOV CHAIN MONTE CARLO

We introduce a Markov chain Monte Carlo (MCMC) method for posterior inference with the proposed model. In particular, a Gibbs sampler is provided and all the variables are sampled from their full conditional distributions.
However, as is common, we sample the weights via Hamiltonian Monte Carlo (HMC) (Neal, 1996) within the Gibbs procedure. We will provide more details on HMC when we derive the full conditional distribution for the weights below. Additionally, using the sampled values for the parameters $\theta$ we extend the model for prediction of future unobserved values. For simplicity, from now on we consider scalar observations and consequently, $\Lambda$ stands for precision instead of precision matrix.

3.1. System Identification via MCMC

A single sweep of the sampling algorithm has to sample the variables $(d_t, R_t)_{t=1}^n$, the vector of parameters $\theta$, as well as the weights and atoms of the random measure $\pi_k, \Lambda_k, 1 \leq k \leq R^*$. For density estimation of the noise process, the sampler must additionally sample from the noise predictive distribution

$$
\Pi(d_{z_{n+1}} | z_1:n) = \int \Pi(d_{z_{n+1}} | \mathcal{P}) \Pi(d\mathcal{P} | z_1:n).
$$

(8)

First, for $k = 1, \ldots, R^*$ we construct the geometric weights

$$
\pi_k = \phi (1 - \phi)^{k-1}.
$$

(9)

Then we sample the atoms of the random measure from their posterior distribution. That is, we sample

$$
f(\Lambda_k | \cdots) \propto p_0(\Lambda_k) \prod_{d_j = k} N(y_t | g(y^d, \theta), \Lambda_k^{-1}) \cdot
$$

(10)

for $1 \leq k \leq R^*$. Here, $p_0(\cdot)$ is the density of $P_0$ with respect to the Lebesgue measure.

Having updated the atoms we proceed to the sampling of the clusters. For $t = 1, \ldots, n$ the clustering variable is a sample from the discrete distribution

$$
pr(d_t = k | R_t = r, \cdots) \propto \pi_k N(y_t | g(y^d, \theta), \Lambda_k^{-1}) \cdot 1(k \leq r).
$$

(11)

Given the clustering variables, we sample the $R_t$’s which make the sum finite from a truncated geometric distribution

$$
pr(R_t = r | d_t = k) \propto (1 - \phi)^{r-1} 1(k \leq r).
$$

(12)

Under the $\text{Be}(a_\phi, b_\phi)$ prior, the geometric probability is sampled from its full conditional distribution

$$
f(\phi | \cdots) \propto \phi^{a_\phi + 2n-1} (1 - \phi)^{b_\phi + \sum R_i - n - 1},
$$

(13)

a Beta distribution $\text{Be}(a_\phi + 2n, b_\phi + \sum R_i - n)$.

Now our attention goes to the sampling of the parameters $\theta$ of the neural network. The weights and biases in each layer $l, 1 \leq l \leq L$ are assigned independent Gaussian priors parametrized by precisions $\tau^{(l)}$, $1 \leq l \leq N_l$, that is, $f(\theta^{(l)} | \tau^{(l)}, 1 \leq l \leq L) = N(\theta^{(l)} | 0, \tau^{(l)} \cdot 1)$ where $\theta^{(l)} = \{W^{(l)}, b^{(l)}\}$ is the collection of parameters on layer $l$. Thus, the full conditional distribution for the $\theta$ vector is given by

$$
f(\theta^{(l)} | \cdots) \propto N(\theta^{(l)} | 0, \tau^{(l)} \cdot 1) \prod_{i=1}^n N(y_t | g(y^d, \theta), \Lambda_{a_i}^{-1}).
$$

(14)

This is a nonstandard distribution because the nonlinear function on $\theta$ appearing in the product of Gaussians. We sample this distribution via a Hamiltonian Monte Carlo (HMC) step within the Gibbs sampler. In HMC
(Neal et al., 2011), a new sample is proposed in the parameter space using gradient information of the posterior distribution. Introducing auxiliary variables \( \eta \), where \( \text{dim} \ \eta = \text{dim} \ \theta \), an ergodic Markov chain with stationary distribution the desired posterior distribution \( p(\theta, \eta) = f(\theta^{(i)} | \cdots) \) \( p(\eta) \) is constructed. This posterior can be seen as the canonical distribution of the joint state \( (\theta, \eta) \) corresponding to the Hamiltonian \( H(\theta, \eta) = U(\theta) + T(\eta) = -\log f(\theta^{(i)} | \cdots) + \frac{1}{2} \eta^T M^{-1} \eta \), where \( M \) is a positive definite mass matrix.

