ISSN 2279-9362

Collegio Carlo Alberto

Bayesian functional forecasting with locally-autoregressive dependent processes

Guillaume Kon Kam King, Antonio Canale, Matteo Ruggiero

No. 581
December 2018

Carlo Alberto Notebooks

www.carloalberto.org/research/working-papers

Bayesian functional forecasting with locally-autoregressive dependent processes

Guillaume Kon Kam King
University of Torino and Collegio Carlo Alberto
Antonio Canale
University of Padova
Matteo Ruggiero*
University of Torino and Collegio Carlo Alberto

Motivated by the problem of forecasting demand and offer curves, we introduce a class of nonparametric dynamic models with locally-autoregressive behaviour, and provide a full inferential strategy for forecasting time series of piecewise-constant non-decreasing functions over arbitrary time horizons. The model is induced by a non Markovian system of interacting particles whose evolution is governed by a resampling step and a drift mechanism. The former is based on a global interaction and accounts for the volatility of the functional time series, while the latter is determined by a neighbourhood-based interaction with the past curves and accounts for local trend behaviours, separating these from pure noise. We discuss the implementation of the model for functional forecasting by combining a population Monte Carlo and a semi-automatic learning approach to approximate Bayesian computation which require limited tuning. We validate the inference method with a simulation study, and carry out predictive inference on a real dataset on the Italian natural gas market.

1 Introduction

In this paper we consider the problem of forecasting functional data given by a time series of monotonic step functions that are completely observed and may exhibit local trends over time. As a motivating dataset we consider data from the Italian virtual balancing platform for natural gas trading, which describe the daily offer and demand curves used by the public authority to settle the daily exchange unitary price. In such framework, the interest lies in forecasting the entire curves as these can be used for strategic decision making. Further details are discussed in Section 5.1.

Other approaches to this problem include Canale and Vantini (2016), that introduces a functional autoregressive model for constrained functions and derives one-step ahead point

* Partially supported by MIUR, grant 2008MK3AFZ.

predictions; Rossini and Canale (2018), that extends the previous by including in the model the effect of exogenous scalar variables and by accounting for predictive uncertainty quantification via a block bootstrap technique; Canale and Ruggiero (2016), that proposes a Bayesian nonparametric approach to this forecasting problem based on an underlying system of interacting particles. Each of these approaches left room for improvement, as the first two suffer from a limitation of the available forecast horizon, while the latter showed systematic overestimation of the predictive uncertainty. In particular, this last point can be attributed to the model's inability to account for possible local trends exhibited by the functional time series, resulting in a compensation through higher predictive volatility.

With the aim of reducing the above drawbacks and improving on predictive accuracy over arbitrary forecast horizons, in this paper we propose to extend the approach in Canale and Ruggiero (2016) and construct a Bayesian nonparametric dependent process with locally-autoregressive drift. Such process is still based on an underlying system of interacting particles, with the addition of a mechanism that allows to capture local trend behaviours of the functional time series. This operates by monitoring the past curves in subsets of the particles state space and by inducing appropriate displacements of the particles in each region, in turn determining a drift in the functional time series that acts only locally.

We implement the model for performing predictive inference by combining ideas from the literature on approximate Bayesian computation (ABC), a likelihood-free approach to Bayesian inference based on measuring differences between real and simulated data (see Marin et al., 2012, for a review), which is receiving growing attention as a forecasting tool (Frazier et al., 2018). In a nutshell, an ABC strategy draws sets of parameters at random from a prior distribution and simulates synthetic data from a generative model given the sampled parameters. It then selects, based on carefully chosen summary statistics, a subset of these sampled parameters that have generated data sufficiently similar to the real data. The accepted parameters are then kept as an approximate sample from the posterior distribution and can be used for deriving quantities of interest, such as point estimates or credible regions. Here in particular we combine a semi-automatic summary construction (Fearnhead and Prangle, 2010) with a population Monte Carlo procedure that uses adaptive distances (Prangle, 2016). Together with this implementation strategy, the proposed modelling approach shows a substantial reduction in the predictive uncertainty with respect to the model without drift, while preserving the availability of arbitrary forecast horizons.

The rest of the paper is organised as follows. Section 2 introduces the dependent nonparametric model. This is given by a functional time series induced by an interacting particle system which evolves by resampling and a locally-autoregressive drift mechanism. Section 3

details the inferential strategy for posterior computation based on population Monte Carlo and ABC techniques. After presenting a simulation study in Section 4, Section 5 applies the proposed approach to forecasting demand and offer curves relative to the Italian natural gas market. Finally, Section 6 provides a discussion and some concluding remarks.

2 Locally-autoregressive dependent processes

We introduce a non Markovian dependent process based on locally-autoregressive time series of bounded, non-decreasing step functions. Each of these functions is described—after suitable rescaling—as a cumulative distribution function (cdf). Instead of modelling the functions or the cdfs directly, e.g. through the specification of the jump sizes and locations or by mixing with respect to the cdf parameters, we let these be induced by a set of latent variables, or particles, which allow for a finer control of the local behaviour. Such particles are jointly specified to follow a multivariate stochastic process, whereby the transition of a particle depends on the position of the the others, giving a so-called interacting particle system. The dynamic interaction among these particles then in turn induces a time dependent process for the induced functions.

The particles' evolution is governed by a resampling mechanism, based on a global interaction, that drives the volatility of the functional process, and by a non-homogeneous drift mechanism, which aims at modelling locally the functional trends. The drift acts by monitoring the past behaviour of the time series in certain subsets of the region of interest and by inducing different displacements in different parts of the space as needed. This mechanism allows to separate more efficiently the noise from the signal in the functional data evolution by appropriately accounting for local trends, which ultimately results in an improvement of the predictive accuracy.

Before formally defining such mechanisms, we discuss the transformation of the particles. Here we assume the data allow for rescaling, as is the case in our motivating dataset, and consider a collection $D = \{D_t(\cdot)\}_{t=1,\dots,T}$, $T \in \mathbb{N}$ of time-indexed non decreasing step functions $D_t : [0,1] \to [0,1]$ in the unit square. Each D_t is induced by n atoms $X_t^{(n)} = (X_{t,1}, \dots, X_{t,n})$, with $X_{t,i} \in [0,1]$, via the empirical cdf $D_t(x) = n^{-1} \sum_{i=1}^{n} \mathbb{1}(X_{t,i} \leq x)$. Equivalently, we could group the possible ties in $X_t^{(n)}$, whereby jumps of size $J_{t,j}$ occur at $n_t \leq n$ locations $Z_{t,j}$, giving

$$D_t(x) = \sum_{j=1}^{n_t} J_{t,j} \mathbb{1}(Z_{t,j} \leq x). \tag{1}$$

If the stochastic processes $\{X_{t,i}, t = 1, \ldots, T\}$ are not independent across i, hence their transition function needs to be specified jointly, then $\{X_t^{(n)}, t = 1, \ldots, T\}$ is said to be an interacting particle system. Thus, here each $X_{t,i}^{(n)}$ is a discrete time processes on $[0, 1]$. The transition function of $X_t^{(n)}$ will be implicitly specified by the particles dynamics detailed below, whereas the generic transition function for the single particle can be written $P(X_{t,i} \in A \mid \{X_s^{(n)}, s = t-1, \ldots, t-1-k\})$, so that $X_t^{(n)}$ is Markovian of order $k+1$, the classical Markovian case corresponding to $k = 0$. Note that the number of particles n, by determining the minimal discretisation of the jump sizes, acts as a level of resolution with which the data are modelled and is not to be regarded as a model parameter. Further discussion of this point is postponed to Section 6.

We now describe the two main mechanisms that govern the particles dynamics, which we refer to as the *drift* and the *resampling* step respectively.

Drift step. The drift step aims at capturing the local trends in the observed curves and model future curves accordingly. The mechanism determines a localised, autoregressive interaction among particles, in the sense that each particle is displaced by an amount and in a direction determined by the behaviour of a predetermined number of previous curves in a neighbourhood of the particle. Let $D_{t-1}(x) = n^{-1} \sum_{i=1}^{n} \mathbb{1}(X_{t-1,i} \leq x)$ be the curve at time $t-1$. Define

$$X_{t,i}^* = X_{t-1,i} - \sum_{j=1}^{k} \varepsilon_j \int_{U_{t-1,i}^h} [D_{t-1}(x) - D_{t-1-j}(x)] \mathrm{d}x \tag{2}$$

for $k \geq 0$ and $\varepsilon_1, \ldots, \varepsilon_k > 0$, where, for $h \in [0, 2]$,

$$U_{t-1,i}^h = \Big[X_{t-1,i} - h/2, X_{t-1,i} + h/2\Big] \cap [0, 1]. \tag{3}$$

When $k = 0$, we interpret the above sum as empty, yielding $X_{t,i}^* = X_{t-1,i}$, i.e., null drift. When $X_{t,i}^*$ falls outside $[0, 1]$, we set it equal to the closest boundary, so more formally $X_{t,i}^*$ should be defined as $\max(0, \min(1, x^*))$ where x^* is the right hand side of (2).

The special case for $k = 1$ provides a useful model that lends itself easily to interpretation. The drift step becomes

$$X_{t,i}^* = X_{t-1,i} - \varepsilon \int_{U_{t-1,i}^h} [D_{t-1}(x) - D_{t-2}(x)] \mathrm{d}x, \tag{4}$$

and yields a displacement of the i-th particle from its previous position by an amount proportional to the integrated difference between the last two observed curves, computed

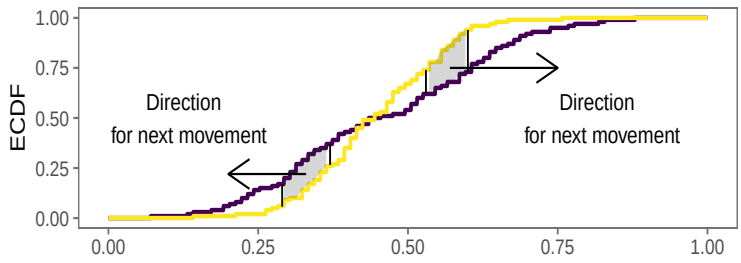

Figure 1: Schematic description of the drift step for $k = 1$, as in (4). The movement of a particle at time t is determined by the (signed) integrated difference between the curve at time $t-1$ (dark-coloured) and that at time $t-2$ (light-coloured), computed in a neighbourhood of the particle.

in the neighbourhood $U^h_{t-1,i}$ of $X_{t-1,i}$. Here $U^h_{t-1,i}$ is chosen to be symmetric around $X_{t-1,i}$ for simplicity, but different formulations can be devised. The quantity h regulates the range of the interaction, and values $h \in [0, 2]$ vary from the complete absence of drift ($h = 0$) to a drift based on global interaction ($h = 2$), in which case $U^h_{t-1,i} \equiv [0, 1]$. Finally, $\varepsilon > 0$ determines the strength of the local interaction, the minus sign being due to the fact that if the local integrated difference between consecutive curves is positive, a similar displacement for the next curve is obtained by moving the particles to the left, and vice versa. Figure 1 depicts the mechanism schematically.

The above defined drift is state-dependent, deterministic conditionally on the last $k + 1$ curves, and induces an overall displacement of the jump locations of D_t, since particles at the same location will imply the same value for (2). Note also that an equivalent formulation in (2) that considers the differences between curves at $t-j$ and $t-j-1$, instead of $t-1$ and $t-1-j$, can be recovered by reparameterising with linear combinations of the coefficients ε_j.

Resampling step. Given the intermediate state of the particles X^*_t yielded by the drift step, the next state is obtained by resampling a random number of particles in X^*_t. Here we let $M \sim \text{Binom}(n, p)$, for $p \in (0, 1)$, and sample M indices uniformly at random from $\{1, \dots, n\}$, denoted here $\{i_{n-M+1}, \dots, i_n\}$. Letting then P_0 be a probability measure on $[0, 1]$, we resample the particles with indices $\{i_{n-M+1}, \dots, i_n\}$ from the Blackwell–MacQueen Pólya

urn scheme (Blackwell and MacQueen, 1973) with parameters $\theta > 0$ and P_0, conditionally on the other $n-M$ particles with indices $\{i_1, \ldots, i_{n-M}\}$. Namely, we have for each $j = 1, \ldots, M$,

$$X_{t,i_{n-M+j}} = \begin{cases} Y \sim P_0, & \text{w.p. } \frac{\theta}{\theta+n-M+j-1}, \\ X_{t,i_w}, & \text{w.p. } \frac{1}{\theta+n-M+j-1}, w = 1, \ldots, n-M+j-1. \end{cases} \tag{5}$$

With probability proportional to θ, a particle is substituted with a new value drawn from P_0, and with the remaining probability it is copied from one of the other particles (including those already resampled). Note that when P_0 is a nonatomic probability measure, Y in (5) is a value not previously observed with probability one.

This resampling mechanism determines a partial reallocation of the jumps, whose sizes are broken down into particle units which are moved across the space. The mass removed from existing jumps is either reallocated to other jumps, chosen with probability proportional to the jump size after the shortening, or is assigned to a new jump, whose location is drawn from P_0. The overall effect is a fluctuation of the current jump sizes and the possible appearance of new jumps. Furthermore, the resampling step can remove particles which ended up at the boundaries with the drift step and bring them back to the interior of the state space. It is interesting to note that the resampling step can be shown to provide, with additional technical conditions and when $n \to \infty$, a weak approximation to the diffusive term of certain classes of measure-valued diffusion processes (see, e.g., Ruggiero and Walker 2009a;b), and can therefore be thought of as providing a *discrete* type of diffusive component of the particle dynamics, which constitutes a more familiar counterpart to a drift term.

3 Model estimation

The parameters of the locally-autoregressive dependent process can be learned for predictive inference purposes by means of a strategy based on approximate Bayesian computation. Here we opt for a likelihood-free approach that circumvents the intractability of the likelihood and is robust with respect to the number of particles used. Indeed, our approach is only based on the structural similarities between the observed curves and a set of synthetic functional time series, measured through summary statistics. Since the summaries are not affected by the number of particles in that, e.g., twice the number of particles allow in principle to reconstruct the same exact curve, the choice of n has minimal impact on the inference, at least for sufficiently large values of n.

We borrow from the recent literature on ABC combined with the semi-automatic summary construction of Fearnhead and Prangle (2010) and the population Monte Carlo ABC

algorithm of Prangle (2016). With the proposed strategy, the model shows improved performance for functional forecasting with respect to the model without drift and compares extremely well, in a simulation study, with *oracle* predictions.

More specifically, let $\pi(D|\xi)$ be a model for generating the datum D given a vector of parameters ξ, $\pi(\xi)$ a prior density for ξ, S a function mapping a dataset into a vector of summary statistics, and d a distance function between vectors of summary statistics. The standard ABC rejection sampling proceeds by sampling candidate parameters $\xi_\ell \overset{\text{iid}}{\sim} \pi(\xi)$, simulating data $D_\ell \sim \pi(\cdot \mid \xi_\ell)$ conditionally on ξ_ℓ, computing summaries $s_\ell = S(D_\ell)$ and $s_{\text{obs}} = S(D_{\text{obs}})$ on the simulated data and on the observed data respectively, and evaluating the associated distances $d_\ell = d(s_\ell, s_{\text{obs}})$ after choosing a suitable distance measure. The candidate parameters ξ_i such that $d_\ell \leq q$, where q is a chosen quantile of the distances computed, are then accepted and can be seen as an approximate sample from the posterior distribution of ξ given the data.

Clearly, the quality of the approximation of the true posterior distribution of the parameters, hence of the estimates to be obtained based on the ABC sample, largely depends on the criterion used for the parameter selection. The challenges for implementing a successful ABC scheme therefore consist in measuring effectively the distance between real and synthetic data, and designing an efficient acceptance rule. The summarising function S should desirably be of low dimension (Beaumont et al., 2002) while capturing as much information as possible on the parameters, the ideal situation being when S is a set of sufficient statistics for the model at hand. When these are not available, the choice of distance becomes critical as it strongly influences the shape and location of the mass of the approximate posterior distribution. Intuitively, this should weight each summary according to its informativeness about the parameters, compensating for the scale of variation and downplaying the influence of uninformative summaries.