Given the current state \( (\theta, \eta) \) of the sampler, an HMC transition is given by first simulating new variables \( \eta' \sim p(\eta) \). Then, the new \( \theta' \) is proposed simulating a trajectory according to Hamilton’s equation of motion \( (\dot{\theta}, \dot{\eta}) = (\partial_\eta H, -\partial_\theta H) \) using leapfrog integrator, a symplectic integrator which alternates between the following updates

\[
\begin{align*}
\eta(k + \epsilon/2) &= \eta(k) - \frac{\epsilon}{2} \partial_\eta H(\theta(k), \eta(k)), \\
\theta(k + \epsilon) &= \theta(k) + \epsilon \partial_\theta H(\theta(k), \eta(k + \epsilon/2)), \\
\eta(k + \epsilon) &= \eta(k + \epsilon/2) - \frac{\epsilon}{2} \partial_\eta H(\theta(k + \epsilon), \eta(k + \epsilon/2)),
\end{align*}
\]

where \( \epsilon \) is the discretization step length, \( S \) is the number of steps giving a trajectory of length \( \epsilon S \), and \( \kappa \) is the time variable. To account for trajectory simulation errors, a Metropolis acceptance step is included and the proposed state is accepted with probability \( \text{acc}(\theta', \theta) = \min(1, p(\theta', \eta') / p(\theta, \eta)) \).

Under the Gamma prior for the hyperparameters \( \tau^{(i)} \), we can update the precisions tied to the prior for each layer by sampling from the semi-conjugate model

\[
f(\tau^{(i)} | \cdots) \propto \text{Ga}(\tau^{(i)} | \alpha, \beta) \ N\left( \tau^{(i)} | 0, \left[ \tau^{(i)} \right]^{-1} \right), \quad 1 \leq i \leq L,
\]

which is a Gamma distribution and easy to sample from.

Finally, we take a sample from the noise predictive given in (8). At each iteration we have weights and locations \((\pi_k, \Lambda_k)\) and we sample a uniform random variable \( u \sim U(0, 1) \). We take \( \Lambda_k \) for which

\[
\sum_{\ell=1}^{k-1} \pi_\ell < u \leq \sum_{\ell=1}^{k} \pi_\ell.
\]

If we run out of weights we sample \( \Lambda_k \) from the prior \( P_0(\Lambda_k) \). Eventually, a sample from the noise predictive is taken from \( N(0, \Lambda_k^{-1}) \).

3.2. Prediction of \( J \) Future Values

It is possible to use the samples obtained from the MCMC described in Section 3.1 to compute functions \( \hat{g} \) in a Monte Carlo fashion. In particular, at each step of the MCMC we have samples \( \theta^{(i)} \) for \( i = 1, \ldots, I \) where \( I \) is the total number of iterations. We can plug in these samples in the neural network and make predictions for the future \( J \) values as

\[
\hat{y}_{n+j}^{(i)} = g \left( y_{n+j}^{\theta^{(i)}} ; \theta^{(i)} \right), \quad j = 1, \ldots, J,
\]

where

\[
y_{n+j}^{\theta^{(i)}} := \left( \hat{y}_{n+j-1}^{(i)}, \ldots, \hat{y}_{n+1}^{(i)}, y_n, \ldots, y_{n+\rho+1} \right).
\]

A biased estimate can be obtained by averaging over the sample of computed functions as

\[
\hat{y}_{n+j} = \frac{1}{I} \sum_{i=1}^{I} g \left( y_{n+j}^{\theta^{(i)}} ; \theta^{(i)} \right), \quad j = 1, \ldots, J.
\]
The associated Monte Carlo error can be computed from the sample variance of predicted values. We note here that similar predictors have been used before in Liang (2005).