In the present framework, we use a semi-automatic data summaries approach in the spirit of Fearnhead and Prangle (2010), combined with a handful of carefully chosen ad hoc summaries that capture peculiar features of the type of data at hand, denoted later $t_1, \ldots, t_7$. The next section (3.1) provides full details on both choices. The semi-automatic approach is based on the observation that when multiple summary statistics are available or the relevant summaries are not immediately identifiable, their relevance can be learnt using regression models where the summaries are the covariates and the parameters are the multivariate responses. The estimated regression coefficients obtained in the dimension reduction then provide weights for the summaries that reflect their predictive power, alleviating the need to be parsimonious in the inclusion of summaries.

3.1 Summary statistics

As first step before the implementation of the main inferential strategy, we run a pilot study and fit a suitable regression model, in order to estimate the semi-automatic summaries. Specifically, let ξ_ℓ be the ℓ-th draw of the vector of parameters from the prior and let D_ℓ be the functional time series generated from $\pi(D_\ell|\xi_\ell)$. We fit the multivariate regression $\xi_\ell = \beta g(D_\ell) + \epsilon_\ell$, where $g(D_\ell)$ is a suitable transformation of the data. Once an estimate $\hat{\beta}$ for β is obtained, we take the quantities $s(D) = \hat{\beta} g(D)$ to be used as summary statistics in the accept-reject routine of the ABC inference. In order to monitor the curves' values at certain spatial locations, we specify g to provide 20 equally spaced quantiles relative to the curves evaluated at the locations $.01, .05, .1, .25, .5, .75, .9, .95, .99$, computed over $t = 1, \dots, T$. To test for sensitivity of this particular choice, the same analysis was run varying the number of quantiles and with finer and coarser locations grid, yielding qualitatively similar results.

Additionally, we insert in the above regression model supplementary predictors given by specific transformations of the data, denoted below $t_1(D), \dots, t_7(D)$. At the end of the section we provide some general comments on the inclusion of these additional covariates. For what concerns instead their specification, aiming at the precision θ in (5) we consider the mean number of jumps in the curves across the time series $t_1(D) = \bar{K} = T^{-1} \sum_{t=1}^{T} K_t$, where K_t is the number of jumps in curve D_t. Regarding the expected resampled proportion p, we consider the average difference between the sums of the squared jumps sizes of consecutive curves, i.e.

$$t_2(D) = \frac{1}{T-1} \sum_{t=2}^{T} \left[\sum_{j=1}^{n_t} (J_{t,j})^2 - \sum_{j=1}^{n_{t-1}} (J_{t-1,j})^2 \right].$$

For the parameters of P_0 in (5), we assume sufficient statistics are available for the chosen parametric family. Since our P_0 is defined on $[0, 1]$, a reasonable choice is to consider a beta(α, β) distribution, with $\alpha, \beta > 0$, so we take

$$t_3(D) = \prod_{t=1}^{T} \prod_{j=1, Z_j \neq 0}^{n_t} Z_{t,j}^{J_{t,j}} \qquad t_4(D) = \prod_{t=1}^{T} \prod_{j=1, Z_j \neq 1}^{n_t} (1 - Z_{t,j})^{J_{t,j}}$$

with the same notation as in (1). Note that here we have excluded locations that happen to be at the boundaries (which is permitted by the drift term). When no locations are at the boundary, the above coincide with the sufficient statistics for the beta parameters. Let now $k = 1$ in (2). Aiming at ε in the resulting drift (4), we consider $t_5(D) = \text{med}[(Z_{t+1,M} - Z_{t,M})^2]$, i.e. the median squared displacement of the largest jump, where $Z_{t,M}$ is the location of the largest jump of D_t, transformed into log scale to bring the predictor to a similar scale as the

other summaries. If at certain times there are ties for the highest jump, which is an event of low probability, the displacement of the largest jump cannot be tracked. When this occurs, we split the time series at these time points and compute the median displacement based on the resulting sub-series.

As summaries that target the global behaviour of the times series, we consider the ergodic average of the curves $t_6(D) = T^{-1}\sum_{t=1}^{T} D_t(\cdot)$, which is informative on the mean of the process, and the ergodic average of the L_2 distances between consecutive curves, i.e.

$$t_7(D) = \frac{1}{T-1}\sum_{t=2}^{T}\sqrt{\int_{[0,1]}\left(D_t(x) - D_{t-1}(x)\right)^2 \mathrm{d}x},$$

which is informative on the functional volatility. t_6 is computed on a discrete grid of $[0, 1]$, which can be uniform or whose choice can be data driven. For our motivating application it is crucial to learn as much as possible near the boundaries of the state space, so we use the quantile function of the beta$(0.5, 0.5)$ distribution computed on a regular grid of 100 points in $[0, 1]$.

Following Fearnhead and Prangle (2010), we also include in the regression model the first four powers of all the above considered predictors, as a simple way to allow for a richer model at a low cost. To limit the effects of multicollinearity, we use partial least squares regression for dimension reduction, implemented using the R-package *plsgenomics*. Lasso regression or elastic-net regression (Friedman et al., 2010) were also tested as alternatives and gave very similar results while proving, in our framework, computationally less efficient. It is known that the type and degree of penalisation in these regression models may impact the results (which in our case concern the pilot run and not the final output of inference). Here we choose the optimal number of components in the partial least squares regression by cross-validation using the method of Boulesteix and Strimmer (2005).

Note that the partial least squares approach cancels the need, in classical ABC, to be parsimonious with the predictors. On the one hand, the semi-automatic regression step filters out irrelevant predictors and prevents them from impacting the inference, hence one incurs little to no additional costs if some non informative summary is included in the regression model. On the other hand, the addition of $t_1(D)$ showed to improve our ability to learn about θ while the addition of $t_j(D)$ for $j = 2, \ldots, 7$ showed to provides an overall improvement of the parameters posterior uncertainty, with respect to using the quantile-based summaries alone.

3.2 Population Monte Carlo ABC

After selecting the above summary statistics, the ABC strategy needs a suitable distance function for selecting the most relevant draws and, when available, some way of improving over sampling directly from the prior. To both these ends, we adopt the population Monte Carlo approach with adaptive distances of Prangle (2016), which dramatically improves the quality of the approximate samples from the posterior along consecutive generations of draws.

The algorithm is structured as follows. The first generation of candidate parameters is sampled from the prior $\pi^{(1)}(\xi) = \pi(\xi)$. At generation $b = 2, \ldots, B$, the proposals are sampled from the mixture

$$\pi^{(b)}(\xi) = \frac{\sum_\ell w_\ell^{(b-1)} \phi(\xi \mid \xi_\ell^{(b-1)}, 2\Sigma^{(b-1)})}{\sum_m w_m^{(b-1)}}, \quad w_\ell^{(b-1)} = \frac{\pi(\xi_\ell^{(b-1)})}{\pi^{(b-1)}(\xi_\ell^{(b-1)})}, \tag{6}$$

where $\phi(\cdot; \varphi, \Omega)$ is the multivariate Gaussian kernel with mean φ and covariance matrix Ω, $\xi_\ell^{(b-1)}$ are the parameters selected from the previous generation, and $\Sigma^{(b-1)}$ is the sample covariance matrix of $\{\xi_\ell^{(b-1)}\}_{\ell \geq 1}$. Here each of the draws $\xi_\ell^{(b-1)}$ selected from the previous generation is perturbed by means of a Gaussian kernel with covariance matrix given by twice the weighted sample covariance of the accepted parameters of the previous generation, calculated using weights $w_\ell^{(b-1)}$. This is then reweighed by $w_\ell^{(b-1)}$, which accounts for the relative importance of the prior weight and the occurrence in the previous generation, as in a sequential importance sampling scheme.

The values $\xi^{(b-1)}$ used in (6) are the best half of generation $b-1$ with reference to the distance function

$$d(s_{\text{obs}}, s_\ell) = \left(\sum_{j=1}^{m} [r_j (s_{\text{obs},j} - s_{\ell,j})]^2 \right)^{1/2}. \tag{7}$$

Here $s_{\text{obs},j}$ and $s_{\ell,j}$ denote the jth summary computed on the real and simulated data respectively, and $r_j = (\text{MAD}_j)^{-1}$ is the inverse median absolute deviation computed on $s_{\ell,j}$, with $\text{MAD}_j = \text{med}(|s_{\ell,j} - \text{med}(s_{\ell,i})|)$. At each generation the distance is thus scaled to reflect the variation within the current summaries computed on the simulated data. This calibration-free automatic scaling is particularly useful if the summaries do not vary on the same scale, as it prevents one summary from dominating all the others in the distance function. Notice that the semi-automatic summary statistics $S(D) = \hat{\beta} g(D)$ estimated through the regression from these transformations are continuous random variables, so MAD_j never vanishes.

Algorithm 1: PMC-ABC draws generation

Data: D_{obs}

Initialise

- Set $N \in \mathbb{N}$, $L \in \mathbb{N}$, $l = 0$, $\alpha \in [0,1]$, $M = \lceil \alpha N \rceil$, $b = 1$, $h^{(1)} = \infty$, $w_i^{(0)} = 1 \quad \forall i$
- Compute $s_{\text{obs}} = S(D_{\text{obs}})$

Repeat

- Set $\mathbf{x} = \emptyset$, $\mathbf{y} = \emptyset$, $\mathbf{z} = \emptyset$, $n = 0$
- **While** $n < N$ **repeat**
  - Sample $\xi^* \sim \pi^{(b)}(\xi)$ as in (6) and $D^* \sim \pi(D|\xi^*)$
  - Compute $s^* = S(D^*)$
  - **If** $\pi(\xi^*) > 0$ **do**
    - if $d^{(k)}(s^*, s_{\text{obs}}) \leq h^{(k)}$ for all $k \leq b$, set $\mathbf{x} = (\mathbf{x}, \xi^*)$ and $\mathbf{y} = (\mathbf{y}, s^*)$
    - else set $\mathbf{z} = (\mathbf{z}, s^*)$
  - Compute $n = \text{card}(\mathbf{x})$
  - if $l = L$ **return** $\{\xi_i^{(k)}, i \leq M, k < b\}$
  - else set $l = l + 1$
- Compute $\sigma_i^{(b)} = \text{MAD}(s_i; s \in \mathbf{y} \cup \mathbf{z})$
- Compute $d_i^* = d^{(b)}(s_i^*, s_{\text{obs}})$ as in (7) with $r_j = 1/\sigma_j^{(b)}$ for $1 \leq i \leq N$ Set $h^{(b+1)}$ to be the M^{th} smallest d_i^* value
- Set $(\xi_i^{(b)})_{1 \leq i \leq M} = \{\xi_i^*; d_i^* \leq h^{(b+1)}\}$
- Set $b = b + 1$

Algorithm 1 summarises the pseudo code for the generation of PMC-ABC samples outlined above. A similar approach to the one outlined above was developed in Bonassi and West (2015).

4 Simulation study

We evaluate the inference strategy presented above by fitting simulated data for a range of parameters which produces realistic functional time series, relatively to our motivating dataset. To emphasise the difference between the present model and that of Canale and Ruggiero (2016), we compare the results obtained with the model presented in Section 2 with those obtained with the model without locally-autoregressive drift, in the following called for short the *autoregressive* and *basic* model respectively. Both are fitted using the

	Mean L_2 distance between consecutive curves	Mean L_2 distance to the ergodic mean
Offer data	4.510^{-2}	4.910^{-3}
Demand data	5.810^{-2}	1.810^{-3}
Simulated data	9.810^{-2}	4.610^{-2}

Table 1: For real and simulated data, L_2 distances between consecutive curve in the time series or with respect to the ergodic mean of the series, averaged over the time series. The latter quantity for the simulated dataset is an order of magnitude higher, showing higher volatility.

same inferential setting and strategy. We also compare them with *oracle* predictions, based on the true model parameters, which provide a benchmark for the best possible prediction with the current model.

We consider a scenario whose parameters produce a challenging dataset, with highly volatile behaviour and exhibition of trends. Table 1 shows some descriptive statistics, highlighting greater volatility for the simulated dataset than for the real dataset (cf. Figure 7 below). Specifically, for simulating the data we set parameter values of $\theta = 40, p = 0.4, \alpha = 0.25, \beta = 0.3, \varepsilon = 4.5$, and use 500 particles to generate a yearly time series of length 365, of which the first 300 curves are used for training the model and the last 65 are kept as test set. Other choices of parameter values were thoroughly tested by combining values in the ranges $\theta \in [10, 60]$, $p \in [0.05, 0.99]$, $\alpha, \beta \in [0.1, 0.9]$, $\varepsilon \in [1.5, 10]$ and the results were qualitatively similar.

We use the locally-autoregressive dependent model with order $k = 1$ (cf. (2)), whose drift is based on the displacement of the last two available curves and was identified as the most interesting model. Concerning the choice of h in (3), an indication of the most interesting range yielding small to moderate local interactions turned out to be $[.025, .075]$, lower values being close to no interaction at all. Our experience showed that while the range $[.075, .15]$ can still be considered of interest, larger values quickly increase the effect of h on the drift in a way that resembles a global interaction. Different choices of h within the range $[.025, .075]$ were tested and showed to provide little qualitative difference in the results, so we chose the midpoint $h = .05$. As for the resampling step, we let $P_0 = \text{beta}(\alpha, \beta)$ be the distribution of new values of the resampled particles. The model unknown parameters are therefore $\theta, p, \alpha, \beta, \varepsilon$.

To complete the prior specification, we choose a mildly informative prior for the precision parameter θ in (5) given by the normal density $\mathcal{N}(20, 20)$ truncated to positive values. Not

constraining θ to low values can be important in order to accommodate for a certain degree of smoothness in the curves, if the data require it. At the same time, very large values for θ are uninteresting if not detrimental, since they force all new particles in (5) to be sampled from the baseline probability measure P_0. This in turn determines a reversion to the mean effect in the functional time series, going in the opposite direction of learning about the recent data and propagating the trends. For α and β we choose uniform priors on $[0, 1]$. These are informative, since they rule out values greater than 1 and constrain the beta(α, β) cdf to a certain range of shapes between the identity line and a wide plateau with most of the jumps near the boundaries. This choice is based on the general shape of the curves, which is similar to those in Figure 7. Finally, we assign a uniform prior on $[0, 1]$ to p, the proportion of particles resampled at each step, and a uniform prior on $[0, 10]$ to ε in (2), as values greater than 10 yield a degenerate behaviour of the curves with all the jumps at the boundaries of the $[0, 1]$ domain.

To assess the regression step in the semi-automatic ABC procedure, we performed a predictive check diagnostic of the regression model trained on the pilot run. Within the range of values with significant mass in the prior distribution, the regression model did not show any particular problem. Concerning the length of the pilot run, note that the semi-automatic summary statistics obtained after the pilot run are random variables, feature that grants the semi-automatic ABC posterior the calibration property (cf. Theorem 1 in Fearnhead and Prangle, 2010). This essentially means that it is enough to reach a number of points sufficient to correctly fit a regression model, so the only concern in deciding the length of the pilot run is the robustness of the semi-automatic summaries. Here we use 3000 iterations for the pilot run. To check for robustness, we verified that both a double number of iterations and a different starting seed with the same number of iterations led to the same posterior distributions. For the population ABC implementation, we used 20000 iterations, which turned out to produce 6 generations of candidate parameters. A higher number of iterations yields strongly diminishing returns, e.g. twice as many iterations produce only 2 additional generations. Therefore we regarded 20000 as a good compromise between computational cost and accuracy of approximation of the true posterior distributions.

For both the simulation study and the application of Section 5 we used Julia as the programming language (Bezanson et al., 2017), and the implementation of Algorithm 1 from the Julia package ABCDistances[1], with the above number of iterations.