4. EXPERIMENTAL EVALUATION

We test the performance of the proposed methodology on real time series namely, the Canadian Lynx data (Campbell and Walker, 1977), quite popular in time series literature. The data consists of the annual trappings of lynx in the Mackenzie River District of North–West Canada for the period from 1821 to 1934 resulting to a total of 114 observations. We aim to identify the process responsible for the generation of the data using the first 100 observations and, additionally, test our model’s predictive capability on the last 14 observations. The source code for the experiment is available at https://github.com/cmerkatas/bayesid.git.

4.1. Experimental Setting

The neural network has the structure \( N_0, 10, 1 \) with tanh activation functions in the hidden layer, where the number of inputs \( N_0 \) will be specified by the lag. As a prior for the geometric probability of the GSB process we assign a uniform \( \text{Be}(\phi | 1, 1) \) distribution. The precisions \( \tau^0 \) for each layer’s 1 \( \leq l \leq L \) weights and biases are having a \( \text{Ga}(\tau^0 | 5, 5) \) Gamma hyper-prior assigned. These choices for the hyperparameters will moderate the precisions of the Normal priors used for the weights of the network (Liang, 2005). The base measure of the GSB process is a Gamma measure \( \text{Ga}(d\Lambda | a_\Lambda, b_\Lambda) \propto \Lambda^{a_\Lambda - 1} \exp(-b_\Lambda\Lambda) d\Lambda \) with \( (a_\Lambda, b_\Lambda) = (0.05, 0.05) \). The MCMC sampler ran for 40000 iterations where the first 2000 samples are discarded as burn-in period and a sample is kept (thinning) after 50 iterations.

We perform a comparison with related models in terms of prediction accuracy. Letting \( \hat{y} \) be the predicted values, we compare the proposed model against under the following metrics:

1. mean square error (MSE): \( \sum_{t=n+1}^{n+J} (\hat{y}_t - y_t)^2 / J \),
2. root mean squared error, the square root of MSE (RMSE): \( \sqrt{\sum_{t=n+1}^{n+J} (\hat{y}_t - y_t)^2 / J} \),
3. mean absolute error (MAE): \( \sum_{t=n+1}^{n+J} |\hat{y}_t - y_t| / J \),
4. mean absolute prediction error (MAPE) defined as: \( \frac{1}{J} \sum_{t=n+1}^{n+J} \left| \frac{\hat{y}_t - y_t}{y_t} \right| \),
5. Theil’s \( U \) statistic (Pindyck and Rubinfeld, 1998): \( U = \frac{1}{\left( \frac{1}{J} \sum_{t=n+1}^{n+J} (\hat{y}_t - y_t)^2 \right)^{1/2}} \left( \frac{1}{J} \sum_{t=n+1}^{n+J} (\hat{y}_t - y_t)^2 \right) \). Theil’s statistic \( U \) takes values in \( [0, 1] \) predictions such that \( U = 0 \) means perfect prediction.

4.2. Analysis

As is common, we take the log\(_{10}\) transformation of the dataset. For the Lynx data, the partial autocorrelation for lags greater than 2 is statistically insignificant, indicating that the time series is generated from an AR(2) process. This, along with the values of the WAIC in Table I indicates to set the number of inputs in the neural net \( N_0 = 2 \) which corresponds to an AR(2) process. After some preliminary trials, the HMC sampler parameters have been set to \((\epsilon, L) = (0.005, 20)\) so that the acceptance rate is close to the optimal value 0.651 (Beskos et al., 2013). We note here that we do not test values \( \rho \geq 10 \) since with fixed \( N_0 = 10 \) these will correspond to linear models. For \( \rho \geq 10 \) it is necessary to increase the hidden layer neurons which might lead to overfitting.

In Figure 1 we present the ergodic means for the number of clusters estimated during the fitting process from which we conclude that the algorithm is mixing well and converges to the stationary distribution.