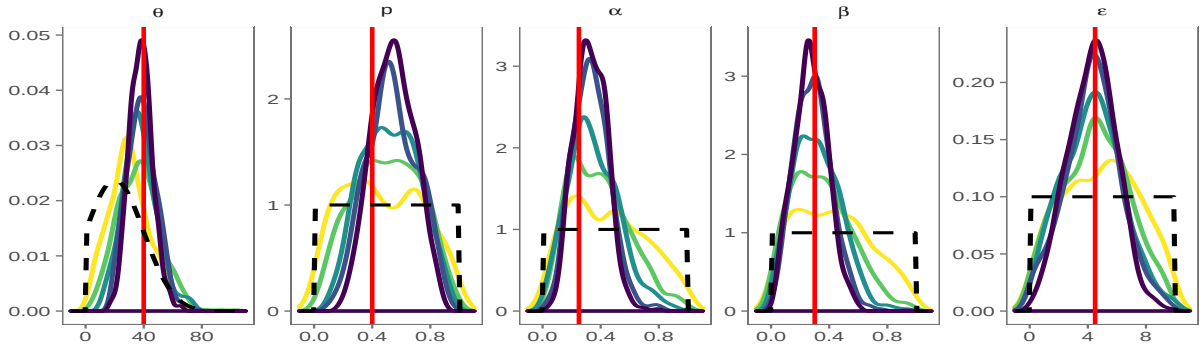

Figure 2: Posterior densities of the parameters for the autoregressive model, with line colours corresponding to successive ABC generations and lines getting darker with each iteration, together with the true parameter values (vertical solid lines) and the prior distributions (dashed lines).

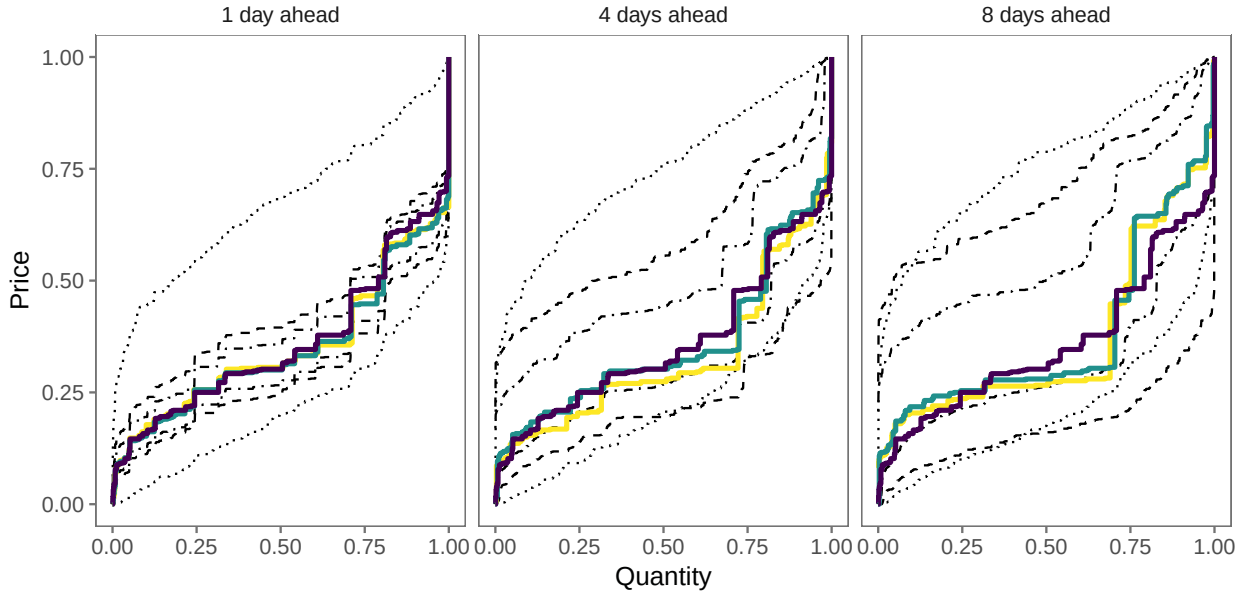

Figure 3: 95% pointwise credible intervals for 1-, 4- and 8-step-ahead forecasts of the darkest solid line, using the two other solid lines. The lighter colour denotes older data. Prediction are obtained with the basic model (dotted), the autoregressive model (dashed) and the oracle prediction (dashed-dotted).

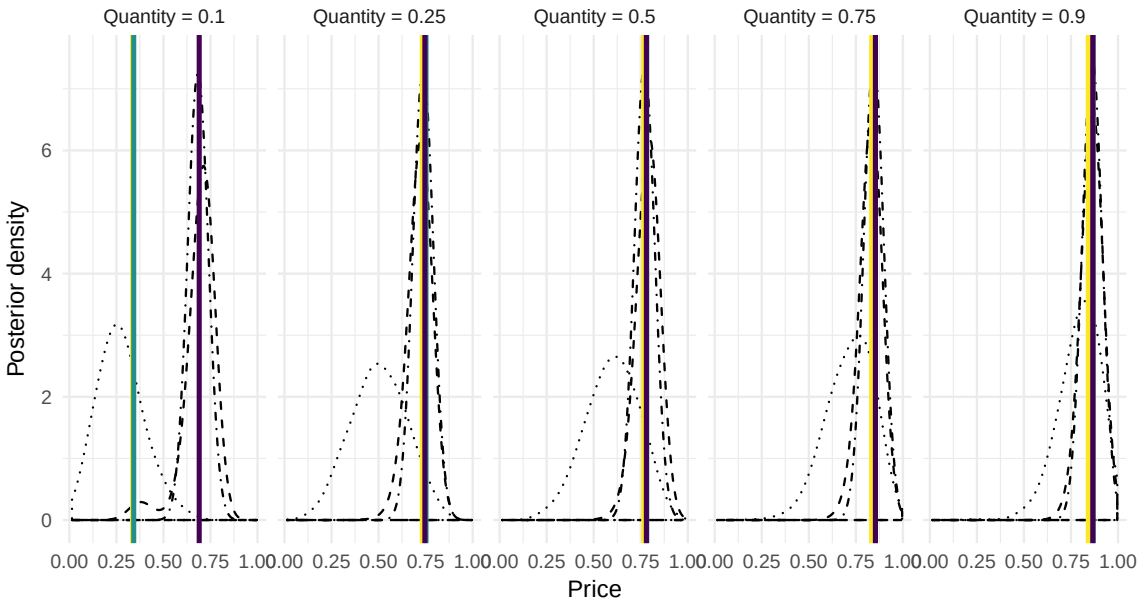

Figure 4: Posterior predictive densities corresponding to five vertical sections of the leftmost panel of Figure 3 for the basic model (dotted), the autoregressive model (dashed) and the oracle prediction (dashed-dotted). The forecasts use the two other solid lines. Lighter colour denotes older data (predicted day - 1 and predicted day - 2).

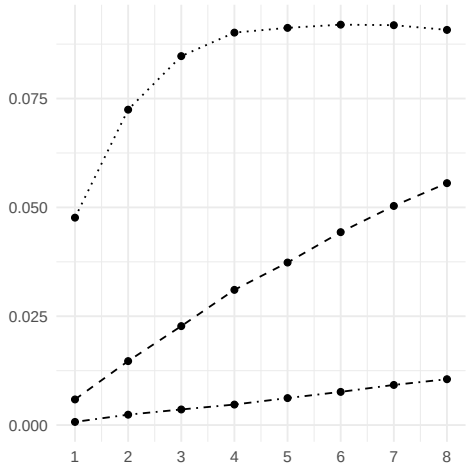

Figure 5: L_2 prediction error for the simulated data, as a function of the forecast horizon (days), for the basic model (dotted), the autoregressive model (dashed) and the oracle prediction (dashed-dotted lines).

Figure 2 shows the PMC-ABC posterior densities of the parameters of the autoregressive model. The posteriors can be seen to concentrate progressively around the true values along subsequent generations, only p showing a little bias. Figure 3 shows pointwise credible regions for the 1-, 4- and 8-step-ahead curve forecasts. For an easier comparison, we plot forecasts of the same curve at t performed from $t-1, t-4, t-8$. Here the basic model accounts for the data dynamics mostly through volatility, whereas the autoregressive model, separating drift and volatility, is able to reduce the uncertainty. This is especially true for the 1-step-ahead forecast, where the difference between the two models is substantial whereas the difference between the autoregressive model and the oracle predictions is moderate. For longer forecast horizons the prediction is challenging for this dataset, as shown by the width of the oracle prediction intervals. Figure 4 shows 1-step ahead posterior forecast densities for five given locations, corresponding to vertical sections of the leftmost panel of Figure 3. The autoregressive model provides forecast densities which are very similar to the oracle prediction, whereas the basic model proves incapable of accounting for the trend in the data and compensates with a larger uncertainty. Finally, Figure 5 shows the L_2 forecast error as a global measure of out-of-sample predictive performance, averaged over the PMC-ABC

[1] Available at https://github.com/dennisprangle/ABCDistances.jl.

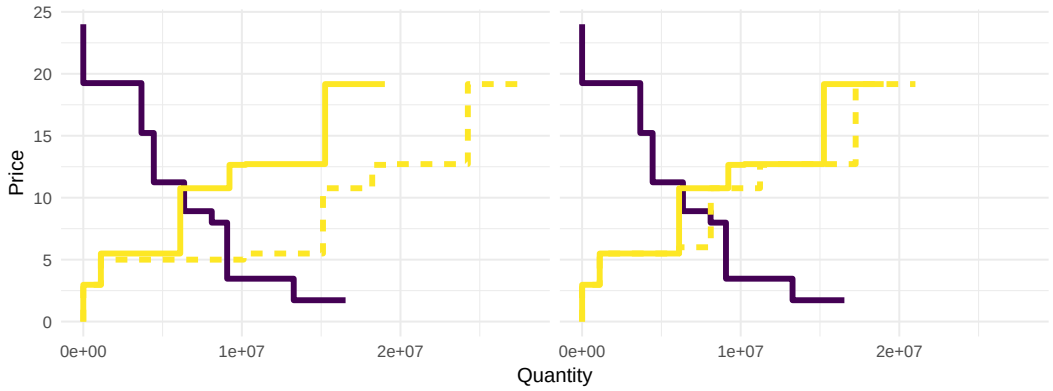

Figure 6: Illustration of the effect of placing an offer bid of $9 \cdot 10^6$ gigajoule (GJ) of gas at 5 Euro/GJ (left) or an offer bid of $2 \cdot 10^6$ GJ of gas at 6 Euro/GJ (right) on the exchange price. Offer bids are first sorted by price and then the related quantities are cumulated so that the abscissas represent the cumulative quantities of gas. Placing an offer bid at 5 Euro/GJ (left) or 6 Euro/GJ (right) increases the amount of gas available at that price, shifting the quantities of gas with higher prices to the right and thus inducing a global shift of the offer curve. The equilibrium price without intervention is identified by the intersection of the estimated curves (solid). After the potential offer bids, the new offer curves (dashed) move also the intersection resulting in a different exchange prices. Changing the amount and price of the bid, it is possible to modify the exchange price to fit one's strategies and needs.

sample and the test set, as a function of the forecast horizon, for the basic, the autoregressive and the oracle prediction. The improvement with respect to the basic model is substantial in comparison with the lowest possible error of the oracle prediction.

5 Forecasting in the Italian natural gas market

5.1 Context

Since the mid 1990's, the European gas market has been gradually converted from a local to a regional scale market through a series of deregulation measures aiming at creating a unique, fluid and competitive market at the continental level (Defeuilley, 2009). One of these measures is the separation of distribution and retail activities, the former being operated by single national entities subject to strict regulations, while the latter have been liberalised.

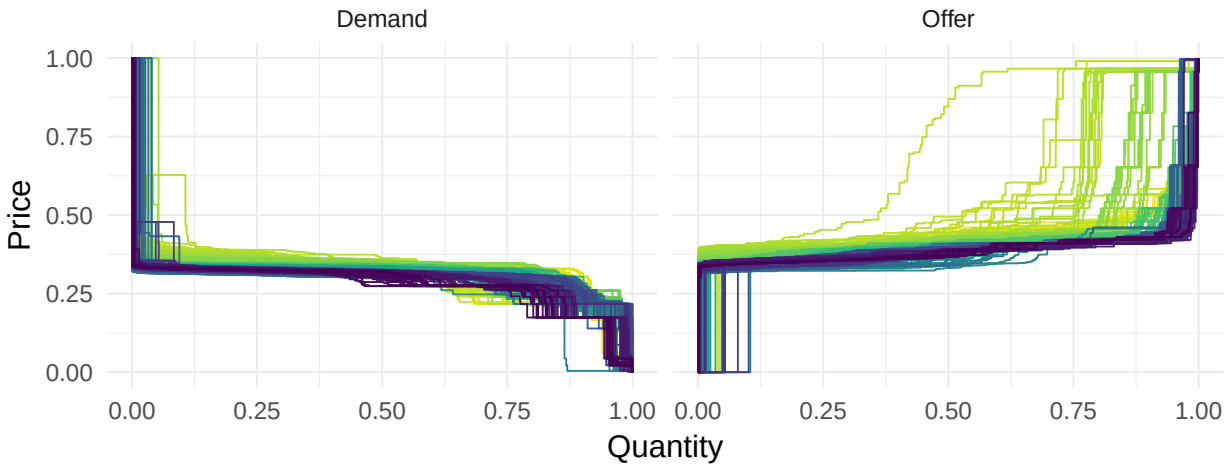

Figure 7: Rescaled daily demand (left) and offer (right) curves from the Italian natural gas market dataset. Darker colours indicate more recent curves.

Here, we consider the Italian natural gas market, where the system recommended by the EU has been implemented through the Italian Natural Gas Balancing Platform. Snam, the entity responsible for gas transportation and storage, aims at the daily compensation of the imbalance between the gas injections and the actual consumption by submitting a demand bid (in case of gas shortage) or a supply offer (in case of gas excess) on the balancing platform market for a volume equal to the global imbalance. On the other hand, operators on the gas network submit their own bids and offers according to their gas or storage availability, at the price at which they deem the operation profitable. Sorting all bids by increasing price and cumulating the related quantities then provides the offer and demand curves, which are the price to be paid per gigajoule (GJ) as a function of the quantity of gas bought or sold. The market is organised such that the intersection of these curves determines the daily exchange price, and all offers below and all demands above the intersection are accepted and performed at the exchange price. Bidding decisions are based on the evaluation of the profitability which depends on the predicted price as a function of the traded volume. To make those decisions, the traders need to account for the effect of their own next bid, which affects the intersection price. Figure 6 illustrates schematically this point. The main statistical goal in the present context is therefore to forecast the entire demand and offer curves, which can be used for deriving price predictions and to evaluate the effect of possible

bids on the equilibrium price.

5.2 Model specification and implementation

We focus our analysis on a dataset on the Italian Natural Gas Balancing Platform consisting of 366 daily demand and offer curves for the leap year 2012. Let x denote the gas quantity (in GJs) and let $t = 1, \ldots, 366$ index consecutive days. We use the first 300 for training the algorithm and retain the last 66 as a test set for out-of-sample prediction. Let $Y_t^{\text{off}}(x)$ denote the offer price for the quantity x at day t, and consider the rescaling

$$y_t^{\text{off}}(x) = \frac{1}{23} Y_t^{\text{off}} \left(\frac{x - L_t}{R_t - L_t} \right)$$

where L_t and R_t are respectively the leftmost and rightmost jump location of the curve on day t. Recall also that the curves are bounded between 0 and 23 Euro/GJ by regulation, so $y_t^{\text{off}} : [0,1] \to [0,1]$. The demand curves are similarly rescaled and inverted to obtain a monotone increasing function, denoted by $y_t^{\text{dem}}(x)$. Figure 7 shows the dataset after the rescaling.

In 2012, Snam provided very precise estimates of the location of the first (or leftmost) jump of each curve, representing the amount and sign of network imbalance. Since deviations from this estimate and the true values are negligible, we consider this parameter as known and fixed. The location of the last (or rightmost) jump is subject to slightly different constraints than the rest of the curve[2], which makes it relatively independent from the functional dynamics and the overall shape of the curves. We choose therefore to model the location of the last jump independently from the rest of the curve, assuming

$$r_t = \rho r_{t-1} + \varepsilon, \quad r_t = \ln R_{t+1} - \ln R_t, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2), \tag{8}$$

where R_t is the rightmost jump of the curve on day t. Prediction is then performed by combining the predictions of the scaled curves with predictions for the largest jump. We assign non informative priors $\rho \sim \mathcal{N}(0, 1000)$ and $\sigma^2 \sim \text{I-Ga}(0.01, 0.01)$ where I-Ga denotes the inverse gamma distribution. The posterior computation for (8) is performed via MCMC sampling using Stan (Stan Developement Team, 2015) (details available in the Supplementary Material).