In Figure 2 we represent the model fitted from the proposed method NP-BNN and the out of sample predictions for the next 14 values. In Table II we provide a comparison in terms of the evaluation metrics introduced in this section between the NP-BNN model and the following models. An AR(11) model and a ARMA(2, 2) model.
Table I. Lag $\rho$ selection using the widely applicable information criterion WAIC

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>WAIC</td>
<td>81.151</td>
<td>6.2605</td>
<td>5.9923</td>
<td>5.7026</td>
<td>3.9773</td>
<td>0.5742</td>
<td>2.9155</td>
<td>1.2139</td>
<td>0.29</td>
</tr>
</tbody>
</table>

Figure 1. Ergodic means for the number of the active clusters during fitting

Figure 2. Fitted time series and 14-step ahead predictions for the Canadian Lynx data (blue dots). The blue line is the mean posterior function computed by the Bayesian neural network and the shaded blue lines represent the uncertainty associated with the function values based on sampled functions. In purple we represent the mean posterior prediction function for the 14 future observations with their associated uncertainties (purple-shaded lines)

for which the lengths have been determined by the AIC criterion (Akaike, 1974). Additionally, a neural network with deterministic weights and an autoregressive Bayesian neural network with parametric, Gaussian noise have been added in the comparison. All neural networks have the same architecture. Finally, using the \texttt{bayesf} package (Frühwirth-Schnatter, 2006), we include the Bayesian Markov switching autoregressive model (Albert and Chib, 1993) with 2 states and lag 2 MSAR(2, 2) selected for predictive performance. This model is able to capture nonlinear dynamics by representing different regions of the state space with different AR models. We note here that AIC indicates a MSAR model with 3 hidden states and lag 11 as the best model however, the predictive capability of this model was the worst among any of the presented models.

\cite{J.Timeser., 2022}
Table II. Evaluation metrics on Canadian Lynx data based on the prediction for the 14 future values. NP-BNN outperforms the models in the comparisons. For each model, the number in parentheses denotes the lags used for estimation.

<table>
<thead>
<tr>
<th>Method</th>
<th>NP-BNN(2)</th>
<th>AR-BNN(2)</th>
<th>ANN(2)</th>
<th>AR(11)</th>
<th>ARMA(2,2)</th>
<th>MSAR(2,2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>0.0322</td>
<td>0.0897</td>
<td>0.0371</td>
<td>0.0822</td>
<td>0.0734</td>
<td>0.0497</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.1792</td>
<td>0.2994</td>
<td>0.1926</td>
<td>0.2866</td>
<td>0.2709</td>
<td>0.2228</td>
</tr>
<tr>
<td>MAE</td>
<td>0.1422</td>
<td>0.2273</td>
<td>0.1487</td>
<td>0.2374</td>
<td>0.2383</td>
<td>0.1677</td>
</tr>
<tr>
<td>MAPE (%)</td>
<td>4.4421</td>
<td>6.9048</td>
<td>4.920</td>
<td>7.995</td>
<td>7.6283</td>
<td>5.1842</td>
</tr>
<tr>
<td>Theil's U</td>
<td>0.0294</td>
<td>0.0502</td>
<td>0.0318</td>
<td>0.0476</td>
<td>0.0448</td>
<td>0.0368</td>
</tr>
</tbody>
</table>

The NP-BNN model outperforms all models in this comparison study with additional advantage that it can be used without any hypotheses about linearity of the time series or Gaussianity of its distribution.

5. CONCLUSION

In this article, we have presented an autoregressive Bayesian neural network with a nonparametric noise model (NP-BNN) for the estimation and prediction of nonlinear time series based on observed time series data. To sample the model we have proposed an MCMC algorithm based on Gibbs sampling. The application of the model on a real data set indicates that the proposed model gives slightly better results than the compared more traditional methods used for nonlinear time series modeling.

ACKNOWLEDGEMENTS

The authors would like to thank the Academy of Finland (project 321891) for funding the research and the two referees for contributing in the significant improvement of the scientific content as well as the presentation of this manuscript.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in the Github repository https://github.com/cmerkatas/bayesid.git.

REFERENCES