[2]If some of the traders are unwilling to sell or buy, they can submit an offer at 23 or a demand bid at 0 and not perform any transaction.

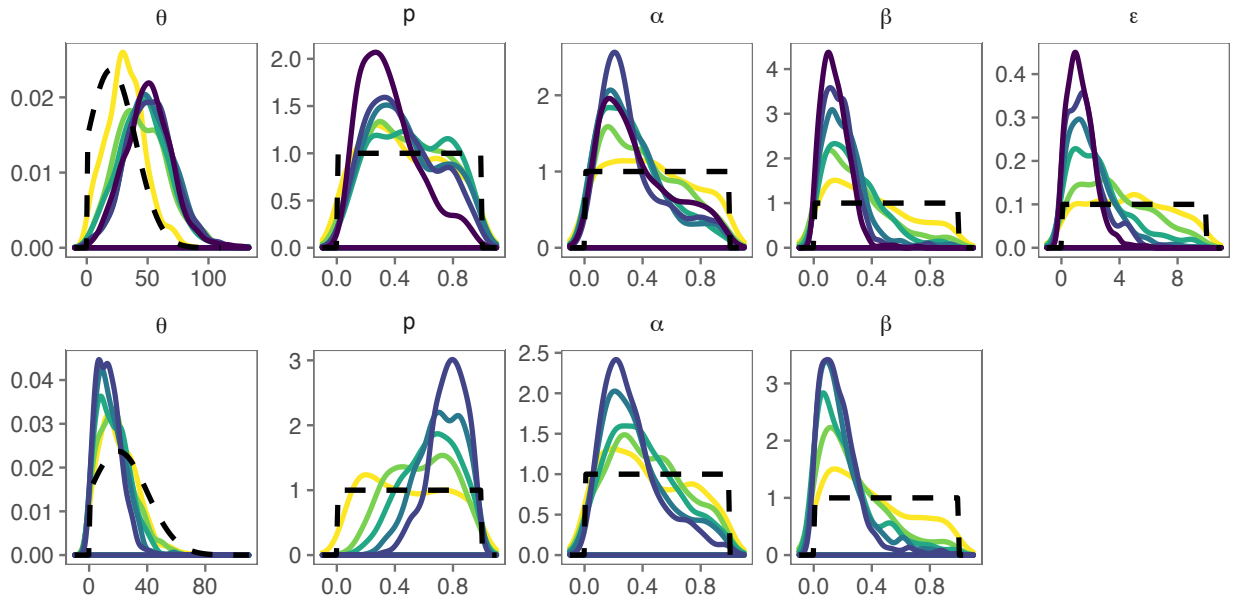

Figure 8: Posterior densities of the parameters of the autoregressive (top) and of the basic model (bottom) fitted to the gas data. Darker lines correspond to successive ABC generations, dashed lines indicate the priors.

For the curve dynamics, we use the same prior specification as the simulation study. Namely, we use a locally-autoregressive dependent model of order $k = 1$, let $h = 0.05$ and $P_0 = \text{beta}(\alpha, \beta)$, and assign priors $\theta \sim \mathcal{N}(20, 20)\mathbb{1}(\theta > 0)$, $\alpha, \beta, p \overset{\text{iid}}{\sim} \text{U}(0, 1)$ and $\varepsilon \sim \text{U}(0, 10)$. Models based on 500 and 2000 particles were tested and provided qualitatively similar results. The former requires a shorter computational effort, but here we show the results using the latter, which is still a good compromise between computational efficiency and resolution of the jump sizes. The computing time amounts to approximately 10 hours on a single laptop core (Intel i-7@2.60GHz, Unix, 16 GB RAM) for the offer dataset with the autoregressive model. The computations for the offer and demand datasets are independent and can be run at the same time on different cores. The pilot runs depend only on the model and can be shared between the offer and demand set of curves.

As in the simulation study, we compare the predictive performance obtained with the autoregressive model with those obtained using the basic model when fitted using the same inferential strategy and setting.

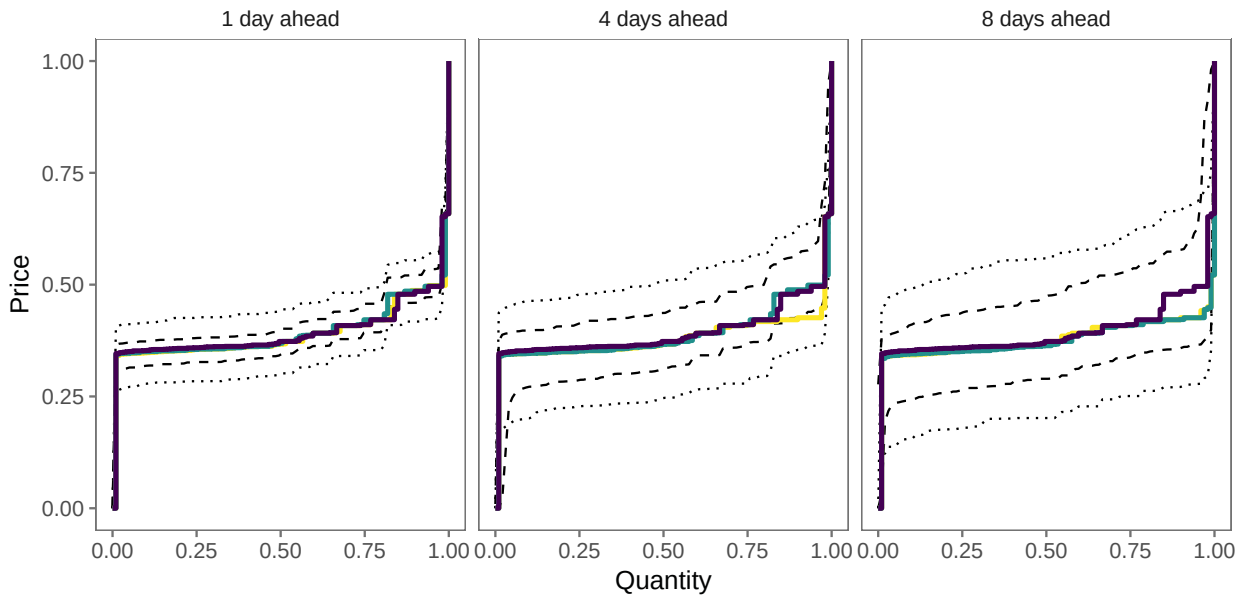

Figure 9: 1-, 4- and 8-step-ahead curve forecasts of the 345th day for the Italian gas market dataset. The solid lines denote the real data, the darkest being the most recent to be predicted, and pointwise credible intervals are given for the basic (dotted) and the autoregressive model (dashed).

5.3 Results

Figure 8 shows the PMC-ABC posterior densities for the parameters of the autoregressive model and of the basic model. Here we observe that the basic model, lacking a drift mechanism (hence the parameter ε), tends to account for the time series dynamics using volatility, which results in pushing the resampling rate p close to 1. The autoregressive model is instead able to separate drift and volatility by exploiting the presence of an additional parameter. Figure 9 shows forecasts of the 345th day curve, performed from 1, 4 and 8 days before, where the autoregressive model gains in terms of predictive uncertainty with respect to the basic model. This improvement carries over to the one-dimensional price forecast obtained as the intersection of the demand and offer curves, which is one of the quantities of interest for the bidders. Figure 10 shows the posterior 1-, 4- and 8-step-ahead prediction densities for the price on the last day of the test set.

An overall picture of the quality of these estimates is provided by summarising measures

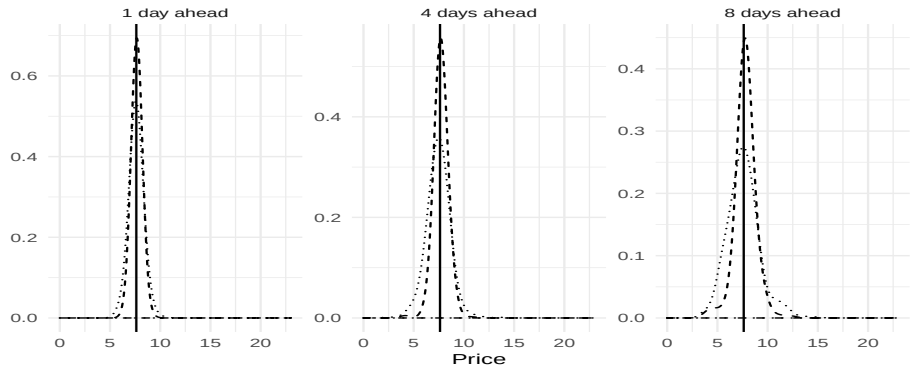

Figure 10: 1-, 4- and 8-step-ahead price forecast using the basic model (dotted lines) and the autoregressive model (dashed lines), the vertical solid line being the target of prediction.

of predictive accuracy for the offer data and the price data. Here we consider for the former the L_2 prediction error in the transformed scale and for the latter the root mean squared error. Both are averaged over the ABC sample and the whole test set, as a function of the forecast horizon. These are provided in Figure 11 and show an appreciable improvement over the basic model. The comparison of the two approaches in terms of forecasting power thus substantially supports the introduction of the drift mechanism. Computation of the *continuous ranked probability score* (Gneiting and Raftery, 2007) for the price data gave similar results.

6 Discussion and concluding remarks

We have introduced a non Markovian Bayesian nonparametric dependent process for functional forecasting with locally-autoregressive behaviour that substantially improves over the model without drift of Canale and Ruggiero (2016) and retains the availability of all forecast horizons. An aspect of the proposed approach is the presence of two tuning parameters, the number of particles n and the autoregression bandwidth h. Our formulation considers the number of particles n not as a model parameter but as a tuning of the precision for describing the curves shape, similar to retaining a given number of digits in a numerical experiment. Although it is tempting to try to deduce a minimum resolution from all the jump values in

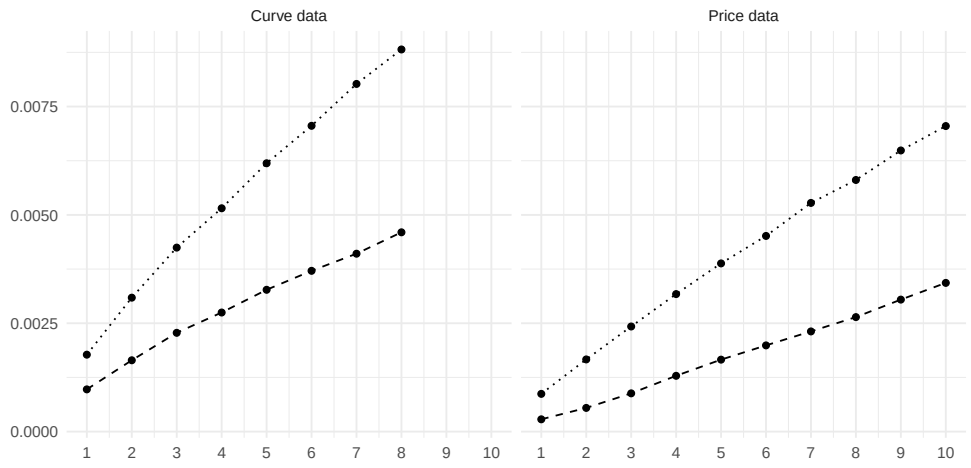

Figure 11: L_2 prediction error (left) and root mean squared error (right) as measures of predictive accuracy for the functional and the price forecasts respectively, as a function of the forecast horizon. The basic model is denoted by the dotted lines, while the autoregressive model is denoted by the dashed lines.

the observed time series, new jumps in future data points may require a different resolution level. Furthermore, a suitable maximum value for n cannot be determined, as using, say, $2n$ particles would also satisfy the same minimum resolution requirement and yield at least as good a prediction. Hence the value for n should be determined by a compromise between the desired accuracy and computational cost. Similarly, h is a tuning parameter related to the type of interactions one wants to model, and its choice could be determined by external factors. In our case, we observed little sensitivity to the particular value chosen within the interesting range (cf. Section 4), but in the context of another application choosing h might require a different approach.

Among potential further developments of the present proposal, one possibility would be to model jointly the demand and supply curves by means of a bivariate functional process. This would allow to borrow information on the supply side in order to anticipate changes in the demand side or vice-versa, possibly leading to more accurate predictions. Operationally, such model could for example be based on an appropriate specification of the bivariate system of dynamically interacting Pólya urns introduced in Prünster and Ruggiero (2013), perhaps together with an accordingly more elaborate drift mechanism. Another appealing extension would be to enlarge the model to account for covariates that provide exogenous information,

such as, for example, specific financial indices and seasonal/meteorological variables known to be correlated with the traded quantities/prices (Rossini and Canale, 2018). These tasks will be pursued elsewhere.

An interesting specificity of our inferential strategy is that it could accommodate quite naturally online learning when small amounts of new data become available at successive times. Extending previously simulated time series by a few steps to match the new length is computationally cheap. All simulations can then be recycled in the semi-automatic summary selection step, including those used in the accept-reject algorithm. Summaries should be re-computed, but the operation is much faster than the data simulation. Furthermore, one could start the population Monte-Carlo from a proposal density given by the posterior distribution conditional on the previous batch of data. Then the overall posterior update in light of the new data should be reasonably faster than running the inference anew.

Acknowledgements

The last two authors are partially supported by the Italian Ministry of Education, University and Research (MIUR) through PRIN 2015SNS29B.

References

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). "Approximate Bayesian computation in population genetics." *Genetics*, 162(4): 2025–2035.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). "Julia: A Fresh Approach to Numerical Computing." *SIAM Review*, 59: 65–98.

Blackwell, D. and MacQueen, J. B. (1973). "Ferguson distributions via Pólya urn schemes." *The annals of statistics*, 353–355.

Bonassi, F. V. and West, M. (2015). "Sequential monte carlo with adaptive weights for approximate bayesian computation." *Bayesian Analysis*, 10(1): 171–187.

Boulesteix, A.-L. and Strimmer, K. (2005). "Predicting transcription factor activities from combined analysis of microarray and ChIP data: a partial least squares approach." *Theoretical Biology and Medical Modelling*, 2(1): 23.

Canale, A. and Ruggiero, M. (2016). "Bayesian nonparametric forecasting of monotonic functional time series." *Electronic Journal of Statistics*, 10(2): 3265–3286.

Canale, A. and Vantini, S. (2016). "Constrained functional time series: Applications to the Italian gas market." *International Journal of Forecasting*, 32(4): 1340–1351.

Defeuilley, C. (2009). "Le gaz naturel en Europe Entre libéralisation des marchés et géopolitique." *Flux*, (75): 99–111.

Fearnhead, P. and Prangle, D. (2010). "Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation." *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 74(3): 419–474.

Frazier, D. T., Maneesoonthorn, W., Martin, G. M., and McCabe, B. P. M. (2018). "Approximate Bayesian Forecasting." *International Journal of Forecasting*, (to appear).

Friedman, J., Hastie, T., and Tibshirani, R. (2010). "Regularization paths for generalized linear models via coordinate descent." *Journal of statistical software*, 33(1): 1.

Gneiting, T. and Raftery, A. E. (2007). "Strictly Proper Scoring Rules, Prediction, and Estimation." *Journal of the American Statistical Association*, 102(477): 359–378.

Marin, J. M., Pudlo, P., Robert, C. P., and Ryder, R. J. (2012). "Approximate Bayesian computational methods." *Statistics and Computing*, 22(6): 1167–1180.

Prangle, D. (2016). "Adapting the ABC distance function." *Bayesian Analysis*, 1–21.

Prünster, I. and Ruggiero, M. (2013). "A Bayesian nonparametric approach to modeling market share dynamics." *Bernoulli*, 19(1): 64–92.

Rossini, J. and Canale, A. (2018). "Quantifying prediction uncertainty for functional-and-scalar to functional autoregressive models under shape constraints." *Journal of Multivariate Analysis*, (to appear).

Ruggiero, M. and Walker, S. G. (2009a). "Bayesian nonparametric construction of the Fleming-Viot process with fertility selection." *Statistica Sinica*, 19(2): 707–720.

— (2009b). "Countable representation for infinite dimensional diffusions derived from the two-parameter poisson-dirichlet process." *Electronic Communications in Probability*, 14: 501–517.

Stan Developement Team (2015). "Stan: A C++ library for probability and sampling, Version 2.8.0."